HP A5820X & A5800 Switch Series

MPLS

Configuration Guide

Abstract

This document describes the software features for the HP A Series products and guides you through the software configuration procedures. These configuration guides also provide configuration examples to help you apply software features to different network scenarios.

This documentation is intended for network planners, field technical support and servicing engineers, and network administrators working with the HP A Series products.

Part number: 5998-1634
Software version: Release 1211
Document version: 5W100-20110430

Legal and notice information

Contents

Configuring MCE

The term *router* in this document refers to both routers and Layer 3 switches.

The Layer 3 Ethernet interface in this document refers to an Ethernet port that can perform IP routing and inter-VLAN routing. You can set an Ethernet port as a Layer 3 Ethernet interface by using **port link-mode route** (see Layer 2—LAN Switching Configuration Guide).

This chapter covers MCE configuration. For information about routing protocols, see *Layer 3—IP Services Configuration Guide*.

MPLS L3VPN

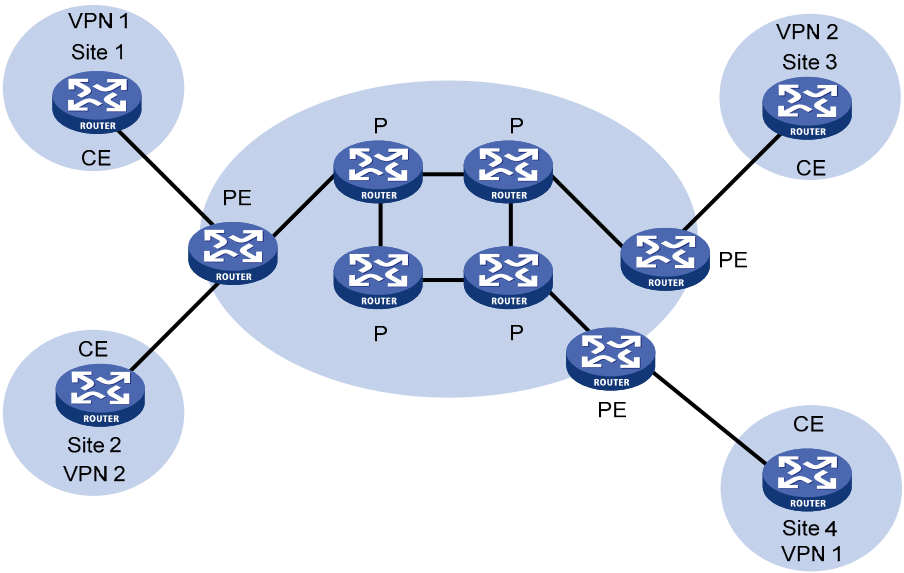
MPLS L3VPN is a kind of PE-based L3VPN technology for service provider VPN solutions. It uses BGP to advertise VPN routes and uses MPLS to forward VPN packets on service provider backbones.

MPLS L3VPN provides flexible networking modes, excellent scalability, and convenient support for MPLS QoS and MPLS TE. Hence, it is widely used.

The MPLS L3VPN model consists of three kinds of devices:

- **CE device**—A CE resides on a customer network and has one or more interfaces directly connected with service provider networks. It can be a router, a switch, or a host. It neither can "sense" the existence of any VPN nor needs to support MPLS.
- **PE device**—A PE resides on a service provider network and connects one or more CEs to the network. On an MPLS network, all VPN processing occurs on the PEs.
- **P device**—A P device is a core device on a service provider network. It is not directly connected with any CE. It only needs to be equipped with basic MPLS forwarding capability.

Figure 1 Network diagram for MPLS L3VPN model



CEs and PEs mark the boundary between the service providers and the customers.

After a CE establishes adjacency with a directly connected PE, it advertises its VPN routes to the PE and learns remote VPN routes from the PE. A CE and a PE use BGP/IGP to exchange routing information. You can also configure static routes between them.

After a PE learns the VPN routing information of a CE, it uses BGP to exchange VPN routing information with other PEs. A PE maintains routing information about only VPNs that are directly connected, rather than all VPN routing information on the provider network.

A P router only maintains routes to PEs and does not deal with VPN routing information.

When VPN traffic travels over the MPLS backbone, the ingress PE functions as the ingress LSR, the egress PE functions as the egress LSR, while P routers function as the transit LSRs.

MPLS L3VPN concepts

Site

Sites are often mentioned in the VPN. 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 to implement.
- 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 adjacent to each other geographically in most cases.
- The devices at a site can belong to multiple VPNs.
- A site is connected to a provider network through one or more CEs. A site can contain many 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.

Address space overlapping

Each VPN independently manages the addresses that it uses. The assembly of such addresses for a VPN is called an address space.

The address spaces of VPNs may overlap. For example, if both VPN 1 and VPN 2 use the addresses on network segment 10.110.10.0/24, address space overlapping occurs.

VPN instance

In MPLS VPN, routes of different VPNs are identified by VPN instance.

A PE creates and maintains a separate VPN instance for each VPN at a directly connected site. Each VPN instance contains the VPN membership and routing rules of the corresponding site. If a user at a site belongs to multiple VPNs at the same time, the VPN instance of the site contains information about all VPNs.

For independence and security of VPN data, each VPN instance on a PE maintains a relatively independent routing table and a separate LFIB. VPN instance information contains these items: the LFIB, IP routing table, interfaces bound to the VPN instance, and administration information of the VPN instance. The administration information of the VPN instance includes the RD, route filtering policy, and member interface list.

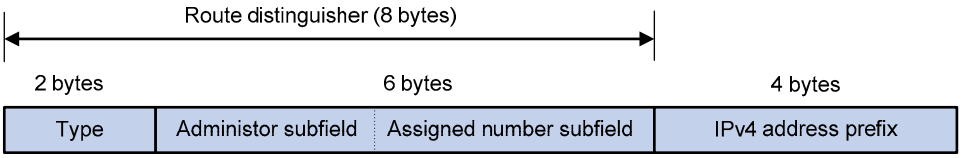
VPN-IPv4 address

Traditional BGP cannot process VPN routes which have overlapping address spaces. If, for example, both VPN 1 and VPN 2 use addresses on the segment 10.110.10.0/24 and each advertise a route to the segment, BGP selects only one of them, which results in loss of the other route.

PEs use MP-BGP to advertise VPN routes, and use VPN-IPv4 address family to solve the problem with traditional BGP.

A VPN-IPv4 address consists of 12 bytes. The first eight bytes represent the RD, followed by a 4-byte IPv4 address prefix, as shown in Figure 2.

Figure 2 VPN-IPv4 address structure



When a PE receives an ordinary IPv4 route from a CE, it must advertise the VPN route to the peer PE. The uniqueness of a VPN route is implemented by adding an RD to the route.

A service provider can independently assign RDs provided the assigned RDs are unique. Thus, a PE can advertise different routes to VPNs even if the VPNs are from different service providers and are using the same IPv4 address space.

Configure a distinct RD for each VPN instance on a PE, so that routes to the same CE use the same RD. The VPN-IPv4 address with an RD of 0 is in fact a globally unique IPv4 address.

By prefixing a distinct RD to a specific IPv4 address prefix, you get a globally unique VPN IPv4 address prefix.

An RD can be related to an AS number, in which case it is the combination of the AS number and a discretionary number; or it can be related to an IP address, in which case it is the combination of the IP address and a discretionary number.

An RD can be in one of the following three formats distinguished by the Type field:

- When the value of 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 value of 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 value of 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 RDs, do not set the Administrator subfield to any private AS number or private IP address.

VPN target attributes

MPLS L3VPN uses the BGP extended community attributes called VPN target attributes, or route target attributes, to control the advertisement of VPN routing information.

A VPN instance on a PE supports two types of VPN target attributes:

- Export target attribute: A local PE sets this type of VPN target attribute for VPN-IPv4 routes learnt from directly connected sites before advertising them to other PEs.
- Import target attribute: A PE checks the export target attribute of VPN-IPv4 routes advertised by other PEs. If the export target attribute matches the import target attribute of the VPN instance, the PE adds the routes to the VPN routing table.

In other words, VPN target attributes define which sites can receive VPN-IPv4 routes, and from which sites that a PE can receive routes.

Like RDs, VPN target attributes can be 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.

Multi-VPN-instance CE

Using tunnels, MPLS L3VPN implements private network data transmission over the public network. However, the traditional MPLS L3VPN architecture requires that each VPN instance exclusively use a CE to connect with a PE, as shown in Figure 1.

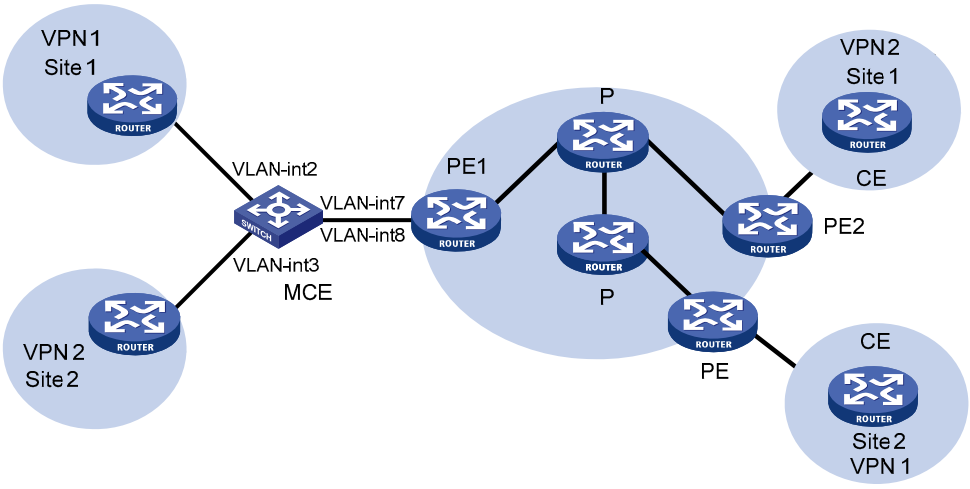
For better services and higher security, a private network is usually divided into multiple VPNs to isolate services. To meet these requirements, you can configure a CE for each VPN, which increases users' device expenses and maintenance costs. Or, you can configure multiple VPNs to use the same CE and the same routing table, which sacrifices data security.

Using the MCE function of the Ethernet switches, you can remove the contradiction of low cost and high security in multi-VPN networks. With MCE configured, a CE can bind each VPN in a network with a VLAN interface on the CE, and create and maintain a separate routing table (multi-VRF) 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.

How MCE works

The following uses the networking illustrated in Figure 3 as an example to introduce how an MCE maintains the routing entries of multiple VPNs and how an MCE exchanges VPN routes with PEs.

Figure 3 Network diagram for the MCE function



On the left-side network, there are two VPN sites, both of which are connected to the MPLS backbone through the MCE device. VPN 1 and VPN 2 on the left-side network must establish a tunnel with VPN 1 and VPN 2 on the right-side network respectively.

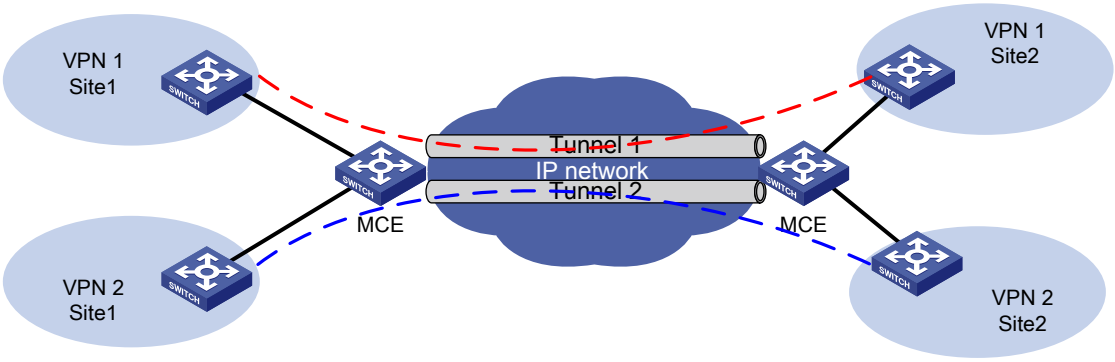
With MCE enabled, routing tables can be created for VPN 1 and VPN 2 individually, VLAN-interface 2 can be bound to VPN 1, and VLAN-interface 3 can be bound to VPN 2. When receiving a piece of routing information, MCE determines the source of the routing information according to the number of the interface receiving the information and then maintains the corresponding routing table accordingly.

You must also to bind the interfaces to the VPNs on PE 1 in the same way as those on the MCE device. The MCE device is connected to PE 1 through a trunk, which permits packets of VLAN 2 and VLAN 3 with VLAN tags carried. In this way, PE 1 can determine the VPN a received packet belongs to according to the VLAN tag of the packet and passes the packet to the corresponding tunnel.

Using MCE in tunneling applications

In addition to MPLS L3VPN, tunneling technologies can also be used to implement other types of VPNs. The MCE function provided by A5800&A5820X switches can be applied in VPN applications based on tunneling, as shown in Figure 4.

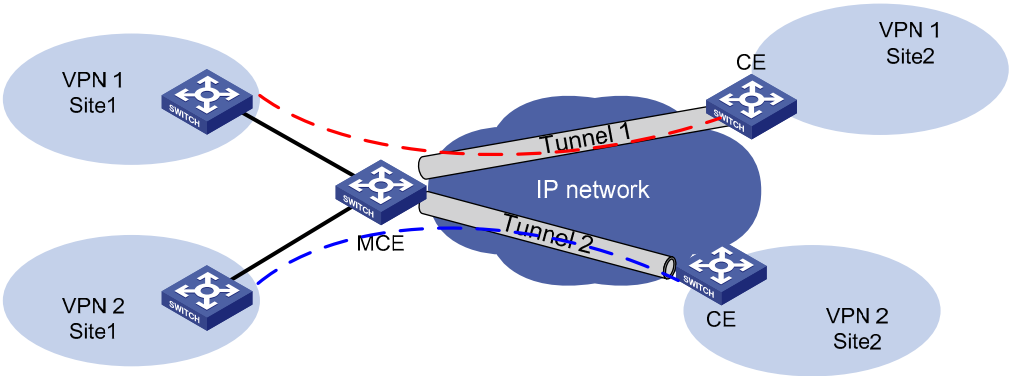
Figure 4 Network diagram for using MCE in a tunneling application (1)



By establishing multiple tunnels between two MCE devices and binding the tunnel interfaces with VPN instances, you can make the routing information and data of the VPN instances delivered to the peer devices through the bound tunnel interfaces. According to the tunnel interfaces receiving the routes, an MCE device determines the VPN instances that the routes belong to and advertises the routes to the corresponding sites. As shown in Figure 4, you can bind Tunnel 1 with VPN 1 to make the MCE devices deliver the routing information and data of VPN 1 through the tunnel.

An MCE can also be used in a tunneling application as shown in Figure 5 to connect with multiple remote CEs through tunnels. In this scenario, the CE devices only need to receive and advertise routes as usual, while the MCE advertises and receives VPN routing information based on the bindings between tunnel interfaces and VPNs.

Figure 5 Network diagram for using MCE in a tunneling application (2)



MCE devices in a tunneling application can exchange VPN routing information with their peer MCE devices or CE devices directly, just as MCE devices in an MPLS L3VPN application do with the corresponding PEs. For more information, see "Route exchange between an MCE and a PE."

These types of tunnels support MCE: GRE tunnel, IPv4 over IPv4 tunnel, and IPv4 over IPv6 tunnel.

For introduction and configuration of tunnel types, see *Layer 3—IP Services Configuration Guide*.

Routing information exchange on an MCE

Interface-to-VPN-instance binding enables MCEs and PEs to determine the sources of received packets and then forward the packets according to the routing information concerning the corresponding VPNs. The following sections describe the way how MCE transmits the private routing information of multiple VPNs to PEs properly.

Route exchange between an MCE and a VPN site

An MCE can adopt the following routing protocols to exchange VPN routes with a site:

- Static route
- RIP
- OSPF
- IS-IS
- IBGP
- EBGP

This introduces the cooperation of routing protocols and MCE in brief. For information about the routing protocols, see *Layer 3—IP Routing Configuration Guide*.

Static routes

An MCE can communicate with a site through static routes. As static routes configured for traditional CEs take effect globally, address overlapping between multiple VPNs remains a problem till the emergence of MCE. MCE allows static-route-to-VPN-instance binding, which isolates the static routes of different VPNs.

RIP

An A5800&A5820X switch can bind RIP processes to VPN instances. With these bindings on the MCE, private network routes of different VPNs can be exchanged between MCE and sites through different RIP processes, thus isolating and securing VPN routes.

OSPF

An A5800&A5820X switch can bind OSPF processes to VPN instances and isolate the routes of different VPNs.

For an OSPF process bound to a VPN instance, the router ID of the public network configured in system view is invalid. So you must specify the router ID when creating an OSPF process.

An OSPF process can be bound to only one VPN instance, however, a VPN instance can use multiple OSPF processes for private network route transmission. To make sure routes can be advertised properly, you must configure the same domain ID for all OSPF processes bound to a VPN instance.

Normally, when an OSPF route is imported to the BGP routing table as a BGP route on a PE, some attributes of the OSPF route get lost. When the BGP route is imported to the OSPF routing table on the remote CE, not all attributes of the original OSPF routes can be restored. As a result, the route cannot be distinguished from the routes imported from other domains. To distinguish OSPF routes imported from different OSPF domains, the OSPF routes to be imported to the BGP routing tables on PEs must carry an attribute (the OSPF domain ID) to identify the OSPF domains. The domain ID of an OSPF process is contained in the routes generated by the process. When an OSPF route is imported to BGP, the domain ID is added to BGP VPN routes as the extended BGP community.

In cases where a VPN has multiple MCE devices attached to it and when an MCE device advertises the routes learned from BGP within the VPN, the routes may be learned by other MCE devices, thus generating route loops. To prevent route loops, you can configure route tags for different VPN instances on each MCE. HP recommends you assign the same route tag to the same VPN on all MCEs.

IS-IS

Similar to those in OSPF, IS-IS processes can be bound to VPN instances for private network routes to be exchanged between MCE and sites. An IS-IS process can be bound to only one VPN instance.

IBGP

To use IBGP to exchange private routes between an MCE and a site, you must configure IBGP peers for VPN instances on the MCE and redistribute IGP routing information from corresponding VPNs. If the MCE is connected with multiple sites in the same VPN, you can configure the MCE as an RR and configure the egress routers of the sites as clients, thus making the MCE reflect routing information between the sites. This eliminates the necessity for BGP connections between sites, reducing the number of BGP connections and simplifying network configuration.

EBGP

To use EBGP to exchange private routes between an MCE and a site, you must configure BGP peers for VPN instances on MCE and redistribute IGP routing information from corresponding VPNs. Normally, sites reside in different ASs, and EBGP is used for route exchange. In this case, the following configurations are needed.

1. Configuring to use EBGP to import IGP routes from each site

To advertise private network routes to PEs properly, IGP routes in the sites directly connected to an MCE device must be first imported to the BGP routing table of the MCE device.

2. Configuring a peer group for each VPN instance

For proper route exchange between a CE and a site, you must configure a peer group for each VPN instance and assign AS numbers for these peer groups in BGP IPv4 address family view.

3. Applying filtering policies for route filtering

To make sure that routing information is exchanged between sites and PE devices properly, filtering policies are applied to filter routes received or to be advertised.

Route exchange between an MCE and a PE

Routing information entries are bound to specific VPN instances on an MCE device, and packets of each VPN instance are forwarded between MCE and PE according to interface. As a result, VPN routing information can be transmitted by performing relatively simple configurations between MCE and PE, such as importing the VPN routing entries on MCE devices to the routing table of the routing protocol running between MCE and PEs.

The following routing protocols can be used between MCE and PE devices for routing formation exchange:

- Static route
- RIP
- OSPF
- IS-IS
- IBGP

- EBGP

For information about routing protocol configuration and route import, see *Layer 3—IP Routing Configuration Guide*.

Configuring an MCE

Configuring VPN instances

Configuring VPN instances is required in all MCE networking schemes.

By configuring VPN instances on a PE, you isolate not only VPN routes from public network routes, but also routes of a VPN from those of another VPN. This feature allows VPN instances to be used in networking scenarios besides MCE.

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.

A VPN instance only takes effect after you configure an RD for it. Before configuring an RD for a VPN instance, you can configure no parameters for the instance other than a description.

You can configure a description for a VPN instance to record its related information, such as its relationship with a certain VPN.

To create and configure a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a VPN instance and enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required
3. Configure an RD for the VPN instance.	**route-distinguisher** *route-distinguisher*	Required
4. Configure a description for the VPN instance.	**description** *text*	Optional

For easy management, set the same RD for the same VPN instance on the MCE and PE.

Associating a VPN instance with an interface

In an MPLS L3VPN application, you must associate VPN instances with the interfaces connecting the PEs.

In a tunneling application, you must associate VPN instances with the tunnel interfaces connecting the peer MCE devices or CE devices.

After creating and configuring a VPN instance, you associate the VPN instance with the interface for connecting different VPN sites.

To associate a VPN instance with an interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view.	**interface** *interface-type interface-number*	—

Step	Command	Remarks
3. Associate the current interface with the VPN instance.	**ip binding vpn-instance** *vpn-instance-name*	Required. No VPN instance is associated with an interface by default.

Ip binding vpn-instance clears the IP address of the interface on which it is configured. Be sure to reconfigure an IP address for the interface after configuring the command.

Configuring route related attributes of a VPN instance

The control process of VPN route advertisement is as follows:

- When a VPN route learned from a site gets redistributed into BGP, BGP associates it with a VPN 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 **import-extcommunity** in the VPN target.
- The VPN instance determines how to change the VPN targets attributes for routes to be advertised according to **export-extcommunity** in the VPN target.

To configure route related attributes of a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required.
3. Enter IPv4 VPN view.	**ipv4-family**	Optional.
4. Associate the current VPN instance with one or more VPN targets.	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	Required.
5. Configure the maximum number of routes for the VPN instance.	**routing-table limit** *number* { *warn-threshold* | **simply-alert** }	Optional. Not configured by default.
6. Apply an import routing policy to the current VPN instance.	**import route-policy** *route-policy*	Optional. By default, all routes permitted by the import target attribute can be redistributed into the VPN instance.
7. Apply an export routing policy to the current VPN instance.	**export route-policy** *route-policy*	Optional. By default, all VPN instance routes permitted by the export target attribute can be redistributed.

Only when BGP runs between the MCE and PE, can the VPN target attribute be advertised to the PE along with the routing information. In other cases, configuring this attribute makes no sense.

You can configure route related attributes for IPv4 VPNs in both VPN instance view and IPv4 VPN view. Those configured in IPv4 VPN view take precedence.

A single **vpn-target** command can configure up to eight VPN targets. You can configure up to 64 VPN targets for a VPN instance.

You can define the maximum number of routes for a VPN instance to support, preventing too many routes from being redistributed into the PE.

Create the routing policy you want to associate with a VPN instance; if you do not create a routing policy first, the default routing policy is used.

Configuring routing

MCE can be regarded as a networking solution for implementing service isolation by route isolation. No special configuration is required on an MCE, except that you must enable the MCE function.

After you configure MCE, disable routing loop detection to avoid route loss during route calculation and disable route redistribution between routing protocols to save system resources on the PE.

This section describes the configuration of an MCE in an MPLS L3VPN application. The configuration for route exchange between an MCE and the peer MCE or CEs in a tunneling application is the same at that for route exchange between an MCE and PE in an MPLS L3VPN application.

Configuring static routing

A CE can be connected with a site through a static route. A static route configured for a traditional CE is effective globally, which cannot solve the problem of address space overlapping among VPNs. An MCE supports binding a static route with a VPN instance, so that the static routes of different VPN instances can be isolated from each other.

To configure route exchange through static routes:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Configure a static route for each VPN instance.	**ip route-static vpn-instance** *s-vpn-instance-name&<1-6> dest-address* { *mask* | *mask-length* } { *gateway-address* [**public**] | *interface-type interface-number* [*gateway-address*] | **vpn-instance** *d-vpn-instance-name gateway-address* } [**preference** *preference-value*] [**tag** *tag-value*] [**description** *description-text*]	Required. By default, for a static route, the precedence is 60, the tag is 0, and no description is configured. Generally, to advertise the static routes of a VPN instance to a certain site or PE, use the first command; to advertise static routes globally, use the second.
	ip route-static *dest-address* { *mask* | *mask-length* } { *gateway-address* | *interface-type interface-number* [*gateway-address*] | **vpn-instance** *d-vpn-instance-name gateway-address* } [**preference** *preference-value*] [**tag** *tag-value*] [**description** *description-text*]	
3. Configure the default precedence for static routes.	**ip route-static default-preference** *default-preference-value*	Optional. 60 by default

Configuring RIP

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. By configuring RIP-to-VPN bindings on a CE, you allow routes of different VPNs to be exchanged between the CE and the sites through different RIP processes, ensuring the separation and security of VPN routes.

To configure route exchange through RIP:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter RIP view and bind the RIP process to a VPN instance.	**rip** [*process-id*] **vpn-instance** *vpn-instance-name*	Required.
3. Redistribute routes of the routing protocol processes connecting the MCE with the PE and the sites respectively.	**import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | **route-policy** *route-policy-name* | **tag** *tag*] *	Required. By default, no route of any other protocol is redistributed into RIP. • If the RIP process is for an MCE and a site, the routes to be redistributed should be from the routing protocol process used for advertising the routes of the VPN between the MCE and PE. • If the RIP process is for an MCE and a PE, the routes to be redistributed should be from the routing protocol process used for advertising the routes of the VPN between the MCE and the sites.
4. Configure the default cost value for the redistributed routes.	**default cost** *value*	Optional. 0 by default.

After you configure a RIP process, you must enable RIP. The configuration procedure is the same as the common RIP configuration. For more information about RIP, see *Layer 3—IP Routing Configuration Guide*.

Configuring OSPF

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.

On the MCE, you not only must configure the binding between a VPN instance, an OSPF process, and the router ID, but also must redistribute the VPN routes locally maintained to the routing table of the OSPF process.

To configure route exchange through OSPF:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create an OSPF process for a VPN instance and enter OSPF view.	**ospf** [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*] *	Required.

Step	Command	Remarks
3. Enable the multi-VPN-instance function of OSPF.	**vpn-instance-capability simple**	Required. Disabled by default
4. Configure the OSPF domain ID.	**domain-id** *domain-id* [**secondary**]	Optional. 0 by default. Configure this command for an OSPF process for route exchange with a site. This configuration is Required. on the MCE. On the VPN site, perform the common OSPF configuration.
5. Redistribute routes of the routing protocol processes connecting the MCE with the PE and the sites respectively.	**import-route** *protocol* [*process-id* | **allow-ibgp**] [**cost** *cost* | **type** *type* | **tag** *tag* | **route-policy** *route-policy-name*] *	Required. By default, no route of any other protocol is redistributed into OSPF. • If the OSPF process is for an MCE and a site, the routes to be redistributed should be from the routing protocol process used for advertising the routes of the VPN between the MCE and PE. • If the OSPF process is for an MCE and a PE, the routes to be redistributed should be from the routing protocol process used for advertising the routes of the VPN between the MCE and the sites.
6. Configure a filtering policy to filter the redistributed routes.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*]]	Optional. By default, OSPF does not filter redistributed routes.
7. Configure the default values of the parameters for the redistributed routes, including the route cost, count limit, tag, and type.	**default** { **cost** *cost* | **limit** *limit* | **tag** *tag* | **type** *type* } *	Optional. This configuration is used for connecting the PE. By default, the default values of the parameters are as follows: • Cost: 1 • Number of external routes redistributed in a specified period: 1000 • Tag: 1 • Type: 2.

Step	Command	Remarks
8. Configure the type codes of OSPF extended community attributes.	**ext-community-type** { **domain-id** *type-code1* | **router-id** *type-code2* | **route-type** *type-code3* }	Optional. The defaults are as follows: 0x0005 for Domain ID, 0x0107 for Router ID, and 0x0306 for Route Type. Configure this command on a PE.

An OSPF process that is bound with a VPN instance does not use the public network router ID configured in system view. Therefore, you must configure a router ID when starting the OSPF process. All OSPF processes for the same VPN must be configured with the same OSPF domain ID to ensure correct route advertisement.

An OSPF process can belong to only one VPN instance, but one VPN instance can use multiple OSPF processes to advertise the VPN routes.

After you configure an OSPF instance, you must enable OSPF. The configuration procedure is the same as the common OSPF configuration. For OSPF configuration details, see *Layer 3—IP Routing Configuration Guide*.

Configuring IS-IS

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 route exchange through IS-IS:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create an IS-IS process for a VPN instance and enter IS-IS view.	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	Required.

Step	Command	Remarks
3. Redistribute routes of the routing protocol processes connecting the MCE with the PE and the sites respectively.	**import-route** { **isis** [*process-id*] | **ospf** [*process-id*] | **rip** [*process-id*] | **bgp** [**allow-ibgp**] | **direct** | **static** } [**cost** *cost* | **cost-type** { **external** | **internal** } | [**level-1** | **level-1-2** | **level-2**] | **route-policy** *route-policy-name* | **tag** *tag*] *	Optional. By default, IS-IS does not redistribute routes of any other protocol. If you do not specify the route level in the command, the command redistributes routes to the level-2 routing table by default. • If the IS-IS process is for an MCE and a site, the routes to be redistributed should be from the routing protocol process used for advertising the routes of the VPN between the MCE and PE. • If the IS-IS process is for an MCE and a PE, the routes to be redistributed should be from the routing protocol process used for advertising the routes of the VPN between the MCE and the sites.
4. Configure a filtering policy to filter the redistributed routes.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* | **route-policy** *route-policy-name* } **export** [**isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **bgp** | **direct** | **static**]	Optional. This configuration is used for connecting the PE. By default, IS-IS does not filter the redistributed routes.

After you configure an IS-IS process for a VPN instance, you must enable the IS-IS process. For more information about IS-IS, see *Layer 3—IP Routing Configuration Guide*.

Configuring IBGP

To use IBGP for route exchange between an MCE and a PE or the egress router of a site, configure the MCE and the PE or the egress router of the site as IBGP peers.

If the MCE is connected with multiple sites in the same VPN, configure the MCE and the egress routers of the sites as a cluster and configure the MCE as the route reflector of the cluster.

To configure the MCE to use IBGP for route exchange with a PE or a site:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Enter BGP-VPN instance view.	**ipv4-family vpn-instance** *vpn-instance-name*	Required.
4. Configure the PE or the egress router of the site as the IBGP peer of the VPN.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.

Step	Command	Remarks
5. Configure the egress router of the site as a client of the route reflector.	**peer** { *group-name* | *ip-address* } **reflect-client**	Optional. By default, no route reflector or client is configured.
6. Enable route reflection between clients.	**reflect between-clients**	Optional. By default, route reflection between clients is enabled.
7. Specify a cluster ID for the route reflector.	**reflector cluster-id** *cluster-id*	Optional. By default, each route reflector uses its own Router ID as the cluster ID.
8. Configure the filtering policy for routes to be advertised.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct** | **isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	Optional. By default, routes to be advertised are not filtered.
9. Configure the filtering policy for routes received.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	Optional. By default, routes received are not filtered.

On a PE or the egress router of a site, you must configure the MCE as the IBGP peer. If the routing protocol of the site is not IBGP, redistribute the routes of the routing protocol to IBGP. The configuration required is the same as that of common IBGP. For more information about IBGP, see *Layer 3—IP Routing Configuration Guide*.

Configuring EBGP

If EBGP is used for exchanging routing information between an MCE and a PE, you must configure the MCE and the PE as the peer of each other in BGP-VPN instance view at both sides, redistribute the VPN routes of the VPN sites to the MCE, and then advertise the VPN routes to the PE.

If EBGP is used for exchanging routing information between a MCE and a VPN site, you must configure a BGP peer for each VPN instance respectively and redistribute the IGP routes of each VPN instance into the routing table of the corresponding BGP peer on the CE. As different sites are normally located in different ASs, EBGP is used for route advertisement.

1. Redistribute IGP routes of each VPN site to EBGP

To advertise VPN routes to the PE correctly, the MCE needs to redistribute the IGP routes of each directly connected VPN site to the BGP routing table of the MCE.

2. Configure a peer group for each VPN instance

For the MCE to exchange routing information with each site correctly, you must configure a peer group for each VPN instance and specify an AS number for each peer group in BGP-VPN instance view.

3. Filter routes by using filter policies

For the MCE to send the routing information to the sites and PE correctly, you also must configure a filter policy to filter the received/sent routes.

To configure route exchange through EBGP:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Enter BGP-VPN instance view.	**ipv4-family vpn-instance** *vpn-instance-name*	Required.
4. Add the VPN instance to a peer group.	**peer** { *group-name* | *ip-address* } [**as-number** *as-number*]	
5. Configure the device to allow routing loops, that is, to allow the local AS number to appear in the AS_PATH attribute of a received route, and you can also configure the maximum number of times that such case is allowed to appear.	**peer** { *group-name* | *ip-address* } **allow-as-loop** [*number*]	Required. Use either command.
6. Redistribute routes of the routing protocol processes connecting the MCE with the PE and the sites respectively.	**import-route** *protocol* [*process-id* | **all-processes**] [**med** *med-value* | **route-policy** *route-policy-name*] *	Required. By default, BGP does not redistribute routes of any other protocol. • If the BGP process is for an MCE and a site, the routes to be redistributed should be from the routing protocol process used for advertising the routes of the VPN between the MCE and PE. • If the BGP process is for an MCE and a PE, the routes to be redistributed should be from the routing protocol process used for advertising the routes of the VPN between the MCE and the sites.
7. Configure a filtering policy to filter the routes to be advertised.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct | isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	Optional. By default, BGP does not filter the routes to be advertised.
8. Configure a filtering policy to filter the received routes.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	Optional. By default, BGP does not filter the received routes.

Normally, BGP checks routing loops by examining AS numbers. If EBGP is used between the MCE and a site, when the MCE advertises its routing information with its AS number to the site and then receives routing update information from the site, the route update message carries the AS number of the MCE, making the MCE unable to receive this route update message. In this case, to enable the MCE to receive route updates normally, configure the MCE to allow routing loops.

In standard BGP/OSPF route redistribution, when a route is redistributed into OSPF from BGP on the MCE, the route's original OSPF attribute cannot be restored, making the route unable to be distinguished from routes redistributed from other domains. To distinguish routes of different OSPF domains, you must enable a route to carry the OSPF domain ID when the route is redistributed from OSPF into BGP on the peer PE. Thus, the domain ID of an OSPF process is carried in a route generated by the process. When the OSPF route is redistributed into BGP, the domain ID is added to the BGP VPN route and is transmitted over the network as the extended community attribute of BGP.

After you configure a BGP VPN instance, the BGP route exchange in the VPN instance is the same as the common BGP's. For more information about BGP configuration, see *Layer 3—IP Routing Configuration Guide*.

Displaying and maintaining MCE

Resetting BGP connections

When BGP configuration changes, you can use the soft reset function or reset BGP connections to make new configurations take effect. Soft reset requires that BGP peers have route refreshment capability (supporting Route-Refresh messages).

Task	Command	Remarks
Soft reset the BGP connections in a specified VPN instance.	**refresh bgp vpn-instance** *vpn-instance-name* { *ip-address* | **all** | **external** | **group** *group-name* } { **export** | **import** }	Available in user view
Reset BGP connections of a VPN instance.	**reset bgp vpn-instance** *vpn-instance-name* { *as-number* | *ip-address* | **all** | **external** | **group** *group-name* }	Available in user view

Displaying and maintaining MCE

Task	Command	Remarks
Display information about the routing table associated with a VPN instance.	**display ip routing-table vpn-instance** *vpn-instance-name* [**verbose**] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about a specified or all VPN instances.	**display ip vpn-instance** [**instance-name** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about the FIB of a VPN instance.	**display fib vpn-instance** *vpn-instance-name* [**acl** *acl-number* | **ip-prefix** *ip-prefix-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about the FIB of a VPN instance that matches the specified destination IP address.	**display fib vpn-instance** *vpn-instance-name ip-address* [*mask* | *mask-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

Task	Command	Remarks
Display information about a specified or all BGP VPNv4 peer group.	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **group** [*group-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about BGP VPNv4 routes injected into a specified or all VPN instances.	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **network** [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display BGP VPNv4 AS path information.	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **paths** [*as-regular-expression* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]	Available in any view
Display information about BGP VPNv4 peers.	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **peer** [*group-name* **log-info** | *ip-address* { **log-info** | **verbose** } | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display the BGP VPNv4 routing information of a specified VPN instance.	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **routing-table** [[*network-address* [{ *mask* | *mask-length* } [**longer-prefixes**]] | **as-path-acl** *as-path-acl-number* | **cidr** | **community** [*aa:nn*]&<1-13> [**no-advertise** | **no-export** | **no-export-subconfed**] * [**whole-match**] | **community-list** { *basic-community-list-number* [**whole-match**] | *adv-community-list-number* }&<1-16> | **dampened** | **dampening parameter** | **different-origin-as** | **flap-info** [*network-address* [{ *mask* | *mask-length* } [**longer-match**]] | **as-path-acl** *as-path-acl-number*] | **peer** *ip-address* { **advertised-routes** | **received-routes** } | **statistic**] [**|** { **begin** | **exclude** | **include** } *regular-expression*] | [**flap-info**] **regular-expression** *as-regular-expression*]	Available in any view
Clear the route flap dampening information of a VPN instance.	**reset bgp vpn-instance** *vpn-instance-name* **dampening** [*network-address* [*mask* | *mask-length*]	Available in user view
Clear route flap history information about a BGP peer of a VPN instance.	**reset bgp vpn-instance** *vpn-instance-name* *ip-address* **flap-info** **reset bgp vpn-instance** *vpn-instance-name* **flap-info** [*ip-address* [*mask* | *mask-length*] | **as-path-acl** *as-path-acl-number* | **regexp** *as-path-regexp*]	Available in user view

For commands to display information about a routing table, see *Layer 3—IP Routing Command Reference*.

MCE configuration examples

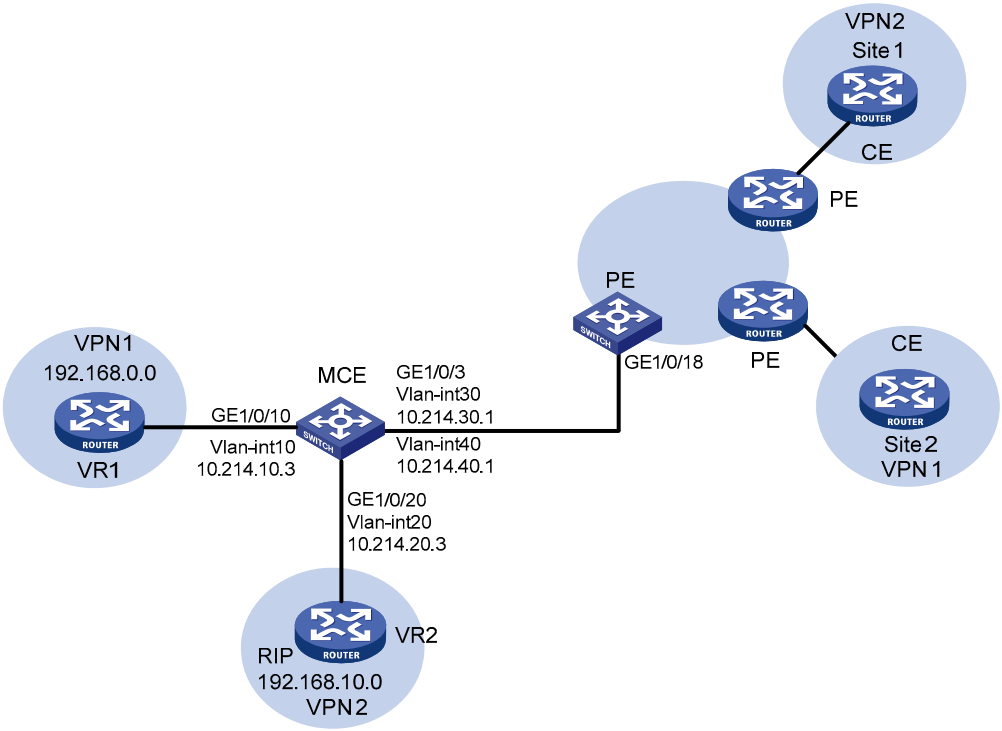
Using OSPF to advertise VPN routes to the PE

Network requirements

See Figure 6.

- An MCE device connects to VPN1 (with the address range being 192.168.0.0/16) through VLAN-interface 10 (with the IP address being 10.214.10.3) and connects to VPN2 (with the address range being 192.168.10.0/24) through VLAN-interface 20 (with the IP address being 10.214.20.3). VPN2 has RIP enabled.
- The MCE device connects to the PE through VLAN-interface 30 and VLAN-interface 40, whose IP addresses are 192.168.30.1/30 and 192.168.40.1/30, respectively.
- The MCE device isolates routes of VPN1 from those of VPN2 and advertises all VPN routes to the PE device using OSPF.

Figure 6 Network diagram for MCE configuration (A)



Procedure

Assume the system name of the MCE device is "MCE", the names of the egress router of VPN1 and VPN2 are "VR1" and "VR2", and the system name of the PE device is "PE".

- Configure VPN instances

Configure two instances VPN1 and VPN2 on the MCE device, with the RD values of the two VPN instances being 10:1 and 20:1.

```
<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
```

Create VLAN 10, add GigabitEthernet 1/0/10 to VLAN 10, and create VLAN-interface 10.

```
[MCE-vpn-instance-vpn2] quit
[MCE] vlan 10
[MCE-vlan10] port GigabitEthernet 1/0/10
[MCE-vlan10] quit
[MCE] interface Vlan-interface 10
```

Bind VLAN-interface 10 to VPN1, and configure IP address 10.214.10.3/24 for VLAN-interface 10.

```
[MCE-Vlan-interface10] ip binding vpn-instance vpn1
[MCE-Vlan-interface10] ip address 10.214.10.3 24
```

Create VLAN 20, add GigabitEthernet 1/0/20 to VLAN 20, create VLAN-interface 20, bind VLAN-interface 20 to VPN2, and configure IP address 10.214.20.3/24 for VLAN-interface 20.

```
[MCE-Vlan-interface10] quit
[MCE] vlan 20
[MCE-vlan20] port GigabitEthernet 1/0/20
[MCE-vlan20] quit
[MCE] interface Vlan-interface 20
[MCE-Vlan-interface20] ip binding vpn-instance vpn2
[MCE-Vlan-interface20] ip address 10.214.20.3 24
[MCE-Vlan-interface20] quit
```

Create VLAN 30, VLAN 40 and the corresponding VLAN interfaces. Then bind VLAN-interface 30 to VPN 1, and VLAN-interface 40 to VPN 2, and configure IP addresses of the VLAN interfaces.

```
[MCE] vlan 30
[MCE-vlan30] quit
[MCE] interface Vlan-interface 30
[MCE-Vlan-interface30] ip binding vpn-instance vpn1
[MCE-Vlan-interface30] ip address 10.214.30.1 30
[MCE-Vlan-interface30] quit
[MCE] vlan 40
[MCE-vlan40] quit
[MCE] interface Vlan-interface 40
[MCE-Vlan-interface40] ip binding vpn-instance vpn2
[MCE-Vlan-interface40] ip address 10.214.40.1 30
[MCE-Vlan-interface40] quit
```

- Configure the routing protocol running between MCE and a site

MCE is directly connected to VPN1, which has no routing protocol enabled. You can configure the device to use static routes between MCE and a site.

Configuration on VR1: Assume VR1 is an A5800 switch, configure IP address 10.214.10.2/24 for the interface connecting to MCE and IP address 192.168.0.1/24 for the interface connecting to VPN1. The operation of adding a port to a VLAN and configuring IP address for a VLAN-interface is omitted here.

Configure a default route on VR1, specifying the next hop address to 10.214.10.3.

```
<VR1> system-view
```

```
[VR1] ip route-static 0.0.0.0 0.0.0.0 10.214.10.3
```

Define a static route on MCE, specify the next hop address 10.214.10.2 for packets destined for the network segment 192.168.0.0, and bind this route to VPN1.

```
[MCE-Vlan-interface10] quit
[MCE] ip route-static vpn-instance vpn1 192.168.0.0 16 10.214.10.2
```

Display the information about the routes of VPN1 maintained on MCE.

```
[MCE] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 5        Routes : 5

Destination/Mask    Proto   Pre  Cost         NextHop         Interface

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
10.214.10.0/24      Direct 0      0            10.214.10.3     Vlan10
10.214.10.3/32      Direct 0      0            127.0.0.1       InLoop0
192.168.0.0/16     Static 60      0            10.214.10.2     Vlan10
```

The output shows that a static route has been specified for VPN1.

On VR2, configure IP address 10.214.20.2/24 for the interface connecting to MCE (configuration procedures omitted), enable RIP and advertise the network segments 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 10.0.0.0
```

RIP is running within VPN2, so you can configure RIP on MCE and involve the RIP on MCE in the routing computation in the site to update the routing information automatically. Create RIP process 20, disable automatic route summarization, redistribute routes from OSPF process 20, and bind the RIP process to VPN2.

```
[MCE] rip 20 vpn-instance vpn2
```

Advertise the network segment 10.214.20.0 and 10.214.40.0.

```
[MCE-rip-20] network 10.0.0.0
[MCE-rip-20] undo summary
[MCE-rip-20] import-route ospf
```

Display the information about the routes of VPN2 on MCE.

```
[MCE-rip-20] display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 5        Routes : 5

Destination/Mask    Proto   Pre  Cost         NextHop         Interface

10.214.20.0/24      Direct 0    0            10.214.20.3     Vlan20
10.214.20.3/32      Direct 0    0            127.0.0.1       InLoop0
10.214.40.0/30      Direct 0    0            10.214.40.1     Vlan40
10.214.40.1/32      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
```

```
192.168.10.0/24     RIP    100  1           10.214.20.2     Vlan20
```

The output shows that MCE has obtained the routes of VPN2 through RIP, and maintains these routes in a routing table different from the routing table for routing information of VPN1 to the network segment 192.168.0.0, isolating the routes of VPN1 from the routes of VPN2.

- Configure the routing protocol running between the MCE and a PE

MCE uses GigabitEthernet 1/0/3 to connect to GigabitEthernet 1/0/18 of PE. Configure the two ports to be trunk ports and permit tagged packets of VLAN 30 and VLAN 40.

```
[MCE-rip-20] quit
[MCE] interface GigabitEthernet 1/0/3
[MCE-GigabitEthernet1/0/3] port link-type trunk
[MCE-GigabitEthernet1/0/3] port trunk permit vlan 30 40
```

Configure GigabitEthernet 1/0/18 of PE.

```
<PE> system-view
[PE] interface GigabitEthernet 1/0/18
[PE-GigabitEthernet1/0/18] port link-type trunk
[PE-GigabitEthernet1/0/18] port trunk permit vlan 30 40
```

Configure IP addresses 10.214.30.2 and 10.214.40.2 for VLAN-interface 30 and VLAN-interface 40 of PE respectively. Then bind VLAN-interface 30 to VPN 1, and VLAN-interface 40 to VPN 2. The configuration procedures are omitted here.

Configure Loopback0 of MCE and CE to specify the router ID for MCE and PE respectively. The IP addresses for Loopback0 of MCE and CE are 101.101.10.1 and 100.100.10.1 respectively. Configuration procedures are omitted here.

Create OSPF process 10 on MCE, bind the process to VPN1, and set the OSPF domain ID to 10, and enable OSPF multi-instance.

```
[MCE-GigabitGigabitEthernet1/0/3] quit
[MCE] ospf 10 router-id 101.101.10.1 vpn-instance vpn1
[MCE-ospf-10] domain 10
[MCE-ospf-10] vpn-instance-capability simple
```

Advertise the network segment 10.214.30.0 within Area0, and import static routes of VPN1.

```
[MCE-ospf-10] area 0
[MCE-ospf-10-area-0.0.0.0] network 10.214.30.0 0.0.0.255
[MCE-ospf-10-area-0.0.0.0] quit
[MCE-ospf-10] import-route static
```

Create OSPF process 10 on PE, bind the process to VPN1, set the OSPF domain ID to 10, enable OSPF multi-instance, and advertise the network segment 10.214.30.0 within Area0.

```
[PE-GigabitEthernet1/0/18] quit
[PE] ospf 10 router-id 100.100.10.1 vpn-instance vpn1
[PE-ospf-10] domain-id 10
[PE-ospf-10] vpn-instance-capability simple
[PE-ospf-10] area 0
[PE-ospf-10-area-0.0.0.0] network 10.214.30.0 0.0.0.255
```

Display the information about the routes of VPN1 on PE.

```
[PE-ospf-10-area-0.0.0.0] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 6        Routes : 6
```

```
Destination/Mask    Proto   Pre  Cost         NextHop         Interface

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
10.214.30.0/24      Direct  0    0            10.214.30.1     Vlan30
10.214.30.2/32      Direct  0    0            127.0.0.1       InLoop0
100.100.10.1/32     Direct  0    0            127.0.0.1       InLoop0
192.168.0.0/16      O_ASE   150  1            10.214.30.1     Vlan30
```

The output shows that the static routes of VPN1 have been imported to the OSPF routing table between MCE and PE.

Take similar procedures to create OSPF process 20 and import the routing information of VPN2. Make sure to redistribute RIP routes rather than static routes to the OSPF routing table of MCE. Perform the following command to verify your configuration.

```
<PE> display ip routing-table vpn-instance vpn2
display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 6        Routes : 6

Destination/Mask    Proto   Pre  Cost         NextHop         Interface

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
10.214.40.0/24      Direct  0    0            10.214.40.1     Vlan40
10.214.40.2/32      Direct  0    0            127.0.0.1       InLoop0
200.200.20.1/32     Direct  0    0            127.0.0.1       InLoop0
192.168.10.0/24     O_ASE   150  1            10.214.40.1     Vlan40
```

The output shows that the routing information of VPN1 and VPN2 can be advertised to PE properly.

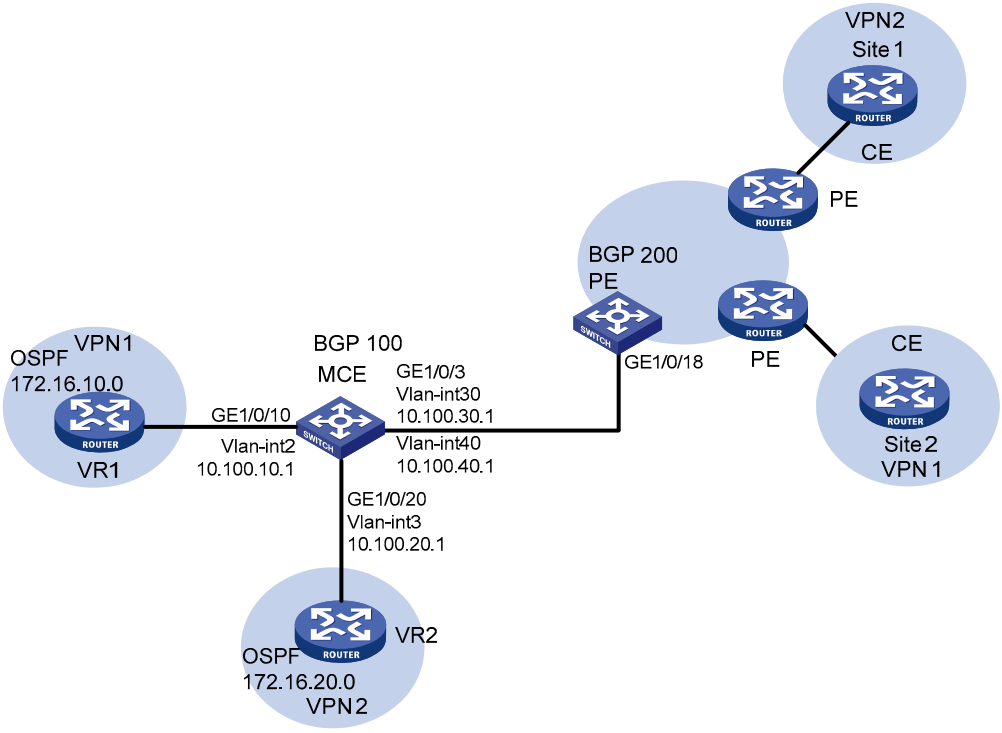
Using BGP to advertise VPN routes to the PE

Network requirements

See Figure 7.

- An A5800 switch functions as an MCE. Advertise the VPN routes of site 1 and site 2 to the PE, so that VPNs at both ends of the MPLS backbone network can communicate with each other properly.
- OSPF runs within both site 1 and site 2, and EBGP runs between MCE and PE.

Figure 7 Network diagram for MCE configuration (B)



Procedure

- Configure VPN instances

Configure two instances VPN1 and VPN2 on the MCE device, with the RD values of the two VPN instances being 10:1 and 20:1. Configure the VPN target values of the two VPN instances as 10:1 and 20:1 for both the import and export extended community attribute list.

```
<MCE> system-view
[MCE] ip vpn-instance vpn1
[MCE-vpn-instance-vpn1] route-distinguisher 10:1
[MCE-vpn-instance-vpn1] vpn-target 10:1 both
[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 both
```

Create VLAN 2, add GigabitEthernet 1/0/10 to VLAN 2, and create VLAN-interface 2.

```
[MCE-vpn-instance-vpn2] quit
[MCE] vlan 2
[MCE-vlan2] port GigabitEthernet 1/0/10
[MCE-vlan2] quit
[MCE] interface Vlan-interface 2
```

Bind VLAN-interface 2 to VPN1, and configure IP address 10.214.10.3/24 for VLAN-interface 2.

```
[MCE-Vlan-interface2] ip binding vpn-instance vpn1
[MCE-Vlan-interface2] ip address 10.214.10.3 24
```

Create VLAN 3, add GigabitEthernet 1/0/20 to VLAN 3, create VLAN-interface 3, bind VLAN-interface 3 to VPN2, and configure IP address 10.214.20.3/24 for VLAN-interface 3.

```
[MCE-Vlan-interface10] quit
[MCE] vlan 3
[MCE-vlan3] port GigabitEthernet 1/0/20
[MCE-vlan3] quit
[MCE] interface Vlan-interface 3
[MCE-Vlan-interface3] ip binding vpn-instance vpn2
[MCE-Vlan-interface3] ip address 10.214.20.3 24
[MCE-Vlan-interface3] quit
```

Create VLAN 30, VLAN 40 and the corresponding VLAN interfaces. Then bind VLAN 30 to VPN 1, and VLAN 40 to VPN 2, and configure IP addresses of the VLAN interfaces.

```
[MCE] vlan 30
[MCE-vlan30] quit
[MCE] interface Vlan-interface 30
[MCE-Vlan-interface30] ip binding vpn-instance vpn1
[MCE-Vlan-interface30] ip address 10.214.30.1 30
[MCE-Vlan-interface30] quit
[MCE] vlan 40
[MCE-vlan40] quit
[MCE] interface Vlan-interface 40
[MCE-Vlan-interface40] ip binding vpn-instance vpn2
[MCE-Vlan-interface40] ip address 10.214.40.1 30
[MCE-Vlan-interface40] quit
```

- Configure the routing protocol running between MCE and a site

The procedure of enabling OSPF in the two VPN instances and advertising the network segments is the same as that in normal OSPF and is omitted.

Create OSPF process 10 for MCE whose router ID is 10.10.10.1, bind the process to VPN1. Redistribute BGP routes from VPN1, enable OSPF multi-instance, and advertise the network segment 10.100.10.0.

```
<MCE> system-view
[MCE] ospf 10 router-id 10.10.10.1 vpn-instance vpn1
[MCE-ospf-10] vpn-instance capability simple
[MCE-ospf-10] import-route bgp
[MCE-ospf-10] area 0
[MCE-ospf-10-area-0.0.0.0] network 10.100.10.0 0.0.0.255
```

Display the information about the routes of VPN1.

```
[MCE-ospf-10-area-0.0.0.0] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

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
10.100.10.0/24      Direct 0    0            10.100.10.1     Vlan2
10.100.10.1/32      Direct 0    0            127.0.0.1       InLoop0
172.16.10.0/24      OSPF   10   1            10.100.10.2     Vlan2
```

The output shows that MCE has obtained the routing information of VPN1 through OSPF process 10.

Create OSPF process 20 for MCE whose router ID is 10.10.20.1, bind the process to VPN2. Redistribute BGP routes from VPN2, enable OSPF multi-instance, and advertise the network segment 10.100.20.0. The procedure of configuring OSPF process 20 is similar to that of configuring OSPF process 10. Perform the following command to verify your configuration.

```
[MCE] display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

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
10.100.20.0/24      Direct 0    0            10.100.20.1     Vlan3
10.100.20.1/32      Direct 0    0            127.0.0.1       InLoop0
172.16.20.0/24      OSPF   10   1            10.100.20.2     Vlan3
```

- Configure the routing protocol running between MCE and PE

The procedure of connecting MCE to PE through trunk ports is similar to that in "Using OSPF to advertise VPN routes to the PE" and is omitted here.

Create BGP process 10 for MCE.

```
[MCE] bgp 100
[MCE-bgp]
```

Enter IPv4 address family view in VPN1.

```
[MCE-bgp] ipv4-family vpn-instance vpn1
[MCE-bgp-vpn1]
```

Configure PE as an EBGP peer and import the routing information of OSPF process 10 (assuming that the address of the interface bound to VPN1 is 10.100.30.3 and the ID of the BGP process is 200).

```
[MCE-bgp-vpn1] peer 10.100.30.3 as-number 200
[MCE-BGP-vpn1] import-route ospf 10
```

Create BGP process 200 on the PE, and configure MCE as an EBGP peer.

```
<PE> system-view
[PE] bgp 200
[PE-bgp] ipv4-family vpn-instance vpn1
[PE-bgp-vpn1] peer 10.100.30.1 as-number 100
```

Display the information about the routes of VPN1 on PE.

```
<PE-bgp-vpn1> display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

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
10.100.30.0/24      Direct 0    0            10.100.10.3     Vlan2
10.100.30.3/32      Direct 0    0            127.0.0.1       InLoop0
```

```
172.16.10.0/24      BGP    255  2           10.100.10.2     Vlan2
```

For VPN2, perform the configurations similar to those on MCE and PE to import the OSPF routing information of VPN2 to the EBGP routing table.

Perform the following command to verify your configuration:

```
<PE> display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

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
10.100.40.0/24      Direct 0    0            10.100.20.3     Vlan3
10.100.40.3/32      Direct 0    0            127.0.0.1       InLoop0
172.16.20.0/24      BGP    255  2            10.100.20.2     Vlan3
```

The output shows that MCE has imported the OSPF routing information of VPN1 and VPN2 to the EBGP routing table of PE properly.

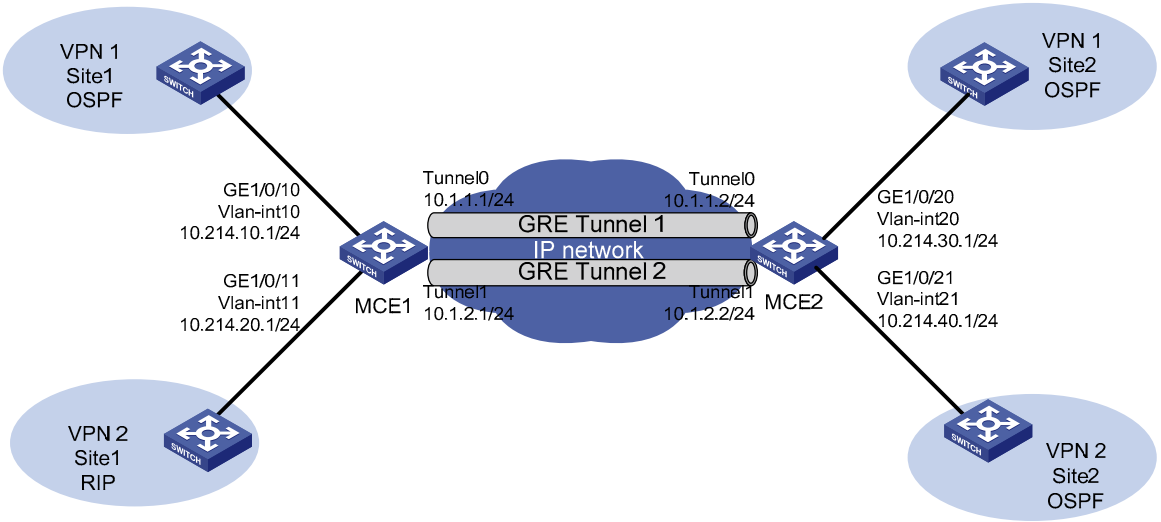
Using tunnels to advertise VPN routes

Network requirements

As shown in Figure 8, MCE 1 and MCE 2 communicate with each other through GRE tunnels, and both are connected with sites of VPN 1 and VPN 2. The sites of VPN 1 use routing protocol OSPF and reside in the backbone area, that is, area 0.The two sites of VPN 2 use RIP and OSPF respectively, and the OSPF area 0 is used.

Configure MCE 1 and MCE 2 to correctly advertise routing information of the two VPNs.

Figure 8 Network diagram for using MCE to advertise VPN routes through tunnels



Configuration considerations

As shown in Figure 8, because a GRE tunnel is configured for each VPN to transmit data and routing information of the VPN, you can create a VPN instance for each VPN and bind the VPN instances to specific interfaces (the tunnel interfaces and interfaces connected to the VPN sites). In this way the current network is simplified into two separate topologies, as shown in Figure 9 and Figure 10. Thus, MCEs advertise routes of different VPNs through different paths.

For VPN 1, advertise interface addresses on the two MCEs in area 0, making the entire VPN a single OSPF domain. For VPN 2, advertise interface addresses for RIP and OSPF calculations, and in addition, redistribute OSPF routes to RIP and RIP routes to OSPF on MCE 1.

Figure 9 Network topology of VPN 1 with the MCEs

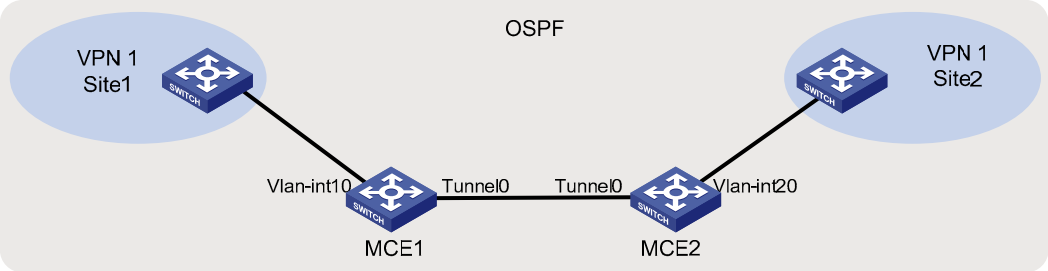
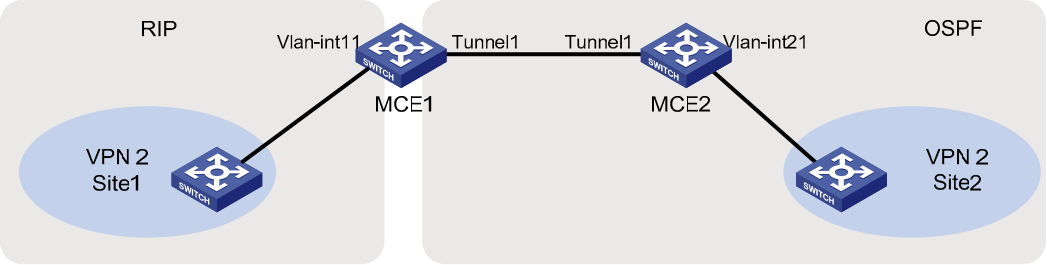


Figure 10 Network topology of VPN 2 with the MCEs



Procedure

1. Configure tunnels

Establish two GRE tunnels between MCE 1 and MCE 2. One tunnel transmits traffic for VPN 1, and the other for VPN 2. The configuration procedure is omitted. For related information, see *Layer 3—IP Services Configuration Guide*.

2. Configure VPN instances.

- Configurations on MCE 1

Create VPN instance **vpn1** for VPN 1, and configure an RD for it.

```
<MCE1> system-view
[MCE1] ip vpn-instance vpn1
[MCE1-vpn-instance-vpn1] route-distinguisher 1:2
[MCE1-vpn-instance-vpn1] quit
```

Create VPN instance **vpn2** for VPN 2, and configure an RD for it.

```
[MCE1] ip vpn-instance vpn2
[MCE1-vpn-instance-vpn2] route-distinguisher 1:3
[MCE1-vpn-instance-vpn2] quit
```

Bind VLAN-interface 10 and Tunnel 0 with VPN instance **vpn1**, and configure IP addresses for the VLAN interface and tunnel interface.

```
[MCE1] vlan 10
[MCE1-vlan10] port gigabitethernet 1/0/10
[MCE1-vlan10] quit
[MCE1] interface vlan-interface 10
[MCE1-Vlan-interface10] ip binding vpn-instance vpn1
[MCE1-Vlan-interface10] ip address 10.214.10.1 24
[MCE1-Vlan-interface10] quit
[MCE1] interface tunnel 0
[MCE1-Tunnel0] ip binding vpn-instance vpn1
[MCE1-Tunnel0] ip address 10.1.1.1 24
```

Bind VLAN-interface 11 and Tunnel 1 with VPN instance **vpn2**, and configure IP addresses for the VLAN interface and tunnel interface.

```
[MCE1] vlan 11
[MCE1-vlan11] port gigabitethernet 1/0/11
[MCE1-vlan11] quit
[MCE1] interface vlan-interface 11
[MCE1-Vlan-interface11] ip binding vpn-instance vpn2
[MCE1-Vlan-interface11] ip address 10.214.20.1 24
[MCE1-Vlan-interface11] quit
[MCE1] interface tunnel 1
[MCE1-Tunnel1] ip binding vpn-instance vpn2
[MCE1-Tunnel1] ip address 10.1.2.1 24
[MCE1-Tunnel1] quit
```

- Configurations on MCE 2

Create VPN instance **vpn1** for VPN 1, and configure the same RD as that configured on MCE 1 for the VPN instance.

```
<MCE2> system-view
[MCE2] ip vpn-instance vpn1
[MCE2-vpn-instance-vpn1] route-distinguisher 1:2
[MCE2-vpn-instance-vpn1] quit
```

Create VPN instance **vpn2** for VPN 2, and configure the same RD as that configured on MCE 1 for the VPN instance.

```
[MCE2] ip vpn-instance vpn2
[MCE2-vpn-instance-vpn2] route-distinguisher 1:3
[MCE2-vpn-instance-vpn2] quit
```

Bind VLAN-interface 20 and Tunnel 0 with VPN instance **vpn1**, and configure IP addresses for the VLAN interface and tunnel interface.

```
[MCE2] vlan 20
[MCE2-vlan20] port gigabitethernet 1/0/20
[MCE2-vlan20] quit
[MCE2] interface vlan-interface 20
[MCE2-Vlan-interface20] ip binding vpn-instance vpn1
[MCE2-Vlan-interface20] ip address 10.214.30.1 24
[MCE2-Vlan-interface20] quit
[MCE2] interface tunnel 0
```

```
[MCE2-Tunnel0] ip binding vpn-instance vpn1
[MCE2-Tunnel0] ip address 10.1.1.2 24
```

Bind VLAN-interface 21 and Tunnel 1 with VPN instance **vpn2**, and configure IP addresses for the VLAN interface and tunnel interface.

```
[MCE2] vlan 21
[MCE2-vlan21] port gigabitethernet 1/0/21
[MCE2-vlan21] quit
[MCE2] interface vlan-interface 21
[MCE2-Vlan-interface21] ip binding vpn-instance vpn2
[MCE2-Vlan-interface21] ip address 10.214.40.1 24
[MCE2-Vlan-interface21] quit
[MCE2] interface tunnel 1
[MCE2-Tunnel1] ip binding vpn-instance vpn2
[MCE2-Tunnel1] ip address 10.1.2.2 24
[MCE2-Tunnel1] quit
```

3. Configure routing protocols.

- Advertise routes of VPN 1.

On MCE 1, configure OSPF process 1 for VPN instance **vpn1**, and configure OSPF to support MCE. Be sure to configure the same OSPF area as that configured at site 1 of VPN 1, area 0 in this example.

```
[MCE1] ospf 1 vpn-instance vpn1 router-id 192.168.1.1
[MCE1-ospf-1] vpn-instance-capability simple
[MCE1-ospf-1] area 0
[MCE1-ospf-1-area-0.0.0.0]
```

Advertise the addresses of interfaces VLAN-interface 10 and Tunnel 0 on MCE 1.

```
[MCE1-ospf-1-area-0.0.0.0] network 10.214.10.1 0.0.0.255
[MCE1-ospf-1-area-0.0.0.0] network 10.1.1.1 0.0.0.255
```

On MCE 2, configure OSPF process 1 for VPN instance **vpn1**, and configure OSPF to support MCE.

```
[MCE2] ospf 1 vpn-instance vpn1 router-id 172.16.1.1
[MCE2-ospf-1] vpn-instance-capability simple
[MCE2-ospf-1] area 0
[MCE2-ospf-1-area-0.0.0.0]
```

Advertise addresses of interfaces VLAN-interface 20 and Tunnel 0 on MCE 2.

```
[MCE2-ospf-1-area-0.0.0.0] network 10.214.30.1 0.0.0.255
[MCE2-ospf-1-area-0.0.0.0] network 10.1.1.2 0.0.0.255
```

- Advertise routes of VPN 2.

On MCE 1, configure OSPF process 2 for VPN instance **vpn2**, and configure OSPF to support MCE. Be sure to configure the same OSPF area as that configured at site 2 of VPN 2, area 0 in this example.

```
[MCE1] ospf 2 vpn-instance vpn2 router-id 192.168.2.1
[MCE1-ospf-2] vpn-instance-capability simple
[MCE1-ospf-2] area 0
[MCE1-ospf-2-area-0.0.0.0]
```

Advertise the address of tunnel interface Tunnel 1.

```
[MCE1-ospf-2-area-0.0.0.0] network 10.1.2.1 0.0.0.255
```

Configure RIP process 1 for VPN instance **vpn2**.

```
[MCE1] rip 1 vpn-instance vpn2
```

```
[MCE1-rip-1]
```

Advertise the IP address of VLAN-interface 11.

```
[MCE1-rip-1] network 10.214.20.1
```

Redistribute routes learned by OSPF process 2 to RIP process 1.

```
[MCE1-rip-1] import-route ospf 2
[MCE1-rip-1] quit
```

Redistribute routes learned by RIP process 1 to OSPF process 2.

```
[MCE1] ospf 2
[MCE1-ospf-2] import-route rip 1
```

On MCE 2, configure OSPF process 2 for VPN instance **vpn2**, and configure OSPF to support MCE. Be sure to configure the same OSPF area as that configured at site 2 of VPN 2, area 0 in this example.

```
[MCE2] ospf 2 vpn-instance vpn2 router-id 172.16.2.1
[MCE2-ospf-2] vpn-instance-capability simple
[MCE2-ospf-2] area 0
[MCE2-ospf-2-area-0.0.0.0]
```

Advertise the addresses of interfaces VLAN-interface 21 and Tunnel 1 on MCE 2.

```
[MCE2-ospf-2-area-0.0.0.0] network 10.214.40.1 0.0.0.255
[MCE2-ospf-2-area-0.0.0.0] network 10.1.2.2 0.0.0.255
```

Configuring IPv6 MCE

In an IPv6 MPLS L3 VPN, an IPv6 MCE advertises IPv6 routing information between the VPN and the connected PE and forwards IPv6 packets. An IPv6 MCE operates in the same way as an IPv4 MCE. For more information, see "MCE configuration."

Configuring VPN instances

By configuring VPN instances on a PE, you isolate not only VPN routes from public network routes, but also routes of a VPN from those of another VPN. This feature allows VPN instances to be used in network scenarios besides MPLS L3VPNs.

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.

A VPN instance only takes effect after you configure an RD for it.

You can configure a description for a VPN instance to record its related information, such as its relationship with a certain VPN.

To create and configure a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a VPN instance and enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required
3. Configure an RD for the VPN instance.	**route-distinguisher** *route-distinguisher*	Required
4. Configure a description for the VPN instance.	**description** *text*	Optional

Associating a VPN instance with an interface

After creating and configuring a VPN instance, you must associate the VPN instance with the interface for connecting the CE. Any LDP-capable interface can be associated with a VPN instance. For information about LDP-capable interfaces, see "MPLS basics configuration."

To associate a VPN instance with an interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view.	**interface** *interface-type interface-number*	—
3. Associate a VPN instance with the interface.	**ip binding vpn-instance** *vpn-instance-name*	Required. No VPN instance is associated with an interface by default.

Ip binding vpn-instance command clears the IPv6 address of the interface on which it is configured. Be sure to re-configure an IPv6 address for the interface after configuring the command.

Configuring route related attributes for a VPN instance

The control process of VPN route advertisement is as follows:

- When a VPN route learned from a CE gets redistributed into BGP, BGP associates it with a VPN 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 **import-extcommunity** in the VPN target.
- The VPN instance determines how to change the VPN targets attributes for routes to be advertised according to **export-extcommunity** in the VPN target.

To configure route related attributes for a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	—
3. Enter IPv6 VPN view.	**ipv6-family**	Optional.
4. Configure VPN targets.	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	Required.
5. Set the maximum number of routes supported.	**routing-table limit** *number* { *warn-threshold* | **simply-alert** }	Optional.
6. Apply an import routing policy.	**import route-policy** *route-policy*	Optional. By default, all routes matching the import target attribute are accepted.
7. Apply an export routing policy.	**export route-policy** *route-policy*	Optional. By default, routes to be advertised are not filtered.

Route related attributes configured in VPN instance view are applicable to both IPv4 VPNs and IPv6 VPNs.

You can configure route related attributes for IPv6 VPNs in both VPN instance view and IPv6 VPN view. Those configured in IPv6 VPN view take precedence.

A single **vpn-target** command can configure up to eight VPN targets. You can configure up to 64 VPN targets for a VPN instance.

You can define the maximum number of routes for a VPN instance to support, preventing too many routes from being redistributed into the PE. The maximum number of routes supported by a PE varies by device.

Create a routing policy before associating it with a VPN instance so that the device can filter the routes to be received and advertised.

Configuring routing

To enable VPN route distribution, bind the interfaces between the MCE and the PE to the IPv6 instances, perform routing configuration, and redistribute VPN routing information to the routing protocol running between the MCE and PE.

The following operations are performed on the MCE. The PE is configured in the same way as it is configured in a basic IPv6 MPLS L3VPN. For information about how to configure the PE, see "MPLS L3VPN configuration."

Configuring IPv6 static routing

To configure static routing between MCE and PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Configure static routes for an IPv6 VPN instance.	**ipv6 route-static** *ipv6-address prefix-length* { *interface-type interface-number* [*next-hop-address*] | *next-hop-address* | **vpn-instance** *d-vpn-instance-name nexthop-address* } [**preference** *preference-value*]	Required. User either command.
	ipv6 route-static vpn-instance *s-vpn-instance-name*&<1-6> *ipv6-address prefix-length* { *interface-type interface-number* [*next-hop-address*] | *nexthop-address* [**public**] | **vpn-instance** *d-vpn-instance-name nexthop-address* } [**preference** *preference-value*]	
3. Configure the default precedence for static routes.	**ipv6 route-static default-preference** *default-preference-value*	Optional. 60 by default

Configuring RIPng

To configure RIPng between MCE and PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a RIPng process for an IPv6 VPN instance and enter RIPng view.	**ripng** [*process-id*] **vpn-instance** *vpn-instance-name*	Required.
3. Redistribute the VPN routes.	**import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | **route-policy** *route-policy-name*] *	Required. By default, no route of any other routing protocol is redistributed into RIPng.
4. Configure the default cost value for the redistributed routes.	**default cost** *value*	Optional. 0 by default
5. Return to system view.	**quit**	—
6. Enter interface view.	**interface** *interface-type interface-number*	—

Step	Command	Remarks
7. Enable the RIPng process on the interface	**ripng** *process-id* **enable**	Required. Disabled by default.

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

Configuring OSPFv3

To configure OSPFv3 between MCE and PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create an OSPFv3 process for an IPv6 VPN instance and enter OSPFv3 view.	**ospfv3** [*process-id*] **vpn-instance** *vpn-instance-name*	Required.
3. Set the router ID.	**router-id** *router-id*	Required.
4. Redistribute the VPN routes.	**import-route** *protocol* [*process-id* | **allow-ibgp**] [**cost** *value* | **route-policy** *route-policy-name* | **type** *type*] *	Required. By default, no route of any other routing protocol is redistributed into OSPFv3.
5. Configure a filtering policy to filter the redistributed routes.	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **export** [**bgp4+** | **direct** | **isisv6** *process-id* | **ospfv3** *process-id* | **ripng** *process-id* | **static**]	Optional. By default, redistributed routes are not filtered.
6. Return to system view.	**quit**	—
7. Enter interface view.	**interface** *interface-type interface-number*	—
8. Enable the OSPFv3 process on the interface.	**ospfv3** *process-id* **area** *area-id* [**instance** *instance-id*]	Required. Disabled by default.

For more information about OSPFv3, see *Layer 3—IP Routing Configuration Guide*.

Configuring IPv6 IS-IS

To configure IPv6 IS-IS between MCE and PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create an IS-IS process for an IPv6 VPN instance and enter IS-IS view.	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	Required.
3. Configure a network entity title.	**network-entity** *net*	Required. Not configured by default.
4. Enable the IPv6 capacity for the IS-IS process.	**ipv6 enable**	Required. Disabled by default.

Step	Command	Remarks
5. Redistribute the VPN routes.	**ipv6 import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | [**level-1** | **level-1-2** | **level-2**] | **route-policy** *route-policy-name* | **tag** *tag*] *	Optional. By default, IS-IS does not redistribute routes of 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.
6. Configure a filtering policy to filter the redistributed routes.	**ipv6 filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* | **route-policy** *route-policy-name* } **export** [*protocol* [*process-id*]]	Optional. By default, IPv6 IS-IS does not filter redistributed routes.
7. Return to system view.	**quit**	—
8. Enter interface view.	**interface** *interface-type interface-number*	—
9. Enable IPv6 for the IS-IS process on the interface.	**isis ipv6 enable** [*process-id*]	Required. Disabled by default.

For more information about IPv6 IS-IS, see *Layer 3—IP Routing Configuration Guide*.

Configuring eBGP

To configure eBGP between MCE and PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Enter IPv6 BGP-VPN instance view.	**ipv6-family vpn-instance** *vpn-instance-name*	Required.
4. Configure the PE as the eBGP peer.	**peer** *ipv6-address* **as-number** *as-number*	Required.
5. Redistribute the VPN routes.	**import-route** *protocol* [*process-id* [**med** *med-value* | **route-policy** *route-policy-name*] *]	Required. By default, No route redistribution is configured.
6. Configure a filtering policy to filter the routes to be advertised.	**filter-policy** { *acl6-number* | **ipv6-prefix** *ip-prefix-name* } **export** [**direct** | **isisv6** *process-id* | **ripng** *process-id* | **static**]	Optional. By default, BGP does not filter the routes to be advertised.
7. Configure a filtering policy to filter the received routes.	**filter-policy** { *acl6-number* | **ipv6-prefix** *ip-prefix-name* } **import**	Optional. By default, BGP does not filter the received routes.

IPv6 BGP runs within a VPN in the same way as it runs within a public network. For more information about IPv6 BGP, see *Layer 3—IP Routing Configuration Guide*.

Displaying and maintaining IPv6 MCE

Resetting BGP connections

When BGP configuration changes, you can use the soft reset function or reset BGP connections to make new configurations take effect. Soft reset requires that BGP peers have route refreshment capability (supporting Route-Refresh messages).

To hard reset or soft reset BGP connections:

Task	Command	Remarks
Soft reset the IPv6 BGP connections in a VPN instance	**refresh bgp ipv6 vpn-instance** *vpn-instance-name* { *ipv6-address* | **all** | **external** } { **export** | **import** }	Available in user view
Hard reset the IPv6 BGP connections of a VPN instance	**reset bgp ipv6 vpn-instance** *vpn-instance-name* { *as-number* | *ipv6-address* | **all** | **external** }	Available in user view

Displaying information about IPv6 MCE

Task	Command	Remarks
Display information about a specific or all VPN instances.	**display ip vpn-instance** [**instance-name** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about the IPv6 FIB of a VPN instance.	**display ipv6 fib vpn-instance** *vpn-instance-name* [**acl6** *acl6-number* | **ipv6-prefix** *ipv6-prefix-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display a VPN instance's FIB entries that match the specified destination IPv6 address.	**display ipv6 fib vpn-instance** *vpn-instance-name ipv6-address* [*prefix-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about BGP VPNv6 peers established between PEs.	**display bgp vpnv6 all peer** [*ipv4-address* **verbose** | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about IPv6 BGP peers established between the PE and CE in a VPN instance.	**display bgp vpnv6 vpn-instance** *vpn-instance-name* **peer** [*ipv6-address* **verbose** | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display all BGP VPNv6 routing information.	**display bgp vpnv6 all routing-table** [*network-address prefix-length* [**longer-prefixes**] | **peer** *ip-address* { **advertised-routes** | **received-routes** } [**statistic**] | **statistic**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

Task	Command	Remarks
Display the BGP VPNv6 routing information of a specified RD.	**display bgp vpnv6 route-distinguisher** *route-distinguisher* **routing-table** [*network-address prefix-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display the BGP VPNv6 routing information of a VPN instance.	**display bgp vpnv6 vpn-instance** *vpn-instance-name* **routing-table** [*network-address prefix-length* [**longer-prefixes**] | **peer** *ipv6-address* { **advertised-routes** | **received-routes** }] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

For commands that display information about a routing table, see *Layer 3—IP Routing Command Reference*.

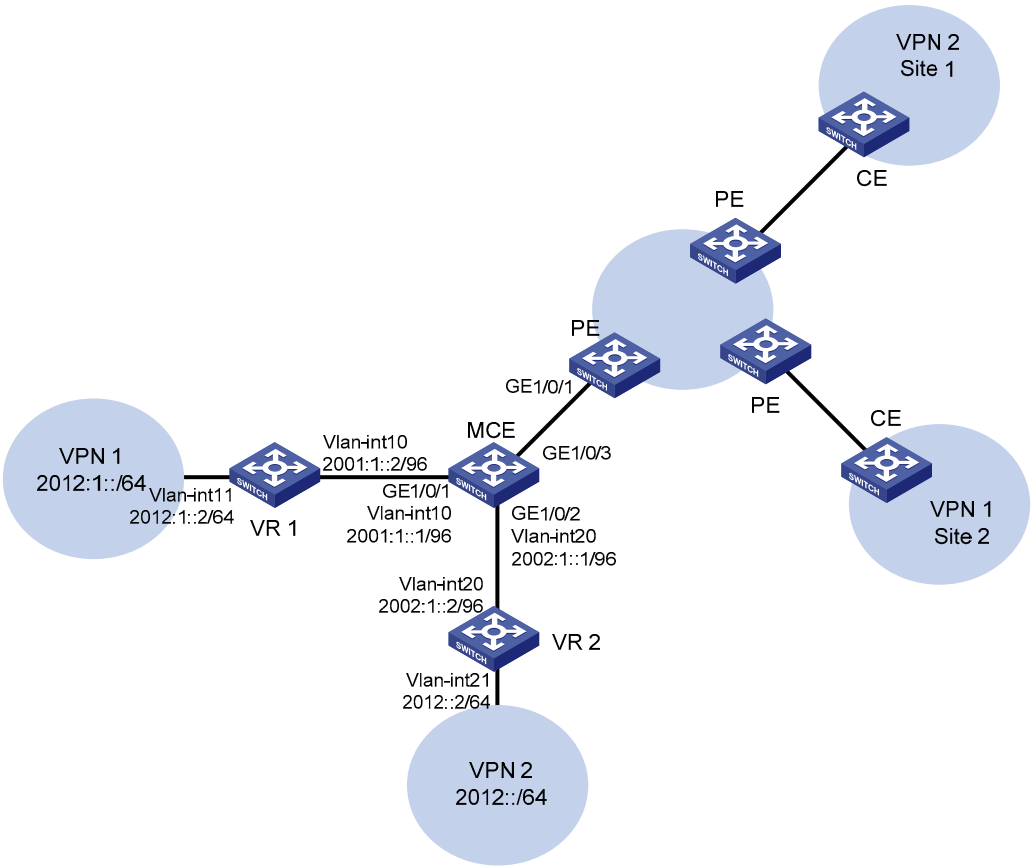
IPv6 MCE configuration examples

Using IPv6 IS-IS to advertise VPN routes to the PE

Network requirements

- The MCE device is connected to VPN 1 through VLAN-interface 10, and to VPN 2 through VLAN-interface 20. VPN 1's network segment is 2012:1::/64. VPN 2's network segment is 2012::/64 and RIPng is used in VPN 2.
- The MCE device separates routes from different VPNs and advertises VPN routes to the PE through IPv6 IS-IS.

Figure 11 Network diagram for MCE configuration



Procedure

Assume that the system name of the MCE device is MCE, those of the edge devices of VPN 1 and VPN 2 are VR 1 and VR 2 respectively, and that of the PE device is PE.

1. Configure the VPN instances on MCE

On MCE, configure VPN instances **VPN1** and **VPN2**, and specify a RD and VPN targets for each VPN instance, using the same value for the RD and VPN targets.

```
<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
```

Create VLAN 10, add port GigabitEthernet 1/0/1 to VLAN 10, and create VLAN-interface 10.

```
[MCE] vlan 10
[MCE-vlan10] port Gigabitethernet 1/0/1
[MCE-vlan10] quit
[MCE] interface vlan-interface 10
```

Bind VLAN-interface 10 with instance VPN1, and configure the IPv6 address of the VLAN interface as 2001:1::1/64.

```
[MCE-Vlan-interface10] ip binding vpn-instance vpn1
[MCE-Vlan-interface10] ipv6 address 2001:1::1 64
[MCE-Vlan-interface10] quit
```

Similarly, configure VLAN 20, add port GigabitEthernet 1/0/2 to VLAN 20, bind VLAN-interface 20 with instance VPN2, and assign an IPv6 address to VLAN-interface 20.

```
[MCE] vlan 20
[MCE-vlan20] port Gigabitethernet 1/0/2
[MCE-vlan20] quit
[MCE] interface Vlan-interface 20
[MCE-Vlan-interface20] ip binding vpn-instance vpn2
[MCE-Vlan-interface20] ipv6 address 2002:1::1 64
[MCE-Vlan-interface20] quit
```

2. Configure routing between MCE and sites

MCE is connected with VPN 1 directly, and no routing protocol is enabled in VPN 1. Therefore, you can configure static routes.

On VR 1, configure the IP address of the interface connected to MCE as 2001:1::2/64, and that of the interface connected to VPN 1 as 2012:1::2/64. Configurations for adding a port to the VLAN and specifying the interface IP address are omitted.

On VR 1, configure a default route, specifying the next hop as 2001:1::1.

```
<VR1> system-view
[VR1] ipv6 route-static :: 0 2001:1::1
```

On MCE, configure a static route to 2012:1::/64, specifying the next hop as 2001:1::2 and binding the static route with instance VPN1.

```
[MCE] ipv6 route-static vpn-instance vpn1 2012:1:: 64 vpn-instance vpn1 2001:1::2
```

Run RIPng in VPN 2. Configure RIPng process 20 for VPN instance VPN2 on MCE, so that MCE can learn the routes of VPN 2 and add them to the routing table of VPN instance VPN 2.

Configure RIPng process 20, binding it with instance VPN2.

```
[MCE] ripng 20 vpn-instance vpn2
```

Advertise network segment 2002:1::/64 through RIPng.

```
[MCE] interface vlan-interface 20
[MCE-Vlan-interface20] ripng 20 enable
[MCE-Vlan-interface20] quit
```

On VR 2, assign IPv6 address 2002:1::2/64 to the interface connected to MCE (omitted).

Configure RIPng, and advertise network segments 2012::/64 and 2002:1::/64.

```
<VR2> system-view
[VR2] ripng 20
[VR2-rip-20] quit
[VR2] interface vlan-interface 20
[VR2-Vlan-interface20] ripng 20 enable
[VR2-Vlan-interface20] quit
[VR2] interface vlan-interface 21
[VR2-Vlan-interface21] ripng 20 enable
```

```
[VR2-Vlan-interface21] quit
```

On MCE, display the routing information maintained for VPN instances VPN1 and VPN2.

```
[MCE] display ipv6 routing-table vpn-instance vpn1
Routing Table :
         Destinations : 5        Routes : 5

Destination: ::1/128                                    Protocol  : Direct
NextHop    : ::1                                        Preference: 0
Interface  : InLoop0                                    Cost      : 0

Destination: 2001:1::/64                                Protocol  : Direct
NextHop    : 2001:1::1                                  Preference: 0
Interface  : Vlan10                                     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  : Vlan10                                     Cost      : 0

Destination: FE80::/10                                  Protocol  : Direct
NextHop    : ::                                         Preference: 0
Interface  : NULL0                                      Cost      : 0

[MCE] display ipv6 routing-table vpn-instance vpn2
Routing Table :
         Destinations : 5        Routes : 6

Destination: ::1/128                                    Protocol  : Direct
NextHop    : ::1                                        Preference: 0
Interface  : InLoop0                                    Cost      : 0

Destination: 2002:1::/64                                Protocol  : Direct
NextHop    : 2002:1::1                                  Preference: 0
Interface  : GE0/1/1                                   Cost      : 0

Destination: 2002:1::1/128                              Protocol  : Direct
NextHop    : ::1                                        Preference: 0
Interface  : InLoop0                                    Cost      : 0

Destination: 2012::/64                                  Protocol  : RIPng
NextHop    : FE80::200:5EFF:FE01:1C03                   Preference: 100
Interface  : Vlan20                                     Cost      : 1

Destination: FE80::/10                                  Protocol  : Direct
NextHop    : ::                                         Preference: 0
```

```
Interface  : NULL0                                    Cost      : 0
```

The output shows that MCE has learned the private routes of VPN 2. 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 route exchange between MCE and the PE

On MCE, configure the port connected to the PE as a trunk port, and configure it to permit packets of VLAN 10 and VLAN 20 to pass with VLAN tags.

```
[MCE] interface gigabitethernet 1/0/3
[MCE-GigabitEthernet1/0/3] port link-type trunk
[MCE-GigabitEthernet1/0/3] port trunk permit vlan 10 20
[MCE-GigabitEthernet1/0/3] quit
```

On the PE, configure the port connected to MCE as a trunk port, and configure it to permit packets of VLAN 10 and VLAN 20 to pass with VLAN tags.

```
<PE> system-view
[PE] interface gigabitethernet 1/0/1
[PE-GigabitEthernet1/0/1] port link-type trunk
[PE-GigabitEthernet1/0/1] port trunk permit vlan 10 20
[PE-GigabitEthernet1/0/1] quit
```

On the PE, assign IPv6 addresses 2001:1::4 and 2002:1::4 to VLAN-interface 10 and VLAN-interface 20 respectively. The configuration procedure is omitted.

On MCE, start IPv6 ISIS process 10, bind the process with instance VPN1, and redistribute the static route of VPN 1.

```
[MCE] isis 10 vpn-instance vpn1
[MCE-isis-10] ipv6 enable
[MCE-isis-10] network 47.0001.0001.0002.00
[MCE-isis-10] import-route static
[MCE-isis-10] quit
```

Enable IPv6 ISIS on interface VLAN-interface 10.

```
[MCE] interface vlan-interface 10
[MCE-Vlan-interface10] isis ipv6 enable 10
[MCE-Vlan-interface10] quit
```

On the PE, start IPv6 ISIS process 10, and bind the process with instance VPN1.

```
[PE] isis 10 vpn-instance vpn1
[PE-isis-10] ipv6 enable
[PE-isis-10] network 47.0001.0001.0003.00
[PE-isis-10] quit
```

Enable IPv6 ISIS on interface VLAN-interface 10.

```
[PE] interface vlan-interface 10
[PE-Vlan-interface10] isis ipv6 enable 10
[PE-Vlan-interface10] quit
```

On the PE, display the routing information of VPN1.

```
[PE] display ipv6 routing-table vpn-instance vpn1
Routing Table :
         Destinations : 5        Routes : 5

Destination: ::1/128                                  Protocol  : Direct
```

```
NextHop    : ::1                                    Preference: 0
Interface  : InLoop0                                Cost      : 0

Destination: 2001:1::/64                            Protocol  : Direct
NextHop    : 2001:1::4                              Preference: 0
Interface  : Vlan10                                 Cost      : 0

Destination: 2001:1::4/128                          Protocol  : Direct
NextHop    : ::1                                    Preference: 0
Interface  : InLoop0                                Cost      : 0

Destination: 2012:1::/64                            Protocol  : ISISv6
NextHop    : FE80::200:5EFF:FE01:1C05               Preference: 15
Interface  : Vlan10                                 Cost      : 10

Destination: FE80::/10                              Protocol  : Direct
NextHop    : ::                                     Preference: 0
Interface  : NULL0                                  Cost      : 0
```

The output shows that the PE has learned the private route of VPN 1 through IPv6 ISIS.

The procedure for configuring IPv6 ISIS process 20 between MCE and PE and redistributing VPN 2's RIPng routing information is similar to the configuration procedure above. The difference is that MCE redistributed RIPng routes to IPv6 ISIS. The following output shows that the PE has learned the private route of VPN 2 through IPv6 ISIS.

```
[PE] display ipv6 routing-table vpn-instance vpn2
Routing Table :
         Destinations : 6        Routes : 6

Destination: ::1/128                                Protocol  : Direct
NextHop    : ::1                                    Preference: 0
Interface  : InLoop0                                Cost      : 0

Destination: 2002:1::/64                            Protocol  : Direct
NextHop    : 2002:1::4                              Preference: 0
Interface  : vlan20                                 Cost      : 0

Destination: 2002:1::4/128                          Protocol  : Direct
NextHop    : ::1                                    Preference: 0
Interface  : InLoop0                                Cost      : 0

Destination: 2012::/64                              Protocol  : ISISv6
NextHop    : FE80::200:5EFF:FE01:1C06               Preference: 15
Interface  : Vlan20                                 Cost      : 10

Destination: FE80::/10                              Protocol  : Direct
NextHop    : ::                                     Preference: 0
Interface  : NULL0                                  Cost      : 0
```

Now, the routing information of the two VPNs has been added into the routing tables on the PE.

Configuring MPLS basics

- The term router in this document refers to both routers and Layer 3 switches.
- For information about VPN, see "Configuring MPLS L2VPN" and "Configuring MPLS L3VPN".
- The Layer 3 Ethernet interface refers to the Ethernet port that can perform IP routing and inter-VLAN routing. You can set an Ethernet port as a Layer 3 Ethernet interface by using the **port link-mode route** command (see *Layer 2—LAN Switching Configuration Guide*).
- The A5820X switch series does not support MPLS.

MPLS is an IP backbone technology. It introduces connection-oriented label switching into connectionless IP networks, and seamlessly integrates the flexibility of IP routing and the simplicity of Layer 2 switching.

MPLS is widely used on large-scale networks for it features the following advantages:

- On an MPLS network, devices forward packets according to short- and fixed-length labels, instead of doing Layer 3 header analysis and complicated routing table lookup independently. This is a highly effective and fast data transmission method on backbone networks.
- Residing between the link layer and the network layer, MPLS can work on various link layer protocols (for example, PPP, ATM, frame relay, and Ethernet), provide connection-oriented services for various network layer protocols (for example, IPv4, IPv6, and IPX), and work with mainstream network technologies.
- As MPLS is connection-oriented and supports label stack, it is used in various services, such as VPN, traffic engineering, and QoS.

Basic concepts of MPLS

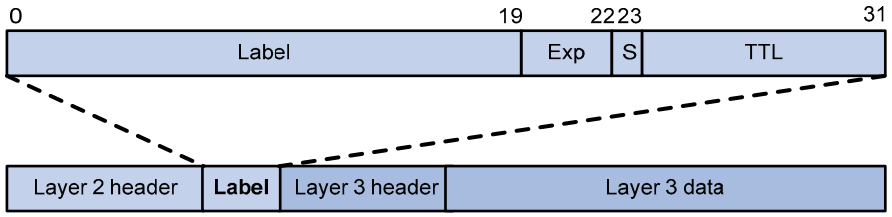
FEC

As a forwarding technology based on classification, MPLS groups packets with the same characteristics (such as packets with the same destination or service class) into a class, called a FEC. Packets of the same FEC are handled in the same way on an MPLS network. The device supports classifying FECs according to the network layer destination addresses of packets.

Label

A label is a short, fixed length identifier for identifying a single FEC. A label is locally significant and must be locally unique.

Figure 12 Format of a label



As shown in Figure 12, a label is encapsulated between the Layer 2 header and Layer 3 header of a packet. A label is four bytes in length and consists of four fields:

- Label: 20 bits in length. Label value for identifying a FEC.
- Exp: Three bits in length. Reserved field, usually used for CoS.
- S: One bit in length. MPLS supports multiple levels of labels. This field is used to indicate whether a label is at the bottom of the label stack. 1 indicates that the label is at the bottom of the label stack.
- TTL: Eight bits in length. Like the homonymous IP header field, it is used to prevent loops.

LSR

An LSR is a fundamental component on an MPLS network. LSRs support label distribution and label swapping.

LER

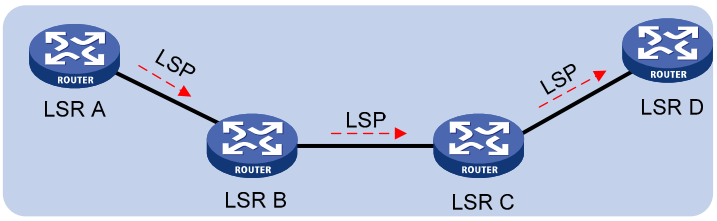
An LER resides at the edge of an MPLS network and is connected with another network.

LSP

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

An LSP is a unidirectional path from the ingress of an MPLS network to the egress. On an LSP, in the packet transfer direction, two neighboring LSRs are called the upstream LSR and downstream LSR respectively. In Figure 13, LSR B is the downstream LSR of LSR A. LSR A is the upstream LSR of LSR B.

Figure 13 Diagram for an LSP



LFIB

On an MPLS network, labeled packets are forwarded according to the LFIB, which is like the FIB for IP packet forwarding on an IP network.

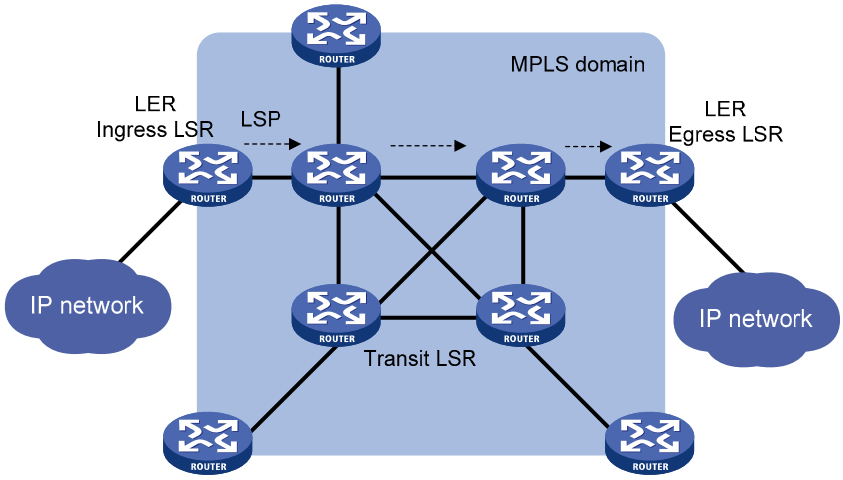
Control plane and forwarding plane

An MPLS node consists of two planes, control plane and forwarding plane.

- Control plane: Assigns labels, selects routes, establishes the LFIB, establishes and removes LSPs.
- Forwarding plane: Forwards packets according to the LFIB.

Structure of the MPLS network

Figure 14 Diagram for the MPLS network structure



As shown in Figure 14, the element of an MPLS network is LSR. LSRs in the same routing or administrative domain form an MPLS domain.

An MPLS domain consists of three types of LSRs:

- Ingress LSRs for receiving and labeling packets coming into the MPLS domain.
- Transit LSRs for forwarding packets along LSPs to their egress LERs according to the labels.
- Egress LSRs for removing labels from packets and IP forwarding the packets to their destination networks.

In a word, transit LSRs perform MPLS forwarding based on labels of packets; the ingress and egress LSRs deal with the switchover between MPLS and IP forwarding.

LSP establishment and label distribution

LSP establishment

Establishing LSPs is to bind FECs with labels on each LSR involved and notify its adjacent LSRs of the bindings, so as to establish the LFIB on each LSR. LSPs can be established through manual configuration, or be established dynamically through label distribution protocols.

1. Establishing a static LSP through manual configuration

To establish a static LSP, you must assign a label to the FEC on each LSR along the packet forwarding path. Establishment of static LSPs consumes fewer resources than dynamic LSP establishment. However, static LSPs cannot adapt to network topology changes. Therefore, static LSPs are suitable for small-scale networks with simple, stable topologies.

2. Establishing an LSP through a label distribution protocol

Label distribution protocols are MPLS signaling protocols. They can classify FECs, distribute labels, and establish and maintain LSPs. Label distribution protocols include protocols designed specifically for label distribution, such as the LDP, and protocols extended to support label distribution, such as BGP and RSVP-TE.

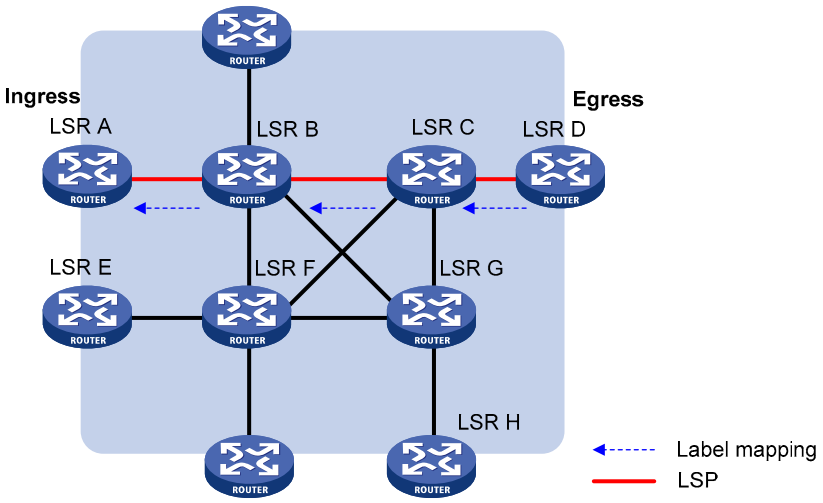
This document only discusses LDP.

In this document, the term "label distribution protocols" represents all protocols for label distribution, and the term "LDP" refers to the Label Distribution Protocol defined in RFC 5036. For detailed information on LDP, see "LDP."

As shown in Figure 15, a dynamic LSP is established in the following procedure:

A downstream LSR classifies FECs according to destination addresses, assigns a label to a FEC, and distributes the FEC-label binding to its upstream LSR, which then establishes an LFIB entry for the FEC according to the binding information. After all LSRs along the packet forwarding path establish a LFIB entry for the FEC, an LSP is established for packets of this FEC.

Figure 15 Process of dynamic LSP establishment



Label distribution and management

An LSR informs its upstream LSRs of labels assigned to FECs through label advertisement. According to the label distribution condition and order, the label advertisement mode can be DU and DoD, and the label distribution control mode can be independent or ordered.

MPLS has two label retention modes—liberal and conservative—to manage the received label bindings that are not useful at the moment.

1. Label advertisement modes

Figure 16 Label advertisement modes

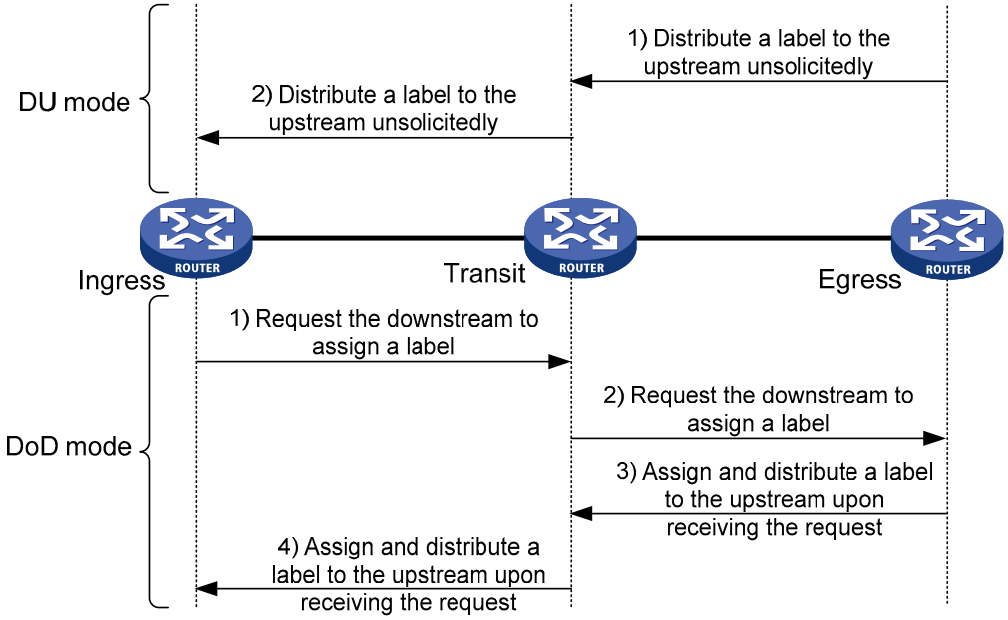


Figure 16 shows the two label advertisement modes.

- DU: In this mode, an LSR assigns a label to a FEC and then distributes the FEC-label binding to its upstream LSR unsolicitedly.
- DoD: In this mode, an LSR assigns a label to a FEC and distributes the FEC-label binding to its upstream LSR only when it receives a label request from the upstream LSR.

The A5800 series support only the DU mode.

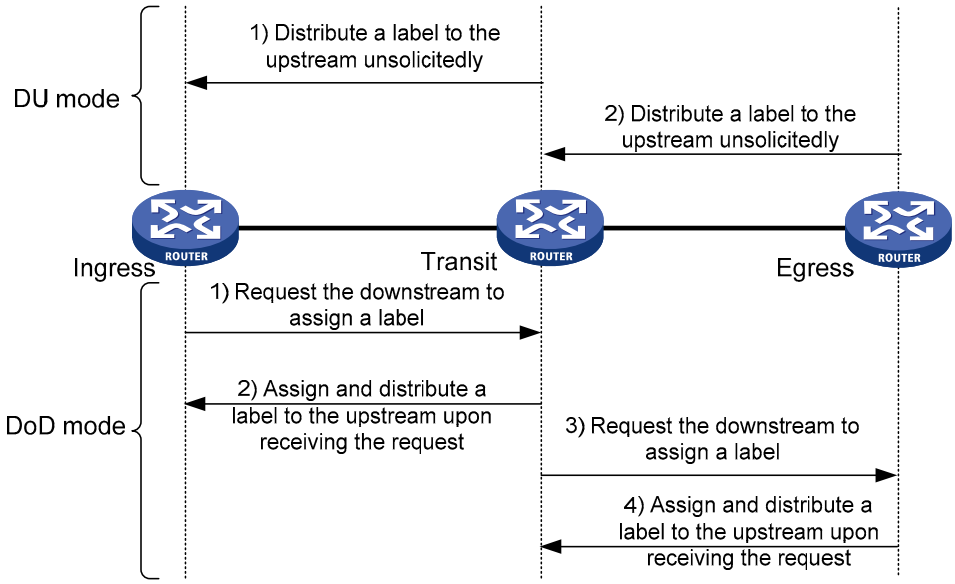
An upstream LSR and its downstream LSR for an FEC must use the same label advertisement mode so that LSPs can be established normally.

2. Label distribution control modes

Two label distribution control modes are available: independent and ordered.

- In independent mode, an LSR can distribute label bindings upstream at anytime. This means that an LSR may have distributed a label binding to its upstream LSR before it receives a binding from its downstream LSR. As shown in Figure 17, in independent label distribution control mode, if the label advertisement mode is DU, an LSR assigns labels to its upstream even if it has not obtained labels from its downstream; if the label advertisement mode is DoD, the LSR distributes a label to its upstream as long as it receives a label request from the upstream.

Figure 17 Independent label distribution control mode



- In ordered mode, an LSR distributes its label binding for a FEC upstream only when it receives a label binding for the FEC from its downstream or it is the egress of the FEC. In Figure 16, label distribution control is in ordered mode. In this case, if the label advertisement mode is DU, an LSR distributes a label upstream only when it receives a label binding for the FEC from its downstream; if the label advertisement mode is DoD, after an LSR (Transit in this example) receives a label request from its upstream (Ingress), the LSR (Transit) sends a label request to its downstream (Egress). Then, after the LSR (Transit) receives the label binding from its downstream (Egress), it distributes a label binding to the upstream (Ingress).

3. Label retention modes

Two label retention modes are available, liberal and conservative.

- In liberal mode, an LSR keeps any received label binding regardless of whether the binding is from its next hop for the FEC or not. This allows for quicker adaptation to route changes but wastes label resources as LSRs must keep extra labels.
- In conservative mode, an LSR keeps only label bindings that are from its next hops for the FECs. This allows LSRs to maintain fewer labels but makes LSRs slower in adapting to route changes.

The A5800 series support only the liberal mode.

MPLS forwarding

LFIB

The LFIB consists of the following parts:

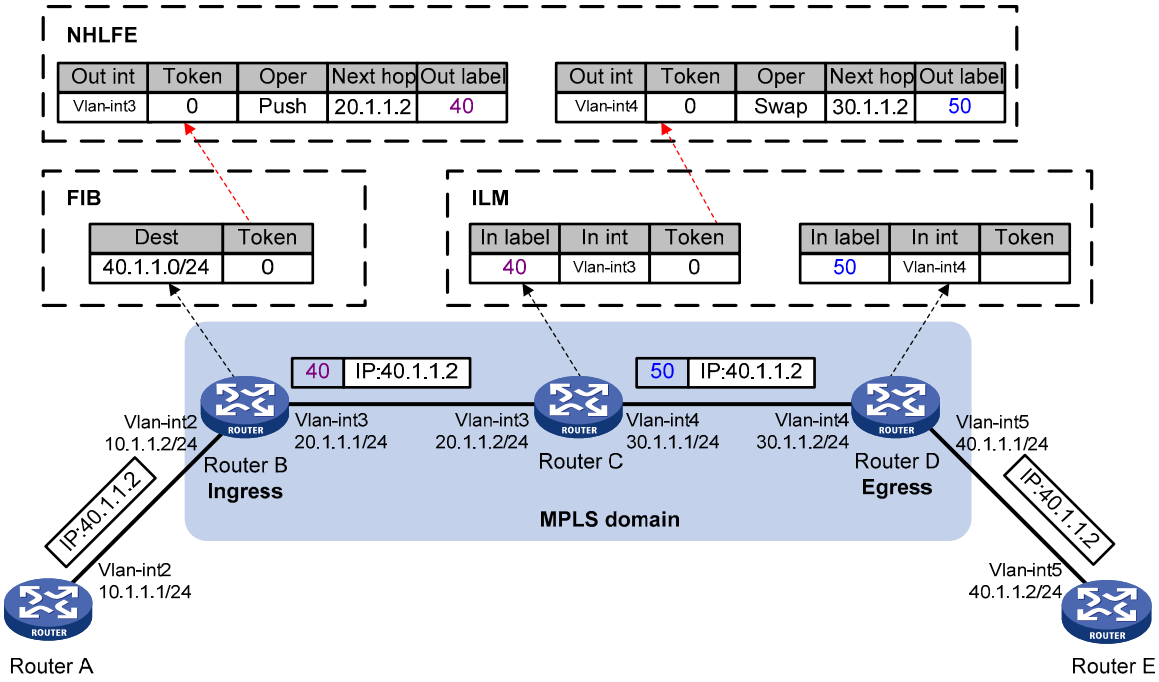
- **NHLFE**—Describes the label operation to be performed. It is used when forwarding MPLS packets.
- **FTN map**—FTN maps each FEC to a set of NHLFEs at the ingress LSR. The FTN map is used for forwarding unlabeled packets that need MPLS forwarding. When an LSR receives an unlabeled packet, it looks for the corresponding FIB entry. If the Token value of the FIB entry is not Invalid, the packet needs to be forwarded through MPLS. The LSR then looks for the corresponding NHLFE entry according to Token value to determine the label operation to be performed.

- **ILM**—ILM maps each incoming label to a set of NHLFEs. It is used when forwarding labeled packets. When an LSR receives a labeled packet, it looks for the corresponding ILM entry. If the Token value of the ILM entry is not null, the LSR looks for the corresponding NHLFE entry to determine the label operation to be performed.

FTN and ILM are associated with NHLFE through Token.

MPLS data forwarding

Figure 18 MPLS forwarding process diagram



As shown in Figure 18, in an MPLS domain, a packet is forwarded in the following procedure:

1. The ingress (Router B) receives a packet carrying no label. Router B determines the FEC of the packet according to the destination address, and searches the FIB table for the Token value. As the Token value is not Invalid, Router B looks for the corresponding NHLFE entry of the Token value. According to the NHLFE entry, Router B pushes label 40 to the packet, and then forwards the labeled packet to the next hop LSR (Router C) through the outgoing interface (Vlan-interface3).
2. Upon receiving the labeled packet, Router C looks for the ILM entry according to the label (40) to get the Token value. As the Token value is not null, Router C looks for the corresponding NHLFE entry of the Token value. According to the NHLFE entry, Router C swaps the original label with label 50, and then forwards the labeled packet to the next hop LSR (Router D) through the outgoing interface (Vlan-interface4).
3. Upon receiving the labeled packet, Router D (the egress) looks for the ILM entry according to the label (50) to get the Token value. As the Token is null, Router D removes the label from the packet. If the ILM entry records the outgoing interface, Router D forwards the packet through the outgoing interface; if no outgoing interface is recorded, router D forwards the packet according to the IP header of the packet.

PHP

In an MPLS network, when an egress node receives a labeled packet, it looks up the LFIB, pops the label of the packet, and then performs the next level label forwarding or performs IP forwarding. An egress node needs to do forwarding table lookup twice to forward a packet: looks up the LFIB twice, or looks up the LFIB once and the FIB once.

In this case, the PHP feature can pop the label at the penultimate node, relieving the egress of the label operation burden and improving the packet processing capability of the MPLS network.

PHP is configurable on the egress node. The label assigned by a PHP-capable egress node to the penultimate hop can be 0 or 3.

- A label value of 0 represents an IPv4 explicit null label and is valid only when it appears at the bottom of a label stack. An egress assigns an IPv4 explicit null label to a FEC and advertises the FEC-label binding to the upstream LSR. When forwarding an MPLS packet, the upstream LSR substitutes the label at the stack top with the explicit null label and then sends the packet to the egress. When the egress receives the packet, which carries a label of 0, it does not look up for the LFIB entry but pops the label directly and performs IPv4 forwarding.
- A label value of 3 represents an implicit null label and never appears in the label stack. When an LSR finds that it is assigned an implicit null label, it directly performs a pop operation, rather than substitutes the implicit null label for the original label at the stack top. Then, it forwards the packet to the egress. The egress thus can directly perform the next level forwarding upon receiving the packet.

LDP

The LDP protocol is used to establish LSPs dynamically. Using LDP, LSRs can map network layer routing information to data link layer switching paths.

Basic concepts of LDP

- LDP session

LDP sessions are established between LSRs based on TCP connections and used to exchange messages for label binding, label releasing, and error notification.

- LDP peer

Two LSRs with an LDP session established between them and using LDP to exchange FEC-label bindings are called LDP peers.

LDP message type

LDP messages fall into the following types:

- **Discovery messages**—declare and maintain the presence of LSRs on a network.
- **Session messages**—establish, maintain, and terminate sessions between LDP peers.
- **Advertisement messages**—create, alter, or remove FEC-label bindings.
- **Notification messages**—provide advisory information and to notify errors.

For reliable transport of LDP messages, TCP is used for LDP session messages, advertisement messages, and notification messages. UDP is used only for discovery messages.

LDP operation

LDP goes through four phases in operation:

1. Discovery

Every LSR that wants to establish LDP sessions sends Hello messages periodically to notify neighboring LSRs of its presence. In this way, LSRs can automatically discover their LDP peers. LDP provides two discovery mechanisms:

- **Basic discovery mechanism**—discover local LDP peers, or, LSRs directly connected at the link layer. In this mechanism, an LSR periodically sends LDP link hello messages to multicast address 224.0.0.2, or, all routers on the subnet, so that all LSRs directly connected at the link layer can discover this LSR.
- Extended discovery mechanism—discover remote LDP peers, or, LSRs not directly connected at the link layer. In this mechanism, an LSR periodically sends LDP targeted Hello messages to a given IP address so that the LSR with the IP address can discover the LDP peer.

2. Session establishment and maintenance

After an LSR finds a LDP peer, they start to establish a session. The LSRs goes through two steps to establish a session:

- Establishing a TCP connection between them.
- Initializing negotiation of session parameters such as the LDP version, label advertisement mode, and Keepalive interval.

After establishing a session between them, the two LDP peers send Hello messages and Keepalive messages to maintain the session.

3. LSP establishment and maintenance

LDP sends label requests and label binding messages, so as to advertise label bindings between LDP peers and thereby establish LSPs.

For the LSP establishment process, see "LSP establishment and label distribution."

4. Session termination

An LSR terminates its LDP session with an LDP peer in the following cases:

- All Hello adjacencies are deleted between the two peers

LDP peers periodically send Hello messages to indicate that they intend to keep the Hello adjacency. If an LSR does not receive any Hello message from a peer before the Hello timer expires, it deletes the Hello adjacency with this peer. An LDP session has one or more Hello adjacencies. When the last Hello adjacency for the session is deleted, the LSR sends a Notification message to terminate the LDP session.

- Loss of session connectivity

An LSR determines the integrity of an LDP session according to the LDP PDU (which carries one or more LDP messages) transmitted on the session. Before the Keepalive timer times out, if two LDP peers have no information to exchange, they can send Keepalive messages to each other to maintain the LDP session. If an LSR does not receive any LDP PDU from its peer during a Keepalive interval, it closes the TCP connection and terminates the LDP session.

- Receiving a shutdown message from the peer

An LSR can also send a Shutdown message to its LDP peer to terminate the LDP session. Therefore, when receiving the Shutdown message from an LDP peer, an LSR terminates the session with the LDP peer.

Protocols

MPLS related protocols include:

- RFC 3031, *Multiprotocol Label Switching Architecture*
- RFC 3032, *MPLS Label Stack Encoding*
- RFC 5036, *LDP Specification*

MPLS configuration task list

Complete the following tasks to configure MPLS:

Task		Remarks	
Enabling the MPLS function		Required	
Configuring a static LSP		Required	Use either the static or dynamic LSP configuration method.
Establishing dynamic LSPs through LDP	Configuring MPLS LDP capability	Required	
	Configuring local LDP session parameters	Optional	
	Configuring remote LDP session parameters	Optional	
	Configuring PHP	Optional	
	Configuring the policy for triggering LSP establishment	Optional	
	Configuring the label distribution control mode	Optional	
	Configuring LDP loop detection	Optional	
	Configuring LDP MD5 authentication	Optional	
	Configuring LDP label filtering	Optional	
Maintaining LDP sessions	Configuring BFD for MPLS LDP	Optional	
	Resetting LDP sessions	Optional	
	Configuring TTL processing mode at ingress	Optional	
	Configuring sending of MPLS TTL timeout messages	Optional	
	Configuring LDP GR	Optional	
Configuring MPLS statistics	Setting the interval for collecting LSP statistics	Optional	
Inspecting LSPs	MPLS LSP ping	Optional	
	MPLS LSP tracert	Optional	
	Configuring BFD for LSPs	Optional	
	Configuring periodic LSP tracert	Optional	
Enabling MPLS trap		Optional	

Only VLAN-interface supports MPLS capability.

Enabling the MPLS function

In an MPLS domain, you must enable MPLS on all routers for MPLS forwarding before you can configure other MPLS features.

Prerequisites

Before enabling MPLS, complete the following tasks:

- Configure link layer protocols to ensure the connectivity at the link layer.
- Assign IP addresses to interfaces, making all neighboring nodes reachable at the network layer.
- Configure static routes or an IGP protocol, so that LSRs can communicate with each other at the network layer.

Procedure

To enable MPLS:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Configure the MPLS LSR ID.	**mpls lsr-id** *lsr-id*	Required. Not configured by default.
3. Enable MPLS globally and enter MPLS view.	**mpls**	Required. Not enabled by default.
4. Return to system view.	**quit**	—
5. Enter interface view.	**interface** *interface-type interface-number*	—
6. Enable MPLS for the interface.	**mpls**	Required. Not enabled by default.

An MPLS LSR ID is in the format of an IP address and must be unique within an MPLS domain. HP recommends you to use the IP address of a loopback interface as the MPLS LSR ID.

The A5800 switch series support enabling MPLS on only VLAN interfaces.

Because MPLS encapsulates original packets with single layer or multiple layers of labels, after enabling MPLS on the VLAN interface of a VLAN, enable the jumboframe function on the ports of the VLAN to prevent packets from being dropped due to size limit. For more information about the jumboframe function, see *Layer 2—LAN Switching Configuration Guide*.

Configuring a static LSP

The principle of establishing a static LSP is that the outgoing label of an upstream LSR is the incoming label of its downstream LSR.

Prerequisites

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

- Determine the ingress LSR, transit LSRs, and egress LSR for the static LSP.
- Enable MPLS on all these LSRs.

Procedure

To configure a static LSP:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Configure a static LSP taking the current LSR as the ingress.	**static-lsp ingress** *lsp-name* **destination** *dest-addr* { *mask* | *mask-length* } **nexthop** *next-hop-addr* **out-label** *out-label*	Required
3. Configure a static LSP taking the current LSR as a transit LSR.	**static-lsp transit** *lsp-name* **incoming-interface** *interface-type interface-number* **in-label** *in-label* **nexthop** *next-hop-addr* **out-label** *out-label*	Required
4. Configure a static LSP taking the current LSR as the egress.	**static-lsp egress** *lsp-name* **incoming-interface** *interface-type interface-number* **in-label** *in-label*	Required

When configuring a static LSP on the ingress LSR, be sure that there is a route available from the ingress LSR to the FEC destination. This is not required on the transit LSRs and egress LSR.

When you configure a static LSP on the ingress LSR, the next hop specified must be consistent with the next hop of the optimal route in the routing table. If you configure a static IP route for the LSP, be sure to specify the same next hop for the static route and the static LSP.

For an ingress or transit LSR, do not specify the public address of an interface on the LSR as the next hop address.

For information about configuring a static IP route, see *Layer 3—IP Routing Configuration Guide*.

Establishing dynamic LSPs through LDP

Configuring MPLS LDP capability

To configure MPLS LDP capability:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enable LDP capability globally and enter MPLS LDP view.	**mpls ldp**	Required. Not enabled by default.
3. Configure the LDP LSR ID.	**lsr-id** *lsr-id*	Optional. MPLS LSR ID of the LSR by default.

Step	Command	Remarks
4. Return to system view.	**quit**	—
5. Enter interface view.	**interface** *interface-type interface-number*	—
6. Enable LDP capability for the interface.	**mpls ldp**	Required. Not enabled by default.

Only VLAN-interface supports LDP capability.

Disabling LDP on an interface terminates all LDP sessions on the interface. As a result, all LSPs using the sessions are deleted.

Usually, you do not need to configure the LDP LSR ID but use the default, which is the MPLS LSR ID. In some VPN applications (for example, MPLS L3VPN applications), however, you must make sure that different LDP instances have different LDP LSR IDs if the address spaces overlap so that the TCP connections can be established normally.

Configuring local LDP session parameters

CAUTION:

If you configure an LDP transport address by specifying an IP address, the specified IP address must be the IP address of an interface on the device so that the LDP sessions can be established.

LDP sessions established between local LDP peers are referred to as local LDP sessions. To establish a local LDP session:

- Determine the LDP transport addresses of the two peers and make sure that the LDP transport addresses are reachable to each other. This is to establish the TCP connection.
- If many LDP sessions exist between the two LSRs or the CPU is occupied much, adjust timers to ensure the stability of the LDP sessions.

To configure local LDP session parameters:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view.	**interface** *interface-type interface-number*	—
3. Set the link Hello timer	**mpls ldp timer hello-hold** *value*	Optional. 15 seconds by default.
4. Set the link Keepalive timer.	**mpls ldp timer keepalive-hold** *value*	Optional. 45 seconds by default.
5. Configure the LDP transport address.	**mpls ldp transport-address** { *ip-address* | **interface** }	Optional. MPLS LSR ID of the LSR by default.

Configuring remote LDP session parameters

LDP sessions established between remote LDP peers are referred to as remote LDP sessions. Remote LDP sessions are mainly used in Martini MPLS L2VPN, Martini VPLS, and MPLS LDP over MPLS TE. For more information, see "Configuring MPLS L2VPN," "VPLS configuration," and "Configuring MPLS TE."

To establish a remote LDP session:

- Determine the LDP transport addresses of the two peers and make sure that the LDP transport addresses are reachable to each other. This is to establish the TCP connection.
- If many LDP sessions exist between the two LSRs or the CPU is occupied much, adjust timers to ensure the stability of the LDP sessions.

To configure remote LDP session parameters:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a remote peer entity and enter MPLS LDP remote peer view.	**mpls ldp remote-peer** *remote-peer-name*	Required.
3. Configure the remote peer IP address.	**remote-ip** *ip-address*	Required.
4. Configure LDP to advertise prefix-based labels through a remote session.	**prefix-label advertise**	Optional. By default, LDP does not advertise prefix-based labels through a remote session.
5. Set the targeted Hello timer.	**mpls ldp timer hello-hold** *value*	Optional. 45 seconds by default.
6. Set the targeted Keepalive timer.	**mpls ldp timer keepalive-hold** *value*	Optional. 45 seconds by default.
7. Configure the LDP transport address.	**mpls ldp transport-address** *ip-address*	Optional. MPLS LSR ID of the LSR by default.

The IP address specified as the LDP transport address must be the IP address of an interface on the device.

The remote peer IP address to be configured must be different from all existing remote peer IP addresses.

If a local adjacency exists between two peers, no remote adjacency can be established between them. If a remote adjacency exists between two peers, you can configure a local adjacency for them. However, the local adjacency can be established only when the transport address and keepalive settings for the local peer and those for the remote peer match respectively, in which case the remote adjacency is removed. Only one remote session or local session can exist between two LSRs, and the local session takes precedence over the remote session.

By default, LDP does not advertise any prefix-based label mapping message through a remote session, and remote sessions are used only to transfer messages for L2VPNs. For applications of remote sessions, see "MPLS L2VPN configuration."

Configuring PHP

To configure PHP:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Specify the type of the label to be distributed by the egress to the penultimate hop.	**label advertise** { **explicit-null** | **implicit-null** | **non-null** }	Optional. By default, an egress distributes to the penultimate hop an implicit null label.

For the A5800 switch series, a label with a value of 0 can be at the top of a label stack. After receiving a packet with such a label, the switch pops the label directly and check whether there is any inner layer label. If finding an inner layer label, the switch forwards the packet based on the inner layer label. If no inner layer label is found, the switch forwards the packet based on the IP address.

The device supports PHP when it works as a penultimate hop.

For LDP sessions existing before **label advertise** is configured, you must reset the LDP sessions by using **reset mpls ldp** for the PHP configuration to take effect.

Configuring the policy for triggering LSP establishment

You can configure an LSP triggering policy on an LSR, so that only routes matching the policy can trigger establishment of LSPs, reducing the number of LSPs to be established on the LSR and avoiding instability of the LSR caused by excessive LSPs.

An LSR supports two types of LSP triggering policies:

- Allowing all routing entries to trigger establishment of LSPs.
- Filtering routing entries by an IP prefix list, so that static and IGP routes denied by the IP prefix list does not trigger LSP establishment.

To configure the policy for triggering LSP establishment:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Configure the LSP establishment triggering policy.	**lsp-trigger** [**vpn-instance** *vpn-instance-name*] { **all** | **ip-prefix** *prefix-name* }	Optional. By default, only host routes with 32-bit masks can trigger establishment of LSPs.

For an LSP to be established, an exactly matching routing entry must exist on the LSR. For example, on an LSR, to establish an LSP to a loopback address with a 32-bit mask, there must be an exactly matching host routing entry on the LSR.

If the vpn-instance vpn-instance-name combination is specified, the command configures an LSP establishment triggering policy for the specified VPN. If the vpn-instance vpn-instance-name combination is not specified, the command configures an LSP establishment triggering policy for the public network routes.

For information about IP prefix list, see *Layer 3—IP Routing Configuration Guide*.

Configuring the label distribution control mode

With the label re-advertisement function enabled, an LSR periodically looks for FECs not assigned with labels, assigns labels to them if any, and advertises the label-FEC bindings. You can set the label re-advertisement interval as needed.

To configure the LDP label distribution control mode:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS LDP view.	**mpls ldp**	—
3. Specify the label distribution control mode.	**label-distribution** { **independent** | **ordered** }	Optional. Ordered by default. For LDP sessions existing before the command is configured, you must reset the LDP sessions for the specified label distribution control mode to take effect.
4. Enable label re-advertisement for DU mode.	**du-readvertise**	Optional. Enabled by default.
5. Set the interval for label re-advertisement in DU mode.	**du-readvertise timer** *value*	Optional. 30 seconds by default.

Configuring LDP loop detection

LSPs established in an MPLS domain may be looping. The LDP loop detection mechanism can detect looping LSPs and prevent LDP messages from looping forever.

LDP loop detection can be in either of the following two modes:

1. Maximum hop count

A label request message or label mapping message carries information about its hop count, which increments by 1 for each hop. When this value reaches the specified limit, LDP considers that a loop is present and terminates the establishment of the LSP.

2. Path vector

A label request message or label mapping message carries path information in the form of path vector list. When such a message reaches an LSR, the LSR checks the path vector list of the message to see whether its MPLS LSR ID is in the list. If the MPLS LSR ID is not on the list, the LSR adds its LSR ID to the path vector list; if the MPLS LSR ID is on the list, the LSR considers that a loop appears and terminates the establishment of the LSP.

In the path vector mode, you also must specify the maximum number of hops of an LSP. An LSR also terminates the establishment of an LSP when the hop count of the path, or the length of the path vector, reaches the specified limit.

To configure LDP loop detection:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS LDP view.	**mpls ldp**	—
3. Enable loop detection.	**loop-detect**	Required. Disabled by default.
4. Set the maximum hop count.	**hops-count** *hop-number*	Optional. 32 by default.
5. Set the maximum path vector length.	**path-vectors** *pv-number*	Optional. 32 by default.

The loop detection modes configured on two LDP peers must be the same so that the LDP session can be established.

To implement loop detection in an MPLS domain, you must enable loop detection on every LSR in the MPLS domain.

You must configure loop detection before enabling LDP capability on any interface.

All loop detection configurations take effect for only the LSPs established after the configurations. Changing the loop detection configurations does not affect existing LSPs. You can execute **reset mpls ldp** in user view, so that the loop detection configurations can take effect for all LSPs.

LDP loop detection may result in LSP update, which generates redundant information and consume many system resources. Configure the routing loop detection methods to avoid LDP loops.

Configuring LDP MD5 authentication

LDP sessions are established based on TCP connections. To improve the security of LDP sessions, you can configure MD5 authentication for the underlying TCP connections, so that the TCP connections can be established only if the peers have the same authentication password.

To configure LDP MD5 authentication:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS LDP view.	**mpls ldp**	—
3. Enable LDP MD5 authentication and set the password.	**md5-password** { **cipher** | **plain** } *peer-lsr-id password*	Required. Disabled by default.

To establish an LDP session successfully between two LDP peers, make sure that the LDP MD5 authentication configurations on the LDP peers are consistent.

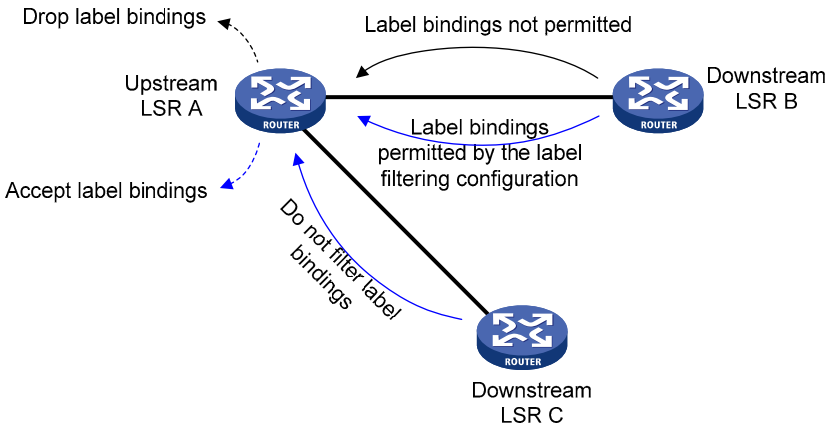
Configuring LDP label filtering

The LDP label filtering feature provides two mechanisms, label acceptance control for controlling which labels are accepted and label advertisement control for controlling which labels are advertised. In complicated MPLS network environments, LDP label filtering can be used to control which LSPs are to be established dynamically and prevent devices from accepting and advertising excessive label bindings.

1. Label acceptance control

Label acceptance control is for filtering received label bindings. An upstream LSR filters the label bindings received from the specified downstream LSR and accepts only those permitted by the specified prefix list. As shown in Figure 19, upstream device LSR A filters the label bindings received from downstream device LSR B. Only if the destination address of an FEC matches the specified prefix list, does LSR A accept the label binding of the FEC from LSR B. LSR A does not filter label bindings received from downstream device LSR C.

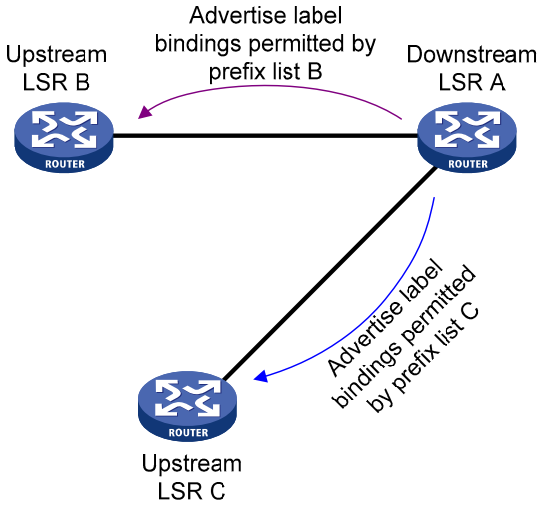
Figure 19 Network diagram for label acceptance control



2. Label advertisement control

Label advertisement control is for filtering label bindings to be advertised. A downstream LSR advertises only the label bindings of the specified FECs to the specified upstream LSR. As shown in Figure 20, downstream device LSR A advertises to upstream device LSR B only label bindings with FEC destinations permitted by prefix list B, and advertises to upstream device LSR C only label bindings with FEC destinations permitted by prefix list C.

Figure 20 Network diagram for label advertisement control



To configure LDP label filtering policies:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS LDP view.	**mpls ldp**	—
3. Configure a label acceptance control policy.	**accept-label peer** *peer-id* **ip-prefix** *ip-prefix-name*	Optional. Not configured by default.
4. Configure a label advertisement control policy.	**advertise-label ip-prefix** *ip-prefix-name* [**peer** *peer-ip-prefix-name*]	Required. Not configured by default.

For two neighboring LSRs, configuring a label acceptance control policy on the upstream LSR and configuring a label advertisement control policy on the downstream LSR can achieve the same effect. To reduce the network load, HP recommends configuring only label advertisement control policies.

Maintaining LDP sessions

Configuring BFD for MPLS LDP

MPLS itself cannot detect a neighbor failure or link failure in time. If communication between two remote LDP peers fails, the LDP session is down, and as a result, MPLS forwarding fails. By cooperating with BFD, MPLS LDP can be quickly aware of communication failures between remote LDP peers, improving performances of existing MPLS networks.

For more information about BFD, see *High Availability Configuration Guide*.

An LSP can be bound to only one BFD session.

To configure BFD for MPLS LDP:

Step	Command	Remarks
1. Enter system view.	system-view	—
2. Enter MPLS LDP remote peer view.	mpls ldp remote-peer *remote-peer-name*	—
3. Enable BFD for MPLS LDP.	remote-ip bfd	Required. Disabled by default.

The cooperation of MPLS LDP and BFD can only be used to detect communication failures between remote LDP peers. For related configuration examples, see "VPLS configuration."

Resetting LDP sessions

If you change LDP session parameters when some LDP sessions are up, the LDP sessions do not function normally. In this case, you must reset the LDP session so that the LDP peers renegotiate parameters and establish new sessions.

Use the following command to reset LDP sessions:

Task	Command	Remarks
Reset LDP sessions	**reset mpls ldp** [**all** | [**vpn-instance** *vpn-instance-name*] [*fec mask* | **peer** *peer-id*]]	Available in user view

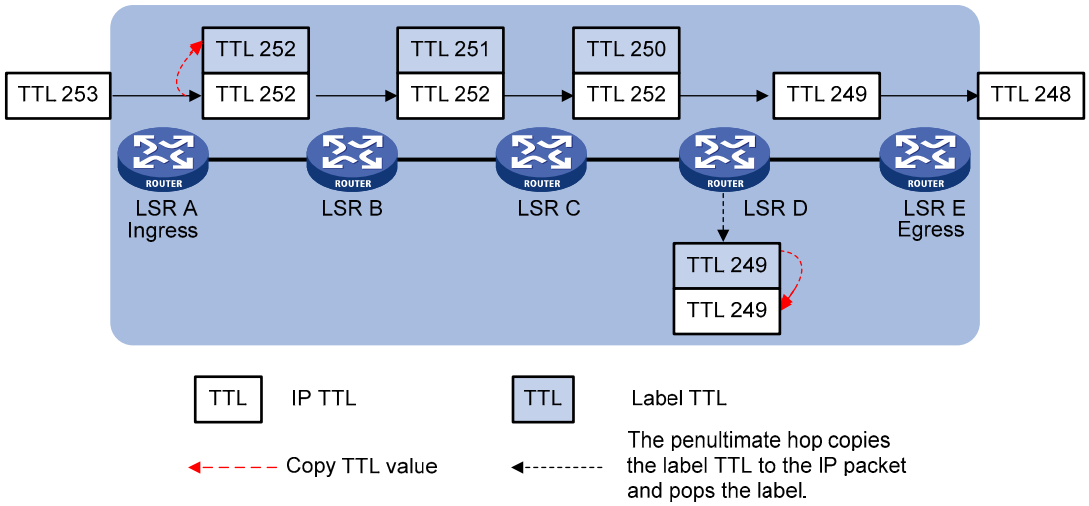
Managing and optimizing MPLS forwarding

Configuring TTL processing mode at ingress

At the ingress of an LSP, a label stack encapsulated with a TTL field is added to each packet. Whether the label TTL takes the IP TTL or not depends on whether IP TTL propagation is enabled:

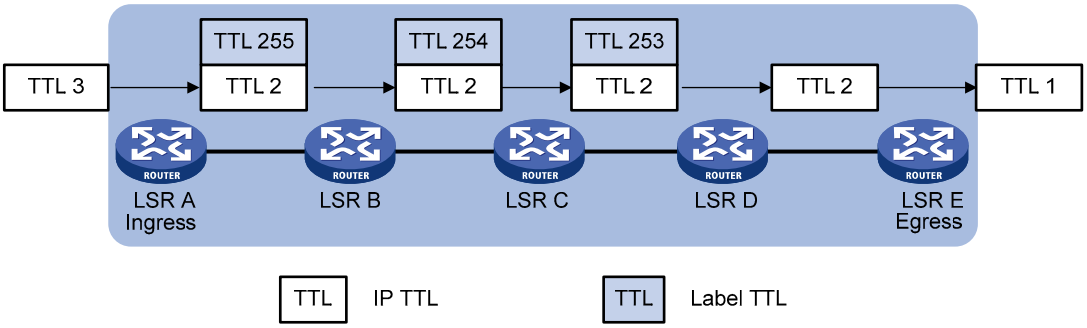
- With IP TTL propagation enabled: When the ingress labels a packet, it copies the TTL value of the original IP packet to the TTL field of the label. When an LSR forwards the labeled packet, it decrements the TTL value of the label at the stack top by 1. When an LSR pops a label, it copies the TTL value of the label at the stack top back to the TTL field of the IP packet. In this case, the TTL value of a packet is decreased hop by hop when forwarded along the LSP. Therefore, the result of tracert reflects the real path along which the packet has traveled.

Figure 21 Label TTL processing when IP TTL propagation is enabled



- With IP TTL propagation disabled: When the ingress labels a packet, it does not copy the TTL value of the original IP packet to the TTL field of the label, and the label TTL is set to 255. When an LSR forwards the labeled packet, it decrements the TTL value of the label at the stack top by 1. When an LSR pops a label, it compares the IP TTL and the label TTL and uses the smaller value as the TTL of the IP packet. In this case, the result of tracert does not show the hops within the MPLS backbone, as if the ingress and egress were connected directly.

Figure 22 Label TTL processing when IP TTL propagation is disabled



To configure IP TTL propagation of MPLS:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Enable MPLS IP TTL propagation.	**ttl propagate** { **public** | **vpn** }	Optional. Enabled for only public network packets by default.

Within an MPLS domain, TTL is always copied between multiple levels of labels. The ttl propagate command affects only the propagation of the IP TTL to the TTL of an MPLS label. Therefore, this command only takes effect when it is configured on the ingress.

For locally generated packets, an LSR always copies the IP TTL value of the packet, regardless of whether IP TTL propagation is enabled or not. This ensures that the local administrator can tracert for network diagnoses.

If you enable MPLS IP TTL propagation for VPN packets on one LSR, HP recommends you to do so on all related PEs, so that you can get the same result when tracerting from those PEs.

Some VPN packets carry two layers of labels, outer and inner, for transmission in the public network and private network respectively. A5800 series can copy the IP TTL of private packets to the outer label directly.

Configuring sending of MPLS TTL timeout messages

After sending of MPLS TTL timeout message is enabled on an LSR, when the LSR receives an MPLS packet that carries a label with TTL being 1, it generates an ICMP time exceeded message, and send the message to the packet sender in one of the following ways:

- If there is a route from the LSR to the packet sender, the LSR sends the TTL timeout message to the packet sender directly through the IP route.
- If there is no route from the LSR to the packet sender, the LSR forwards the TTL timeout message along the LSP to the egress, which sends the TTL timeout message to the packet sender.

Usually, for an MPLS packet carrying only one level of label, the first method is used; for an MPLS packet carrying a multi-level label stack, the second method is used. However, because ASBRs, SPEs in HoVPN applications, and carrier backbone PEs in nested VPNs may receive MPLS VPN packets that carry only one level of labels but these devices have no IP routes to the packet senders, the first method is not applicable. In this case, you can configure **undo ttl expiration pop** on these devices so that the devices use the second method.

For more information about HoVPN and nested VPN, see "MPLS L3VPN configuration."

To configure sending of MPLS TTL timeout messages (ICMP error messages):

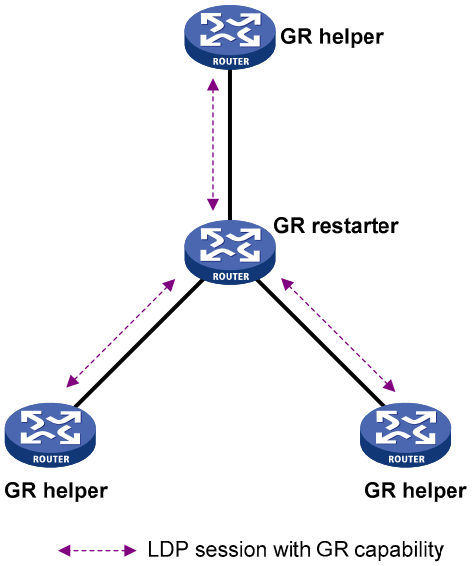
Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Enable the device to send an ICMP error message when the TTL of an MPLS packet expires.	**ttl expiration enable**	Optional. Enabled by default.
4. Specify that TTL timeout messages for MPLS packets with only one-level of labels travel along the IP routes.	**ttl expiration pop**	Optional. Configure one of them as Required. By default, MPLS TTL timeout messages for MPLS packets with only one-level of labels travel along the local IP routes.
5. Specify that MPLS TTL timeout messages for MPLS packets with only one-level of labels travel along the LSPs.	**undo ttl expiration pop**	This configuration does not take effect when the MPLS packets carry multiple levels of labels. MPLS TTL timeout messages for such MPLS packets always travel along the LSPs.

Configuring LDP GR

MPLS has two separate planes: the forwarding plane and the control plane. Using this feature, LDP GR preserves the LFIB information when the signaling protocol or control plane fails, so that LSRs can still forward packets according to LFIB, ensuring continuous data transmission.

- GR Restarter, or graceful restarting router, is where protocol restart is triggered administratively or because of a fault. It must be Graceful Restart capable.
- GR Helper is the neighbor of the GR Restarter. It helps the GR Restarter maintain routing information. It must be Graceful Restart capable.

Figure 23 LDP GR



As shown in Figure 23, two LDP peers perform GR negotiation when establishing an LDP session. The LDP session established is GR capable only when both peers support LDP GR.

The working procedure of LDP GR is as follows:

1. Whenever restarting, the GR restarter preserves all MPLS forwarding entries, marks them as stale, and starts the MPLS forwarding state holding timer for them.
2. After a GR helper detects that the LDP session with the GR restarter is down, it marks the FEC-label bindings learned from the session as stale and keeps these FEC-label bindings for a period of time defined by the FT reconnect time argument. The FT reconnect time is the smaller one between the reconnect time advertised from the peer GR restarter and the neighbor liveness time configured locally.
3. During the FT reconnect time, if the LDP session fails to be re-established, the GR helper deletes the FEC-label bindings marked stale.
4. If the session is re-established successfully, during the LDP recovery time, the GR helper and the GR restarter uses the new LDP session to exchange the label mapping information, update the LFIB, and delete the stale marks of the corresponding forwarding entries. The LDP recovery time is the smaller one between the recovery time configured locally and that configured on the peer GR restarter.
5. After the recovery time elapses, the GR helper deletes the FEC-label bindings that are still marked stale.
6. When the MPLS forwarding state holding timer expires, the GR restarter deletes the label forwarding entries that are still marked stale.

Prerequisites

Configure MPLS LDP capability on each device that acts as the GR restarter or a GR helper.

The A5800 series can act as both a GR restarter and a GR helper.

Configuring LDP GR

To configure LDP GR:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS LDP view.	**mpls ldp**	—
3. Enable MPLS LDP GR.	**graceful-restart**	Required. Disabled by default.
4. Set the FT reconnect time.	**graceful-restart timer reconnect** *timer*	Optional. 300 seconds by default.
5. Set the LDP neighbor liveness time.	**graceful-restart timer neighbor-liveness** *timer*	Optional. 120 seconds by default.
6. Set the LDP recovery time.	**graceful-restart timer recovery** *timer*	Optional. 300 seconds by default.

Gracefully restarting MPLS LDP

To test whether the MPLS LDP GR configuration has taken effect, you can perform graceful restart of MPLS LDP. During the LDP restart process, you can see whether the packet forwarding path is changed and whether packet forwarding is interrupted.

Use the following command to restart MPLS LDP gracefully:

Task	Command	Remarks
Restart MPLS LDP gracefully.	**graceful-restart mpls ldp**	Required. Available in user view.

Do not perform this operation in normal cases.

Configuring MPLS statistics

Setting the interval for collecting LSP statistics

To view MPLS statistics, you must set the interval for collecting the LSP statistics.

To set the interval for collecting LSP statistics:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Set the interval for collecting statistics.	**statistics interval** *interval-time*	Required. 0 seconds by default, meaning that the system does not collect statistics.

Inspecting LSPs

In MPLS, the MPLS control plane is responsible for establishing LSPs. However, when an LSP fails to forward data, the control plane cannot detect the LSP failure or cannot do so in time. This makes network maintenance difficult. To find LSP failures in time and locate the failed node, the device provides the following mechanisms:

- MPLS LSP ping
- MPLS LSP tracert
- BFD for LSPs
- Periodic LSP tracert

MPLS LSP ping

MPLS LSP ping is for checking the validity and availability of an LSP. At the ingress, it adds the label for the FEC to be inspected into an MPLS echo request, which then is forwarded along the LSP to the egress. The egress processes the request packet and returns an MPLS echo reply to the ingress. An MPLS echo reply carrying a success notification indicates that the lSP is normal, and an MPLS echo reply carrying an error code indicates that the LSP has failed.

Use the following command to check the validity and availability of an LSP:

Task	Command	Remarks
Use MPLS LSP ping to check the validity and reachability of an MPLS LSP.	**ping lsp** [**-a** *source-ip* | **-c** *count* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *wait-time* | **-r** *reply-mode* | **-s** *packet-size* | **-t** *time-out* | **-v**] * **ipv4** *dest-addr mask-length* [*destination-ip-addr-header*]	Available in any view

MPLS LSP tracert

MPLS LSP tracert is for locating LSP errors. It consecutively sends the MPLS echo requests along the LSP to be inspected, with the TTL increasing from 1 to a specified value. Then, each hop along the LSP returns an MPLS echo reply to the ingress due to TTL timeout. Thus, the ingress can collect the information of each hop along the LSP, so as to locate the failed node. You can also use MPLS LSP tracert to collect the important information of each hop along the LSP, such as the label allocated.

Use the following command to locate errors of an LSP:

Task	Command	Remarks
Perform MPLS LSP tracert to locate an MPLS LSP error.	**tracert lsp** [**-a** *source-ip* | **-exp** *exp-value* | **-h** *ttl-value* | **-r** *reply-mode* |**-t** *time-out*] * **ipv4** *dest-addr mask-length* [*destination-ip-addr-header*]	Available in any view

After you configure tracert for an LSP, if the label to be assigned by the egress node of the LSP to the penultimate hop is an implicit null label (the default setting), use **ip ttl-expires enable** on the egress node to enable sending of ICMP timeout packets for normal operation of the LSP tracert function. For information about labels distributed by the egress to the penultimate hop, see "Configuring PHP." For information about **ip ttl-expires enable**, see *Layer 3—IP Services Command Reference*.

Configuring BFD for LSPs

You can configure BFD for an LSP to detect the connectivity of the LSP. After the configuration, a BFD session is established between the ingress and egress of the LSP, and the ingress adds the label for the FEC to into a BFD control packet, forward the BFD control packet along the LSP to the egress, and determine the status of the LSP according to the reply received. Upon detecting an LSP failure, BFD triggers a traffic switchover.

A BFD session for LSP connectivity detection can be static or dynamic:

- Static: If you specify the local and remote discriminator values by using the **discriminator** keyword when configuring **bfd enable**, the BFD session is established with the specified discriminator values. Such a BFD session is used to detect the connectivity of a pair of LSPs in opposite directions (one from local to remote, and the other from remote to local) between two devices.
- Dynamic: If you do not specify the local and remote discriminator values when configuring **bfd enable**, the MPLS LSP ping runs automatically to negotiate the discriminator values and then the BFD session is established based on the negotiated discriminator values. Such a BFD session is used for connectivity detection of an LSP from the local device to the remote device.

To configure BFD for LSPs:

Step	Command	Remarks
1. Enter system view.	system-view	—
2. Enable LSP verification and enter the MPLS LSPV view.	mpls lspv	Required. Not enabled by default.
3. Configure BFD to check the connectivity of the LSPs to the specified FEC destination.	**bfd enable** destination-address mask-length [**nexthop** nexthop-address [**discriminator local** local-id **remote** remote-id]]	Required. Not configured by default.

The BFD session parameters are those configured on the loopback interface whose IP address is configured as the MPLS LSR ID, and the BFD packets use the MPLS LSR ID as the source address. Therefore, before enabling BFD for an LSP, you must configure an IP address for the loopback interface and configure the MPLS LSR ID as the IP address of the loopback interface, and you can also configure BFD session parameters for the loopback interface as needed.

For more information about BFD, see *High Availability Configuration Guide*.

You cannot establish both a static BFD session and a dynamic BFD session for the same LSP.

Before establishing a static BFD session, make sure that the local device has an LSP to the remote device and the remote device has an LSP to the local device.

After a static BFD session is established, it is not allowed to modify the discriminator values of the BFD session.

BFD for MPLS LDP is for detecting the IP connectivity between two remote LDP peers. BFD for LSP is for detecting the connectivity of LSPs.

Configuring periodic LSP tracert

The periodic LSP tracert function is for locating faults of an LSP periodically. It detects the consistency of the forwarding plane and control plane and records detection results into logs. You can know whether an LSP has failed by checking the logs.

If you configure BFD as well as periodic tracert for an LSP, once the periodic LSP tracert function detects an LSP fault or inconsistency of the forwarding plane and control plane, the BFD session for the LSP are deleted and a new BFD session is established according to the control plane.

To configure the periodic LSP tracert function:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enable LSP verification and enter the MPLS LSPV view.	**mpls lspv**	Required. Not enabled by default.
3. Configure periodic tracert for an LSP to the specified FEC destination.	**periodic-tracert** *destination-address mask-length* [**-a** *source-ip* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *wait-time* | **-t** *time-out* | **-u** *retry-attempt*] *	Required. Not configured by default.

After you configure periodic tracert for an LSP, if the label to be assigned by the egress node of the LSP to the penultimate hop is an implicit null label (the default setting), use **ip ttl-expires enable** on the egress node to enable sending of ICMP timeout packets for normal operation of the LSP tracert function. For information about labels distributed by the egress to the penultimate hop, see "Configuring PHP." For information about **ip ttl-expires enable**, see *Layer 3—IP Services Command Reference*.

Enabling MPLS trap

With the MPLS trap function enabled, trap packets of the notifications level is generated to report critical MPLS events. Such trap packets are sent to the information center of the device. Whether and where the packets are then output depend on the configurations of the information center. For information on how to configure the information center, see *Network Management and Monitoring Configuration Guide*.

To enable the MPLS trap function:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enable the MPLS trap function.	**snmp-agent trap enable mpls**	Required. Disabled by default.

For more information about **snmp-agent trap enable mpls**, see *Network Management and Monitoring Command Reference*.

Displaying and maintaining MPLS

Displaying MPLS information

Task	Command	Remarks
Display information about one or all interfaces with MPLS enabled.	**display mpls interface** [*interface-type interface-number*] [**verbose**] [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

Task	Command	Remarks
Display information about ILM entries.	**display mpls ilm** [*label*] [**verbose**] [**slot** *slot-number*] [**include** *text* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]	Available in any view
Display information about specified MPLS labels or all labels.	**display mpls label** { *label-value1* [**to** *label-value2*] | **all** } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about LSPs.	**display mpls lsp** [**incoming-interface** *interface-type interface-number*] [**outgoing-interface** *interface-type interface-number*] [**in-label** *in-label-value*] [**out-label** *out-label-value*] [**asbr** | [**vpn-instance** *vpn-instance-name*] [**protocol** { **bgp** | **bgp-ipv6** | **crldp** | **ldp** | **rsvp-te** | **static** | **static-cr** }]] [**egress** | **ingress** | **transit**] [{ **exclude** | **include** } *dest-addr mask-length*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display LSP statistics.	**display mpls lsp statistics** [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display the BFD detection information for an LSP.	**display mpls lsp bfd** [**ipv4** *destination-address mask-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about NHLFE entries.	**display mpls nhlfe** [*token*] [**verbose**] [**slot** *slot-number*] [**include** *text* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]	Available in any view
Display usage information about the NHLFE entries.	**display mpls nhlfe reflist** *token* [**slot** *slot-number*] [**include** *text* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]	Available in any view
Display information about static LSPs.	**display mpls static-lsp** [**lsp-name** *lsp-name*] [{ **exclude** | **include** } *dest-addr mask-length*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display LSP-related route information.	**display mpls route-state** [**vpn-instance** *vpn-instance-name*] [*dest-addr mask-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

Task	Command	Remarks
Display statistics for all LSPs or the LSP with a specified index or name.	**display mpls statistics lsp** { *index* | **all** | **name** *lsp-name* } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display MPLS statistics for one or all interfaces.	**display mpls statistics interface** { *interface-type interface-number* | **all** } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

Displaying MPLS LDP information

Task	Command	Remarks
Display information about LDP.	**display mpls ldp** [**all** [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]]	Available in any view
Display the label advertisement information of the specified FEC.	**display mpls ldp fec** [**vpn-instance** *vpn-instance-name*] *dest-addr mask-length* [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about LDP-enabled interfaces.	**display mpls ldp interface** [**all** [**verbose**] | [**vpn-instance** *vpn-instance-name*] [*interface-type interface-number* | **verbose**]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about LDP peers.	**display mpls ldp peer** [**all** [**verbose**] | [**vpn-instance** *vpn-instance-name*] [*peer-id* | **verbose**]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about Remote LDP peers.	**display mpls ldp remote-peer** [**remote-name** *remote-peer-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about LDP sessions between LDP peers.	**display mpls ldp session** [**all** [**verbose**] | [**vpn-instance** *vpn-instance-name*] [*peer-id* | **verbose**]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display statistics information about all LSP sessions.	**display mpls ldp session all statistics** [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about LSPs established by LDP.	**display mpls ldp lsp** [**all** | [**vpn-instance** *vpn-instance-name*] [*dest-addr mask-length*]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

The **Vpn-instance** *vpn-instance-name* in the commands is used to specify information about an LDP instance. For information about LDP instances, see "MPLS L3VPN configuration."

Clearing MPLS statistics

Task	Command	Remarks
Clear MPLS statistics for one or all MPLS interfaces.	**reset mpls statistics interface** { *interface-type interface-number* | **all** }	Available in user view
Clear MPLS statistics for all LSPs or the LSP with a specified index or name.	**reset mpls statistics lsp** { *index* | **all** | **name** *lsp-name* }	Available in user view

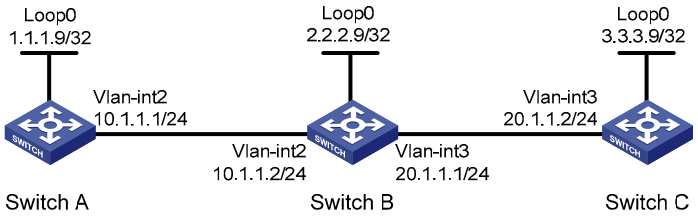
MPLS configuration examples

Configuring static LSPs

Network requirements

- Switch A, Switch B, and Switch C support MPLS and use OSPF as the IGP for the MPLS backbone.
- Establish a static LSP from Switch A to Switch C and a static LSP from Switch C to Switch A.
- Check the validity and connectivity of the LSPs.

Figure 24 Network diagram for configuring static LSPs on switches



Procedure

1. Configure the IP addresses of the interfaces

Configure the IP addresses and masks of the interfaces including the loopback interfaces as required in Figure 24. (Omitted)

2. Configure OSPF to ensure reachability between the switches

Configure OSPF on Switch A.

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

Configure OSPF on Switch B.

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

Configure OSPF on Switch C.

```
<SwitchC> system-view
[SwitchC] ospf
[SwitchC-ospf-1] area 0
[SwitchC-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[SwitchC-ospf-1-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[SwitchC-ospf-1-area-0.0.0.0] quit
[SwitchC-ospf-1] quit
```

Execute **display ip routing-table** on each switch. Each switch has learned the routes to other switches. The following uses Switch A as an example:

```
[SwitchA] display ip routing-table
Routing Tables: Public

         Destinations : 8        Routes : 8

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   1562         10.1.1.2        Vlan2
3.3.3.9/32          OSPF   10   3124         10.1.1.2        Vlan2
10.1.1.0/24         Direct 0    0            10.1.1.1        Vlan2
10.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
20.1.1.0/24         OSPF   10   3124         10.1.1.2        Vlan2
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. Enable MPLS

Configure MPLS on Switch A.

```
[SwitchA] mpls lsr-id 1.1.1.9
[SwitchA] mpls
[SwitchA-mpls] quit
[SwitchA] interface vlan-interface 2
[SwitchA-Vlan-interface2] mpls
[SwitchA-Vlan-interface2] quit
```

Configure MPLS on Switch B.

```
[SwitchB] mpls lsr-id 2.2.2.9
[SwitchB] mpls
[SwitchB-mpls] quit
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] mpls
```

```
[SwitchB-Vlan-interface2] quit
[SwitchB] interface vlan-interface 3
[SwitchB-Vlan-interface3] mpls
[SwitchB-Vlan-interface3] quit
```

Configure MPLS on Switch C.

```
[SwitchC] mpls lsr-id 3.3.3.9
[SwitchC] mpls
[SwitchC-mpls] quit
[SwitchC] interface vlan-interface 3
[SwitchC-Vlan-interface3] mpls
[SwitchC-Vlan-interface3] quit
```

4. Create a static LSP from Switch A to Switch C

Configure the LSP ingress, Switch A.

```
[SwitchA] static-lsp ingress AtoC destination 3.3.3.9 32 nexthop 10.1.1.2 out-label 30
```

Configure the LSP transit node, Switch B.

```
[SwitchB] static-lsp transit AtoC incoming-interface vlan-interface 2 in-label 30 nexthop
20.1.1.2 out-label 50
```

Configure the LSP egress, Switch C.

```
[SwitchC] static-lsp egress AtoC incoming-interface vlan-interface 3 in-label 50
```

5. Create a static LSP from Switch C to Switch A

Configure the LSP ingress, Switch C.

```
[SwitchC] static-lsp ingress CtoA destination 1.1.1.9 32 nexthop 20.1.1.1 out-label 40
```

Configure the LSP transit node, Switch B.

```
[SwitchB] static-lsp transit CtoA incoming-interface vlan-interface 3 in-label 40 nexthop
10.1.1.1 out-label 70
```

Configure the LSP egress, Switch A.

```
[SwitchA] static-lsp egress CtoA incoming-interface vlan-interface 2 in-label 70
```

6. Verify the configuration

Execute **display mpls static-lsp** on each switch to view the static LSP information. The following uses Switch A as an example:

```
[SwitchA] display mpls static-lsp
total static-lsp : 2
Name            FEC               I/O Label  I/O If                    State
AtoC            3.3.3.9/32        NULL/30    -/Vlan2                   Up
CtoA            -/-               70/NULL    Vlan2/-                   Up
```

On Switch A, check the connectivity of the LSP from Switch A to Switch C.

```
[SwitchA] ping lsp ipv4 3.3.3.9 32
 LSP Ping FEC: LDP IPV4 PREFIX 3.3.3.9/32 : 100  data bytes, press CTRL_C to break
 Reply from 20.1.1.2: bytes=100 Sequence=1 time = 76 ms
 Reply from 20.1.1.2: bytes=100 Sequence=2 time = 75 ms
 Reply from 20.1.1.2: bytes=100 Sequence=3 time = 75 ms
 Reply from 20.1.1.2: bytes=100 Sequence=4 time = 75 ms
 Reply from 20.1.1.2: bytes=100 Sequence=5 time = 75 ms
 --- FEC: LDP IPV4 PREFIX 3.3.3.9/32 ping statistics ---
```

```
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 75/75/76 ms
```

On Switch C, check the connectivity of the LSP from Switch C to Switch A.

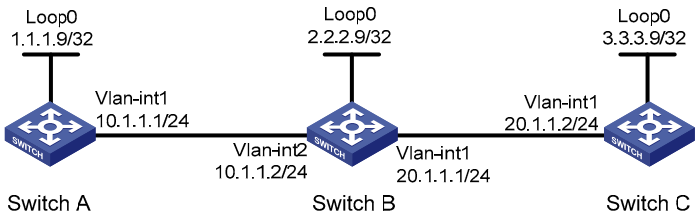
```
[SwitchC] ping lsp ipv4 1.1.1.9 32
 LSP Ping FEC: LDP IPV4 PREFIX 1.1.1.9/32 : 100  data bytes, press CTRL_C to break
 Reply from 10.1.1.1: bytes=100 Sequence=1 time = 75 ms
 Reply from 10.1.1.1: bytes=100 Sequence=2 time = 75 ms
 Reply from 10.1.1.1: bytes=100 Sequence=3 time = 75 ms
 Reply from 10.1.1.1: bytes=100 Sequence=4 time = 74 ms
 Reply from 10.1.1.1: bytes=100 Sequence=5 time = 75 ms

 --- FEC: LDP IPV4 PREFIX 1.1.1.9/32 ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 74/74/75 ms
```

Configuring LDP to establish LSPs dynamically

Network requirements

- Switch A, Switch B, and Switch C support MPLS and use OSPF as the IGP for the MPLS backbone.
- Configure LDP to establish LSPs between Switch A and Switch C dynamically.
- Check the validity and reachability of the LSPs.

Figure 25 Network diagram for establishing LDP LSPs on switches



Procedure

1. Configure the IP addresses of the interfaces

Configure the IP addresses and masks of the interfaces including the loopback interfaces as required in Figure 25. (Omitted)

2. Configure OSPF to ensure reachability between the switches

Configure OSPF on Switch A.

```
<Sysname> system-view
[Sysname] sysname SwitchA
[SwitchA] ospf
[SwitchA-ospf-1] area 0
[SwitchA-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
```

```
[SwitchA-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[SwitchA-ospf-1-area-0.0.0.0] quit
[SwitchA-ospf-1] quit
```

Configure OSPF on Switch B.

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

Configure OSPF on Switch C.

```
<Sysname> system-view
[Sysname] sysname SwitchC
[SwitchC] ospf
[SwitchC-ospf-1] area 0
[SwitchC-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[SwitchC-ospf-1-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[SwitchC-ospf-1-area-0.0.0.0] quit
[SwitchC-ospf-1] quit
```

Execute **display ip routing-table** on each switch. Each switch has learned the routes to other switches. Take Switch A as an example:

```
[SwitchA] display ip routing-table
Routing Tables: Public
         Destinations : 8        Routes : 8

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   1562         10.1.1.2        Vlan1
3.3.3.9/32          OSPF    10   3124         10.1.1.2        Vlan1
10.1.1.0/24         Direct  0    0            10.1.1.1        Vlan1
10.1.1.1/32         Direct  0    0            127.0.0.1       InLoop0
20.1.1.0/24         OSPF    10   3124         10.1.1.2        Vlan1
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. Enable MPLS and MPLS LDP

Configure MPLS and MPLS LDP on Switch A.

```
[SwitchA] mpls lsr-id 1.1.1.9
[SwitchA] mpls
[SwitchA-mpls] quit
[SwitchA] mpls ldp
[SwitchA-mpls-ldp] quit
[SwitchA] interface vlan-interface 1
```

```
[SwitchA-Vlan-interface1] mpls
[SwitchA-Vlan-interface1] mpls ldp
[SwitchA-Vlan-interface1] quit
```

Configure MPLS and MPLS LDP on Switch B.

```
[SwitchB] mpls lsr-id 2.2.2.9
[SwitchB] mpls
[SwitchB-mpls] quit
[SwitchB] mpls ldp
[SwitchB-mpls-ldp] quit
[SwitchB] interface vlan-interface 1
[SwitchB-Vlan-interface1] mpls
[SwitchB-Vlan-interface1] mpls ldp
[SwitchB-Vlan-interface1] quit
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] mpls
[SwitchB-Vlan-interface2] mpls ldp
[SwitchB-Vlan-interface2] quit
```

Configure MPLS and MPLS LDP on Switch C.

```
[SwitchC] mpls lsr-id 3.3.3.9
[SwitchC] mpls
[SwitchC-mpls] quit
[SwitchC] mpls ldp
[SwitchC-mpls-ldp] quit
[SwitchC] interface vlan-interface 1
[SwitchC-Vlan-interface1] mpls
[SwitchC-Vlan-interface1] mpls ldp
[SwitchC-Vlan-interface1] quit
```

After the configurations, two local LDP sessions are established, one between Switch A and Switch B and the other between Switch B and Switch C. Execute **display mpls ldp session** on each switch to view the LDP session information, and execute **display mpls ldp peer** to view the LDP peer information. Take Switch A as an example:

```
[SwitchA] display mpls ldp session
               LDP Session(s) in Public Network
 Total number of sessions: 1
 ------------------------------------------------------------------
 Peer-ID       Status        LAM  SsnRole  FT   MD5  KA-Sent/Rcv
 ------------------------------------------------------------------
 2.2.2.9:0     Operational   DU   Passive  Off  Off  5/5
 ------------------------------------------------------------------
 LAM : Label Advertisement Mode         FT  : Fault Tolerance
[SwitchA] display mpls ldp peer
         LDP Peer Information in Public network
 Total number of peers: 1
------------------------------------------------------------------
 Peer-ID               Transport-Address  Discovery-Source
 ------------------------------------------------------------------
 2.2.2.9:0             2.2.2.9            Vlan-interface1
```

```
 ------------------------------------------------------------------
```

4. Allow all static routes and IGP routes to trigger LDP to establish LSPs.

Configure the LSP establishment triggering policy on Switch A.

```
[SwitchA] mpls
[SwitchA-mpls] lsp-trigger all
[SwitchA-mpls] return
```

Configure the LSP establishment triggering policy on Switch B.

```
[SwitchB] mpls
[SwitchB-mpls] lsp-trigger all
[SwitchB-mpls] quit
```

Configure the LSP establishment triggering policy on Switch C.

```
[SwitchC] mpls
[SwitchC-mpls] lsp-trigger all
[SwitchC-mpls] quit
```

Execute **display mpls ldp lsp** on each switch to view the LDP LSP information. Take Switch A as an example:

```
<SwitchA> display mpls ldp lsp
                              LDP LSP Information
 ------------------------------------------------------------------
 SN  DestAddress/Mask   In/OutLabel   Next-Hop     In/Out-Interface
 ------------------------------------------------------------------
 1    1.1.1.9/32         3/NULL       127.0.0.1     -------/InLoop0
 2    2.2.2.9/32         NULL/3       10.1.1.2      -------/Vlan1
 3    3.3.3.9/32         NULL/1024    10.1.1.2      -------/Vlan1
 4    20.1.1.0/24        NULL/3       10.1.1.2      -------/Vlan1
 ------------------------------------------------------------------
 A '*' before an LSP means the LSP is not established
 A '*' before a Label means the USCB or DSCB is stale
```

On Switch A, check the validity and connectivity of the LDP LSP from Switch A to Switch C.

```
[SwitchA] ping lsp ipv4 3.3.3.9 32
LSP Ping FEC: LDP IPV4 PREFIX 3.3.3.9/32 : 100  data bytes, press CTRL_C to break
 Reply from 20.1.1.2: bytes=100 Sequence=1 time = 75 ms
 Reply from 20.1.1.2: bytes=100 Sequence=2 time = 75 ms
 Reply from 20.1.1.2: bytes=100 Sequence=3 time = 74 ms
 Reply from 20.1.1.2: bytes=100 Sequence=4 time = 75 ms
 Reply from 20.1.1.2: bytes=100 Sequence=5 time = 74 ms
 --- FEC: LDP IPV4 PREFIX 3.3.3.9/32 ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 74/74/75 ms
```

On Switch C, check the validity and connectivity of the LDP LSP from Switch C to Switch A.

```
[SwitchC] ping lsp ipv4 1.1.1.9 32
 LSP Ping FEC: LDP IPV4 PREFIX 1.1.1.9/32 : 100  data bytes, press CTRL_C to break
 Reply from 10.1.1.1: bytes=100 Sequence=1 time = 75 ms
 Reply from 10.1.1.1: bytes=100 Sequence=2 time = 75 ms
```

```
Reply from 10.1.1.1: bytes=100 Sequence=3 time = 74 ms
Reply from 10.1.1.1: bytes=100 Sequence=4 time = 74 ms
Reply from 10.1.1.1: bytes=100 Sequence=5 time = 74 ms

--- FEC: LDP IPV4 PREFIX 1.1.1.9/32 ping statistics ---
5 packet(s) transmitted
5 packet(s) received
0.00% packet loss
round-trip min/avg/max = 74/74/75 ms
```

Configuring BFD for LSP validity check

Network requirements

See Figure 25 for the network diagram. Make sure that two LDP LSPs with opposite directions are established between Switch A and Switch C. Use BFD to check the connectivity of the LSPs.

Procedure

1. Configure LDP sessions. For details, see "Configuring LDP to establish LSPs dynamically."
2. Enable BFD for LSP validity check

Configure Switch A.

```
<SwitchA> system-view
[SwitchA] mpls lspv
[SwitchA -mpls-lspv] bfd enable 3.3.3.9 32
[SwitchA -mpls-lspv] quit
```

Configure Switch C.

```
<SwitchC> system-view
[SwitchC] mpls lspv
[SwitchC-mpls-lspv] bfd enable 1.1.1.9 32
[SwitchC-mpls-lspv] quit
```

3. Verify the configurations

Execute **display mpls lsp bfd** on Switch A and Switch C respectively to view information about the sessions established for the LSPs. Take Switch A as an example:

```
[SwitchA] display mpls lsp bfd

              MPLS BFD Session(s) Information
--------------------------------------------------------------------------------
FEC            : 3.3.3.9/32        Type           : LSP
Local Discr    : 11                Remote Discr   : 11
Tunnel ID      : 0xd2006           NextHop        : 10.1.1.2
Session State  : Up                Source IP      : 1.1.1.9
Session Role   : Active

FEC            : 1.1.1.9/32        Type           : LSP
Local Discr    : 12                Remote Discr   : 12
Tunnel ID      : ---               NextHop        : ---
Session State  : Up                Source IP      : 3.3.3.9
```

```
 Session Role   : Passive

 Total Session Num: 2
```

The output indicates that two BFD sessions have been established between Switch A and Switch C, one for detecting the connectivity of the LSP Switch A-Switch B-Switch C, and the other for detecting the connectivity of the LSP Switch C-Switch B-Switch A.

You can use the following command to view the verbose information of the BFD sessions.

```
[SwitchA] display bfd session verbose

 Total Session Num: 2            Init Mode: Active

 Session Working Under Ctrl Mode:

     Local Discr: 11                 Remote Discr: 11
       Source IP: 1.1.1.9          Destination IP: 127.0.0.1
   Session State: Up                     Interface: LoopBack0
 Min Trans Inter: 400ms           Act Trans Inter: 400ms
  Min Recv Inter: 400ms          Act Detect Inter: 2000ms
  Running Up for: 00:01:06              Auth mode: None
    Connect Type: Indirect              Board Num: 6
        Protocol: MFW/LSPV
       Diag Info: No Diagnostic

     Local Discr: 12                 Remote Discr: 12
       Source IP: 1.1.1.9          Destination IP: 3.3.3.9
   Session State: Up                     Interface: LoopBack0
 Min Trans Inter: 400ms           Act Trans Inter: 400ms
  Min Recv Inter: 400ms          Act Detect Inter: 2000ms
  Running Up for: 00:00:59              Auth mode: None
    Connect Type: Indirect              Board Num: 6
        Protocol: MFW/LSPV
       Diag Info: No Diagnostic
```

Configuring MPLS TE

The term router in this document refers to both routers and Layer 3 switches.

For information about VPN, see "MPLS L2VPN configuration" and "MPLS L3VPN configuration."

The Layer 3 Ethernet interface refers to the Ethernet port that can perform IP routing and inter-VLAN routing. You can set an Ethernet port as a Layer 3 Ethernet interface by using **port link-mode route** (see *Layer 2—LAN Switching Configuration Guide*).

The A5820X switch series do not support MPLS TE.

Traffic engineering and MPLS TE

Traffic engineering

1. TE function

Network congestion is one of the major problems that can degrade your network backbone performance. It may occur either when network resources are inadequate or when load distribution is unbalanced. TE is intended to avoid the latter situation where partial congestion may occur as the result of inefficient resource allocation.

TE can make best utilization of network resources and avoid non-even load distribution by real-time monitoring traffic and traffic load on each network elements to dynamically tune traffic management attributes, routing parameters and resources constraints.

The performance objectives for TE can be classified as either traffic-oriented or resource-oriented.

- Traffic–oriented performance objectives enhance QoS of traffic streams, such as minimization of packet loss, minimization of delay, maximization of throughput and enforcement of SLA.
- Resource–oriented performance objectives optimize resources utilization. Bandwidth is a crucial resource on networks. Efficiently managing it is one major task of TE.

2. TE solution

As existing IGPs are topology-driven and consider only network connectivity, they fail to present some dynamic factors such as bandwidth and traffic characteristics.

This IGP disadvantage can be repaired by using an overlay model, such as IP over ATM or IP over FR. An overlay model provides a virtual topology above the physical network topology for a more scalable network design. It also provides better traffic and resources control support for implementing a variety of traffic engineering policies.

Despite all of the benefits, overlay models are not suitable for implementing traffic engineering in large-sized backbones because of their inadequacy in extensibility. In this sense, MPLS TE is a better traffic engineering solution for its extensibility and ease of implementation.

MPLS TE

MPLS is better than IGPs in implementing traffic engineering for the following reasons:

- MPLS supports explicit LSP routing.
- LSP routing is easy to manage and maintain compared with traditional packet-by-packet IP forwarding.

- CR-LDP is suitable for implementing a variety of traffic engineering policies.
- MPLS TE uses less system resources compared with other traffic engineering implementations.

MPLS TE combines the MPLS technology and traffic engineering. It delivers these benefits:

- Reserve resources by establishing LSP tunnels to specific destinations. This allows traffic to bypass congested nodes to achieve appropriate load distribution.
- When network resources are insufficient, MPLS TE allows bandwidth-hungry LSPs or critical user traffic to occupy the bandwidth for lower priority LSP tunnels.
- In case an LSP tunnel fails or congestion occurs on a network node, MPLS TE can provide route backup and FRR.

With MPLS TE, a network administrator can eliminate network congestion simply by creating some LSPs and congestion bypass nodes. Special offline tools are also available for the traffic analysis performed when the number of LSPs is large.

Basic concepts of MPLS TE

LSP tunnel

On an LSP, after packets are labeled at the ingress node, the packets are forwarded based on label. The traffic is transparent to the transits nodes on the LSP. In this sense, an LSP can be regarded as a tunnel.

MPLS TE tunnel

Reroute and transmission over multiple paths may involve multiple LSP tunnels. A set of such LSP tunnels is called a TE tunnel.

MPLS TE implementation

MPLS TE mainly accomplishes two functions:

- Static CR-LSP processing to create and remove static CR-LSPs. The bandwidth of a static CR-LSP must be configured manually.
- Dynamic CR-LSP processing to handle three types of CR-LSPs: basic CR-LSPs, backup CR-LSPs and fast rerouted CR-LSPs.

Static CR-LSP processing is simple. Dynamic CR-LSP processing involves four phrases: advertising TE attributes, calculating paths, establishing paths, and forwarding packets.

Advertising TE attributes

MPLS TE must be aware of dynamic TE attributes of each link on the network. This is achieved by extending link state-based IGPs such as OSPF and IS-IS.

OSPF and IS-IS extensions add to link states such TE attributes as link bandwidth, color, among which maximum reservable link bandwidth and non-reserved bandwidth with a particular priority are most important.

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 TEDB.

Calculating paths

Link state-based routing protocols use SPF to calculate the shortest path to each network node.

In MPLS TE, the CSPF algorithm calculates the shortest, TE compliant path to a node. It is derived from SPF and makes calculation based on two conditions:

- Constraints on the LSP to be set up with respect to bandwidth, color, preemption/holding priority, explicit path and other constraints. They are configured at the LSP ingress.
- TEDB

CSPF first prunes TE attribute incompliant links from the TEDB and then performs SPF calculation to identify the shortest path to an LSP egress.

Establishing paths

When setting up LSP tunnels, you may use CR-LDP or RSVP-TE as the signaling protocol. Both can carry constraints such as LSP bandwidth, some explicit route information, and color and deliver the same function.

CR-LDP establishes LSPs by using TCP. RSVP-TE establishes LSPs by using raw IP.

RSVP is a well-established technology in terms of its architecture, protocol procedures and support to services. CR-LDP is an emerging technology with better scalability.

Both CR-LDP and RSVP-TE are supported on your device.

Forwarding packets

Packets are forwarded over established tunnels.

CR-LSP

Unlike ordinary LSPs established based on routing information, CR-LSPs are established based on criteria such as bandwidth, selected path, and QoS parameters in addition to routing information.

The mechanism setting up and managing constraints is called “Constraint-based Routing” (CR).

CR-LSP involves these concepts:

- Strict and loose explicit routes
- Traffic characteristics
- Preemption
- Route pinning
- Administrative group and affinity attribute
- Reoptimization

Strict and loose explicit routes

An LSP is called a strict explicit route if all LSRs along the LSP are specified.

An LSP is called a loose explicit route if the downstream LSR selection conditions rather than LSRs are defined.

Traffic characteristics

Traffic is described in terms of peak rate, committed rate, and service granularity.

The peak and committed rates describe the bandwidth constraints of a path. The service granularity specifies a constraint on the delay variation that the CR-LDP MPLS domain may introduce to a path's traffic.

Preemption

CR-LDP signals the resources required by a path on each hop of the route. If a route with sufficient resources cannot be found, existing paths may be rerouted to reallocate resources to the new path. This is called path preemption.

Two priorities, setup priority and holding priority, are assigned to paths for making preemption decision. Both setup and holding priorities range from 0 to 7, with a lower numerical number indicating a higher priority.

For a new path to preempt an existing path, the setup priority of the new path must be greater than the holding priority of the existing path. To initiate a preemption, the Resv message of RSVP-TE is sent.

To avoid flapping caused by improper preemptions between CR-LSPs, the setup priority of a CR-LSP should not be set higher than its holding priority.

Route pinning

Route pinning prevents an established CR-LSP from changing with route changes.

If a network does not run IGP TE extension, the network administrator are unable to identify from which part of the network the required bandwidth should be obtained when setting up a CR-LSP. In this case, loose explicit route (ER-hop) with required resources is used. The CR-LSP thus established however, may change when the route changes, for example, when a better next hop becomes available. If this is undesirable, the network administrator can set up the CR-LSP using route underpinning to make it a permanent path.

Administrative group and affinity attribute

The affinity attribute of an MPLS TE tunnel identifies the properties of the links that the tunnel can use. Together with the link administrative group, it decides which links the MPLS TE tunnel can use.

Reoptimization

Traffic engineering is a process of allocating/reallocating network resources. You may configure it to meet desired QoS.

Normally, service providers use some mechanism to optimize CR-LSPs for best use of network resources. They can do this manually but CR-LSP measurement and tuning are required. Alternatively, they can use MPLS TE where CR-LSPs are dynamically optimized.

Dynamic CR-LSP optimization involves periodic calculation of paths that traffic trunks should traverse. If a better route is found for an existing CR-LSP, a new CR-LSP is established to replace the old one, and services are switched to the new CR-LSP.

RSVP-TE

Two QoS models are available: IntServ and DiffServ.

RSVP is designed for IntServ. It reserves resources on each node along a path. RSVP operates at the transport layer but does not participate in data transmission. It is an Internet control protocol similar to ICMP.

The following are features of RSVP:

- Unidirectional
- Receiver oriented. The receiver initiates resource reservation requests and is responsible for maintaining the reservation information.
- Using soft state mechanism to maintain resource reservation information.

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", which is operating as a signaling protocol for LSP tunnel setup in MPLS TE.

Basic concepts of RSVP-TE

1. Soft state

Soft state is a mechanism used in RSVP-TE to periodically refresh the resource reservation state on a node. The resource reservation state includes the path state and the reservation state. The path state is generated and refreshed by the Path message, and the reservation state is generated and refreshed by the Resv message. A state is to be removed if no refresh messages are received for it in certain interval.

2. Resource reservation style

Each LSP set up using RSVP-TE is assigned a resource reservation style. During an RSVP session, the receiver decides which reservation style can be used for this session and which LSPs can be used.

Two reservation styles are available:

- FF style where resources are reserved for individual senders and cannot be shared among senders on the same session.
- SE style where resources are reserved for senders on the same session and shared among them.

SE is only used for make-before-break because multiple LSPs cannot be present on the same session.

Make-before-break

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

Figure 26 Diagram for make-before-break

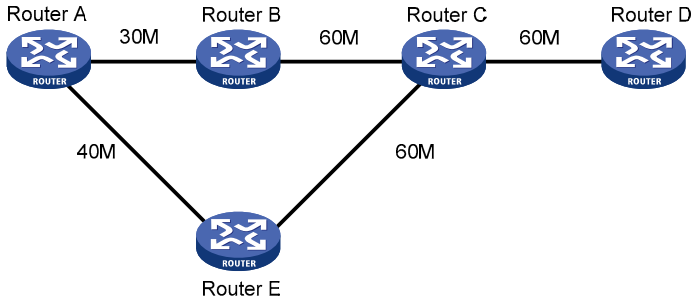


Figure 26 presents a scenario where a path Router A →Router B →Router C →Router D is established with 30 Mbps reserved bandwidth between Router A and Router D. The remaining bandwidth is then 30 Mbps.

If 40 Mbps path bandwidth is requested, the remaining bandwidth of the Router A →Router B →Router C →Router D path is inadequate. The problem cannot be addressed by selecting another path, Router A → Router E →Router C →Router D, because the bandwidth of the Router C →Router D link is inadequate.

To address the problem, you may use the make-before-break mechanism. It allows the new path to share the bandwidth of the original path at the Router C →Router D link. Upon creation of the new path, traffic is switched to the new path and the previous path is torn down. This helps avoid traffic interruption effectively.

RSVP-TE messages

RSVP-TE uses RSVP messages with extensions. The following are RSVP messages:

- Path messages—transmitted along the path of data transmission downstream by each RSVP sender to save path state information on each node along the path.
- Resv messages—sent by each receiver upstream towards senders to request resource reservation and to create and maintain reservation state on each node along the reverse of data transmission path.
- PathTear messages—sent downstream immediately once created to remove the path state and related reservation state on each node along the path.
- ResvTear messages—sent upstream immediately once created to remove the reservation state on each node along the path.
- PathErr messages—sent upstream to report Path message processing errors to senders. They do not affect the state of the nodes along the path.
- ResvErr messages—sent downstream to notify the downstream nodes that error occurs during Resv message processing or reservation error occurs as the result of preemption.
- ResvConf messages—sent to receivers to confirm Resv messages.
- Hello messages—sent between any two directly connected RSVP neighbors to set up and maintain the neighbor relationship that has local significance on the link.

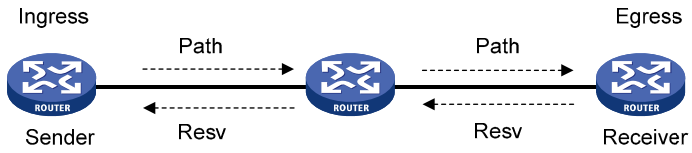
The TE extension to RSVP adds new objects to the Path message and the Resv message. These objects carry not only label bindings but also routing constraints, supporting CR-LSP and FRR.

- New objects added to the Path message include LABEL_REQUEST, EXPLICIT_ROUTE, RECORD_ROUTE, and SESSION_ATTRIBUTE.
- New objects added to the Resv message include LABEL and RECORD_ROUTE

The LABEL_REQUEST object in the Path message requests the label bindings for an LSP. It is also saved in the path state block. The node receiving LABEL_REQUEST advertises the label binding using the LABEL object in the Resv message to the upstream node, thus accomplishing label advertisement and transmission.

Setting up an LSP tunnel

Figure 27 Set up an LSP tunnel



Here is a simplified procedure for setting up an LSP tunnel with RSVP:

1. The ingress LSR sends a Path message that carries the label request information, 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 generates a Resv message carrying the reservation information and label and then forwards the message towards the ingress along the reverse direction of the path along which the Path message travels. The LSRs that the Resv message traverses along the path reserve resources as required.
3. When the ingress LSR receives the Resv message, LSP is established.

As resources are reserved on the LSRs along the path for the LSP established using RSVP-TE, services transmitted on the LSP are guaranteed.

RSVP refresh mechanism

RSVP maintains path and reservation state by periodically retransmitting two types of messages: Path and Resv. These periodically retransmitted Path and Resv messages are called refresh messages. They are sent along the path that the last Path or Resv message travels to synchronize the state between RSVP neighbors and recover lost RSVP messages.

When many RSVP sessions are present, periodically sent refresh messages become a network burden. In addition, for some delay sensitive applications, the refreshing delay they must wait for recovering lost RSVP messages may be unbearable. As tuning refresh intervals is not adequate to address the two problems, the refreshing mechanism was extended in *RFC 2961 RSVP Refresh Overhead Reduction Extensions* to address the problems:

1. Message_ID extension

RSVP itself uses Raw IP to send messages. The Message_ID extension mechanism defined in RFC 2961 adds objects that can be carried in RSVP messages. Of them, the Message_ID object and the Message_ID_ACK object are used to acknowledge RSVP messages, thus improving transmission reliability.

On an interface enabled with the Message_ID mechanism, you may configure RSVP message retransmission. After the interface sends an RSVP message, it waits for acknowledgement. If no ACK is received before the initial retransmission interval (Rf seconds for example) expires, the interface resends the message. After that, the interface resends the message at an exponentially increased retransmission interval equivalent to (1 + Delta) × Rf seconds.

2. Summary refresh extension

Send summary refreshes (Srefreshes) rather than retransmit standard Path or Resv messages to refresh related RSVP state. This reduces refresh traffic and allows nodes to make faster processing.

To use summary refresh, you must use the Message_ID extension. Only states advertised using MESSAGE_ID included Path and Resv messages can be refreshed using summary refreshes.

PSB, RSB and BSB timeouts

To create an LSP tunnel, the sender sends a Path message with a LABEL_REQUEST object. After receiving this Path message, the receiver assigns a label for the path and puts the label binding in the LABEL object in the returned Resv message.

The LABEL_REQUEST object is stored in the PSB on the upstream nodes. The LABEL object is stored in the RSB on the downstream nodes. The state stored in the PSB or RSB object times out and is removed after the number of consecutive non-refreshing times exceeds the PSB or RSB timeout keep-multiplier.

You may sometimes want to store the resource reservation state for a reservation request that does not pass the admission control on some node. This however should not prevent the resources reserved for the request from being used by other requests. To handle this situation, the node transits to the blockade state and a BSB is created on each downstream node. When the number of non-refreshing times exceeds the blockade multiplier, the state in the BSB is removed.

RSVP-TE GR

The RSVP-TE GR function depends on the extended hello capability of RSVP-TE. A GR-capable device advertises its GR capability and relevant time parameters to its neighbors by extended RSVP Hello packets. If a device and all its neighbors have the RSVP GR capability and have exchanged GR

parameters, each of them can function as the GR helper of another device, allowing data to be forwarded without interruption when the GR restarter is rebooting.

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

If a GR helper and the GR restarter reestablish a Hello session before the restart timer expires, the recovery timer is started and signaling packet exchanging is triggered to restore the original soft state. Otherwise, all RSVP soft state information and forwarding entries relevant to the neighbor is removed. If the recovery timer expires, soft state information and forwarding entries that are not restored during the GR restarting process is removed.

A5800 series can act as a GR restarter and a GR helper at the same time.

Traffic forwarding

For traffic to travel along an LSP tunnel, you must make configuration after creating the MPLS TE tunnel. Otherwise, traffic is IP routed.

Even when an MPLS TE tunnel is available, traffic is IP routed if you do not configure it to travel the tunnel. For traffic to be routed along an MPLS TE tunnel, you can use static routing, policy routing, or automatic route advertisement.

Static routing

Static routing is the easiest way to route traffic along an MPLS TE tunnel. You only need to manually create a route that reaches the destination through the tunnel interface.

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

Automatic route advertisement

You can use automatic route advertisement to advertise MPLS TE tunnel interface routes to IGPs, allowing traffic to be routed down MPLS TE tunnels.

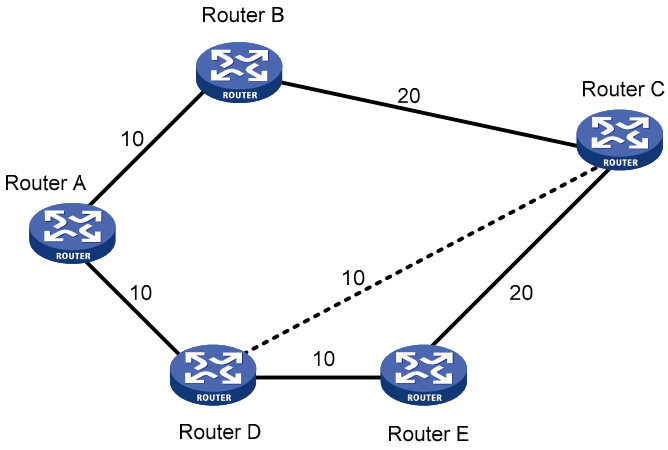
Two approaches are available to automatic route advertisement: IGP shortcut and forwarding adjacency.

OSPF and IS-IS support both approaches where TE tunnels are considered point-to-point links and TE tunnel interfaces can be set as outgoing interfaces.

IGP shortcut, also known as “autoroute announce”, considers a TE tunnel as a logical interface directly connected to the destination when computing IGP routes on the ingress of the TE tunnel.

IGP shortcut and forwarding adjacency are different in that in the forwarding adjacency approach, routes with TE tunnel interfaces as outgoing interfaces are advertised to neighboring devices but not in the IGP shortcut approach. Therefore, TE tunnels are visible to other devices in the forwarding adjacency approach but not in the IGP shortcut approach.

Figure 28 IGP shortcut and forwarding adjacency



As shown in Figure 28, a TE tunnel is present between Router D and Router C. With IGP shortcut enabled, the ingress node Router D can use this tunnel when calculating IGP routes. This tunnel, however, is invisible to Router A; therefore, Router A cannot use this tunnel to reach Router C. With forwarding adjacency enabled, Router A can know the presence of the TE tunnel and thus forward traffic to Router C to Router D though this tunnel.

The configuration of IGP shortcut and forwarding adjacency is broken down into tunnel configuration and IGP configuration. When making tunnel configuration on a TE tunnel interface, take the following issues into consideration:

- The tunnel destination address should be in the same area where the tunnel interface is located.
- The tunnel destination address should be reachable through intra-area routing.

CR-LSP backup

CR-LSP backup provides end-to-end path protection for the entire LSP without time limitation. This is different from FRR which provides quick but temporary per-link or per-node protection on an LSP.

In the same TE tunnel, the LSP that backs up a primary LSP is called a secondary LSP. When the ingress of a TE tunnel detects that the primary LSP is unavailable, it switches traffic to the secondary LSP and after the primary LSP becomes available, switches traffic back. This is how LSP path protection is achieved.

Two approaches are available for CR-LSP backup:

- Hot backup where a secondary CR-LSP is created immediately after a primary CR-LSP is created. MPLS TE switches traffic to the secondary CR-LSP after the primary CR-LSP fails.
- Standard backup where a secondary CR-LSP is created to take over after the primary CR-LSP fails.

Fast reroute

FRR provides a quick per-link or per-node protection on an LSP.

In this approach, once a link or node fails on a path, FRR comes up to reroute the path to a new link or node to bypass the failed link or node. This can happen as fast as 50 milliseconds, minimizing data loss.

Once a link or node on an LSP configured with FRR fails, traffic is switched to the protection link and the headend of the LSP starts attempting to set up a new LSP.

Basic concepts

The following are concepts that FRR involves throughout this document:

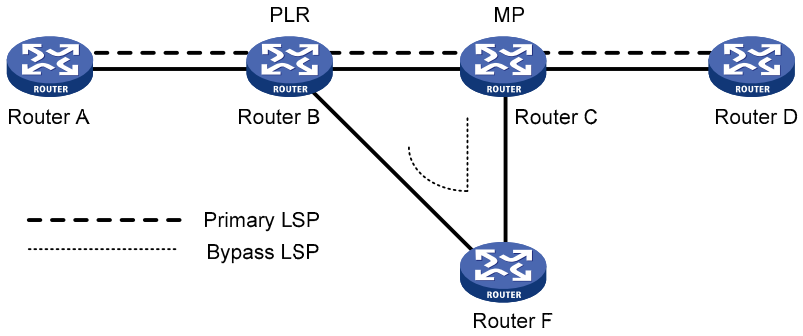
- **Primary LSP**—The protected LSP.
- **Bypass LSP**—An LSP that protects the primary LSP.
- **PLR**—The ingress of the bypass LSP. It must be located on the primary LSP but must not be the egress.
- **MP**—The egress of the bypass LSP. It must be located on the primary LSP but must not be the ingress.

Protection

FRR provides link protection and node protection for an LSP.

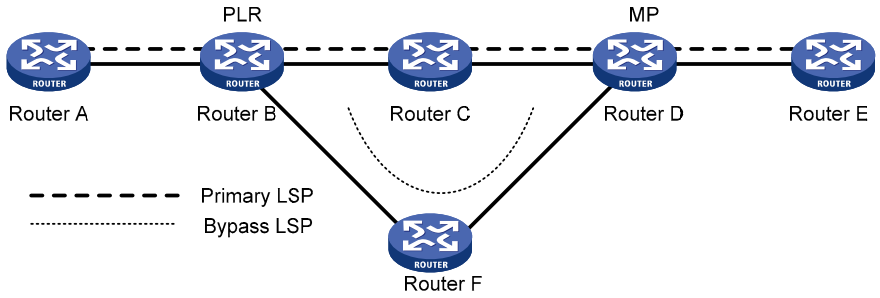
- Link protection, where the PLR and the MP are connected through a direct link and the primary LSP traverses this link. When the link fails, traffic is switched to the bypass LSP. As shown in Figure 29, the primary LSP is Router A →Router B →Router C →Router D, and the bypass LSP is Router B → Router F →Router C.

Figure 29 FRR link protection



- Node protection, where the PLR and the MP are connected through a device and the primary LSP traverses this device. When the device fails, traffic is switched to the bypass LSP. As shown in Figure 30, the primary LSP is Router A →Router B →Router C →Router D →Router E, and the bypass LSP is Router B →Router F→Router D. Router C is the protected device.

Figure 30 FRR node protection



Deploying FRR

When configuring the bypass LSP, make sure the protected link or node is not on the bypass LSP.

As bypass LSPs are pre-established, FRR requires extra bandwidth. When network bandwidth is insufficient, use FRR only for crucial interfaces or links.

Link status detetion methods

FRR can detect the failure of a link timely and reroute traffic to the bypass LSP. It detects the status of a link in one of the following three methods:

- **Link layer protocol status detection**—In this method, the interface type determines how fast the FRR can detect a link failure.
- **Cooperation of RSVP-TE and BFD**—BFD is a fast detection mechanism, which can detect faults of links or nodes timely. In this method, FRR can obtain the link status timely through BFD, so as to implement fast switchover of links.
- **RSVP hello**—In this method, RSVP hello is enabled on each protected node and its neighbor nodes along the primary LSP, so that a device periodically sends hello messages to its peer device on the LSP. If a link or node fails, hello messages are lost. If a device cannot receive hello messages from its peer in three hello intervals, the hello mechanism concludes that a link failure occurs. Hence, the hello mechanism is slower in link failure detection, compared with the other two methods.

Protocols and standards

- RFC 2702, *Requirements for Traffic Engineering Over MPLS*
- RFC 3212, *Constraint-Based LSP Setup using LDP*
- RFC 2205, *Resource ReSerVation Protocol*
- RFC 3209, *RSVP-TE: Extensions to RSVP for LSP Tunnels*
- RFC 2961, *RSVP Refresh Overhead Reduction Extensions*
- RFC 3564, *Requirements for Support of Differentiated Service-aware MPLS Traffic Engineering*

MPLS TE configuration task list

To configure MPLS TE:

Task		Remarks
Configuring MPLS TE basic capabilities		Required.
Configuring an MPLS TE tunnel	Creating MPLS TE tunnel over static CR-LSP	Required.
	Configuring MPLS TE tunnel with dynamic signaling protocol	Use either approach.
Configuring RSVP-TE advanced features		Optional.
Tuning CR-LSP setup		Optional.
Tuning MPLS TE tunnel setup		Optional.
Configuring traffic forwarding	Forwarding traffic along MPLS TE tunnels using static routes	Required. to use either approach.
	Forwarding traffic along MPLS TE tunnels through automatic route advertisement	
Configuring traffic forwarding tuning parameters		Optional.
Configuring CR-LSP backup		Optional.
Configuring FRR		Optional.

Task	Remarks
Inspecting an MPLS TE tunnel	Optional.

Configuring MPLS TE basic capabilities

MPLS TE basic capabilities are essential to MPLS TE feature configurations. After configuring the basic capabilities, you must make other configurations in order to use MPLS TE depending on the actual requirements.

Prerequisites

Before you configure MPLS TE basic capabilities, complete the following tasks:

- Configure static routing or IGPs to make sure all LSRs are reachable.
- Configure MPLS basic capabilities.

For configuration information about MPLS basic capability, see "MPLS basics configuration."

Procedure

To configure MPLS TE basic capabilities:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Enable global MPLS TE.	**mpls te**	Required. Disabled by default.
4. Return to system view.	**quit**	—
5. Enter the interface view of an MPLS TE link.	**interface** *interface-type interface-number*	—
6. Enable interface MPLS TE.	**mpls te**	Required. Disabled by default.
7. Return to system view.	**quit**	—
8. Create a tunnel interface and enter its view.	**interface tunnel** *tunnel-number*	Required.
9. Assign an IP address to the tunnel interface.	**ip address** *ip-address netmask*	Optional.
10. Set the tunnel protocol to MPLS TE.	**tunnel-protocol mpls te**	Required.
11. Configure the destination address of the tunnel.	**destination** *ip-address*	Required.
12. Configure the tunnel ID of the tunnel.	**mpls te tunnel-id** *tunnel-id*	Required.
Submit the current tunnel configuration.	**mpls te commit**	Required.

For information about tunnel interfaces, see *Layer 3—IP Services Configuration Guide*.

Creating MPLS TE tunnel over static CR-LSP

Creating MPLS TE tunnels over static CR-LSPs does not involve configuration of tunnel constraints or the issue of IGP TE extension or CSPF. What you must do is to create a static CR-LSP and a TE tunnel using static signaling and then associate them.

Despite its ease of configuration, the application of MPLS TE tunnels over static CR-LSPs is restricted because they cannot dynamically adapt to network changes.

Static CR-LSPs are special static LSPs. They share the same constraints and use the same label space.

Prerequisites

Before you configure an MPLS TE tunnel over a static CR-LSP, complete the following tasks:

- Configure static routing or an IGP protocol to make sure that all LSRs are reachable.
- Configure MPLS basic capabilities.
- Configure MPLS TE basic capabilities.

Procedure

To create an MPLS TE tunnel over a CR-LSP:

Step		Command	Remarks
1. Enter system view.		**system-view**	—
2. Enter the interface view of an MPLS TE tunnel.		**interface tunnel** *tunnel-number*	—
3. Configure the tunnel to use static CR-LSP.		**mpls te signal-protocol static**	Required.
4. Submit the current tunnel configuration.		**mpls te commit**	Required.
5. Exit to system view.		**quit**	—
6. Create a static CR-LSP on your device depending on its location in the network.	At the ingress	**static-cr-lsp ingress** *tunnel-name* **destination** *dest-addr* **nexthop** *next-hop-addr* **out-label** *out-label-value*	Required. Use any of the commands depending on the location of your device in the network.
	On the transit node	**static-cr-lsp transit** tunnel-name **incoming-interface** interface-type interface-number **in-label** in-label-value **nexthop** next-hop-addr **out-label** out-label-value	
	At the egress	**static-cr-lsp egress** tunnel-name **incoming-interface** interface-type interface-number **in-label** in-label-value	

The *tunnel-name* argument specifies the name of the MPLS TE tunnel carried over the static CR-LSP.

The *tunnel-name* argument in **static-cr-lsp ingress** is case sensitive. Suppose you create a tunnel interface with **interface tunnel 2**. To specify it for the *tunnel-name* in **static-cr-lsp ingress**, you must enter its name in the form of Tunnel2. Otherwise, your tunnel establishment attempt fails. This restriction however does not apply to transit and egress nodes.

The next hop address cannot be a local public address when configuring the static CR-LSP on the ingress or a transit node.

Configuring MPLS TE tunnel with dynamic signaling protocol

Dynamic signaling protocol can adapt the path of a TE tunnel to network changes and implement redundancy, FRR, and other advanced features.

The following describes how to create an MPLS TE tunnel with a dynamic signaling protocol:

- Configure MPLS TE properties for links and advertise them through IGP TE extension to form a TEDB.
- Configure tunnel constraints.
- Use the CSPF algorithm to calculate a preferred path based on the TEDB and tunnel constraints.
- Establish the path by using the signaling protocol RSVP-TE or CR-LDP.

To form a TEDB, you must configure the IGP TE extension for the nodes on the network to send TE LSAs. If the IGP TE extension is not configured, the CR-LSP is created based on IGP routing rather than computed by CSPF.

Prerequisites

Before you configure an MPLS TE tunnel by using a dynamic signaling protocol, complete the following tasks:

- Configure static routing or an IGP protocol to make sure that all LSRs are reachable.
- Configure MPLS basic capabilities.
- Configure MPLS TE basic capabilities.

Procedure

Complete the following tasks to configure an MPLS TE tunnel by using a dynamic signaling protocol:

Task	Remarks
Configuring CSPF	Optional.
Configuring OSPF TE	Required when CSPF is configured. Choose one depending on the IGP protocol used.
Configuring IS-IS TE	
Configuring an MPLS TE explicit path	Optional.
Configuring MPLS TE tunnel constraints	Optional.
Establishing an MPLS TE tunnel with RSVP-TE	Optional. By default, RSVP-TE is used for establishing an MPLS TE tunnel.

Configuring CSPF

To configure CSPF:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view	**mpls**	—
3. Enable CSPF on your device	**mpls te cspf**	Required. Disabled by default

Configuring OSPF TE

Configure OSPF TE if the routing protocol is OSPF and a dynamic signaling protocol is used for MPLS TE tunnel setup.

The OSPF TE extension uses Opaque Type 10 LSAs to carry TE attributes of links. Before configuring OSPF TE, you must enable the opaque capability of OSPF. In addition, for TE LSAs to be generated, at least one neighbor must be in full state.

To configure OSPF TE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter OSPF view.	**ospf** [*process-id*]	—
3. Enable the opaque LSA capability.	**opaque-capability enable**	Required. Disabled by default.
4. Enter OSPF area view.	**area** *area-id*	Required.
5. Enable MPLS TE in the OSPF area.	**mpls-te enable**	Required. Disabled by default.
6. Exit to OSPF view.	**quit**	—

For more information about OSPF opaque LSA, see *Layer 3—IP Routing Configuration Guide*.

MPLS TE cannot reserve resources and distribute labels on OSPF virtual links. MPLS TE cannot establish an LSP tunnel through an OSPF virtual link. Make sure no virtual links exist in the OSPF routing domain when configuring OSPF TE.

Configuring IS-IS TE

CAUTION:

- According to RFC 3784, the length of the IS reachability TLV (type 22) may reach the maximum of 255 octets in some cases.
- For an IS-IS LSP to carry this type of TLV and to be flooded normally on all interfaces with IS-IS enabled, the MTU of any IS-IS enabled interface, including 27 octets of LSP header and two octets of TLV header, cannot be less than 284 octets. If an LSP must also carry the authentication information, the minimum MTU needs to be recalculated according to the packet structure.

When TE is configured, HP recommends that you set the MTU of any interface with IS-IS enabled be equal to or greater than 512 octets to guarantee that IS-IS LSPs can be flooded on the network.

Configure IS-IS TE if the routing protocol is IS-IS and a dynamic signaling protocol is used for MPLS TE tunnel setup. In case both OSPF TE and IS-IS TE are available, OSPF TE takes priority.

The IS-IS TE extension uses the sub-TLV of IS reachability TLV (type 22) to carry TE attributes. Before configuring IS-IS TE, configure the IS-IS wide metric style, which can be wide, compatible, or wide-compatible.

To configure IS-IS TE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter IS-IS view.	**isis** [*process-id*]	—
3. Configure the wide metric attribute of IS-IS.	**cost-style** { **narrow** | **wide** | **wide-compatible** | { **compatible** | **narrow-compatible** } [**relax-spf-limit**] }	Required. By default, IS-IS uses narrow metric style.
4. Enable IS-IS TE.	**traffic-eng** [**level-1** | **level-2** | **level-1-2**]	Required. Disabled by default.
5. Configure the TLV type of the sub-TLV carrying DS-TE parameters.	**te-set-subtlv** { **bw-constraint** *value* | **lo-multiplier** *value* | **unreserved-bw-sub-pool** *value* }	Optional. By default, the **bw-constraint** parameter is carried in sub-TLV 252; the **lo-multiplier** parameter in sub-TLV 253; and the **unreserved-bw-sub-pool** parameter in sub-TLV 251.

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

IS-IS TE does not support secondary IP address advertisement. With IS-IS TE enabled on an interface configured with multiple IP addresses, IS-IS TE advertises only the primary IP address of the interface through the sub-TLV of IS reachability TLV (type 22). HP does not recommend enabling IS-IS TE on an interface configured with secondary IP addresses.

Configuring an MPLS TE explicit path

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

- **Strict**—The two nodes are directly connected.
- **Loose**—The two nodes have devices in between.

When inserting nodes to an explicit path or modifying nodes on it, you may configure the **include** keyword to have the established LSP traverse the specified nodes or the **exclude** keyword to have the established LSP bypass the specified nodes.

To configure an MPLS TE explicit path:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create an explicit path for MPLS TE tunneling and enter its view.	**explicit-path** *path-name* [**disable** | **enable**]	Required.

Step	Command	Remarks
3. Add a node to the explicit path.	**add hop** *ip-address1* [**include** [**loose** | **strict**] | **exclude**] { **after** | **before** } *ip-address2*	Optional. By default, the **include** keyword and the **strict** keyword apply. The explicit path traverses the specified node and the next node is a strict node.
4. Specify a next hop IP address on the explicit path.	**next hop** *ip-address* [**include** [**loose** | **strict**] | **exclude**]	Required. The next hop is a strict node by default. Repeat this step to define a sequential set of the hops that the explicit path traverses.
5. Modify the IP address of current node on the explicit path.	**modify hop** *ip-address1 ip-address2* [[**include** [**loose** | **strict**] | **exclude**]	Optional. By default, the **include** keyword and the **strict** keyword apply. The explicit path traverses the specified node and the next node is a strict node.
6. Remove a node from the explicit path.	**delete hop** *ip-address*	Optional.
7. Display information about the specified or all nodes on the explicit path.	**list hop** [*ip-address*]	Optional.

When establishing an MPLS TE tunnel between areas or ASs, you must use a loose explicit route, specify the ABR or ASBR as the next hop of the route, and make sure that a route is available between the ABRs or ASBRs.

Configuring MPLS TE tunnel constraints

To configure MPLS TE tunnel constraints:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Specify a path for the tunnel to use and set the preference of the path.	**mpls te path** { **dynamic** | **explicit-path** *pathname* } **preference** *value*	Optional. By default, a tunnel uses the dynamically calculated path.
4. Submit current tunnel configuration.	**mpls te commit**	Required.

Establishing an MPLS TE tunnel with RSVP-TE

CAUTION:

To use RSVP-TE as the signaling protocol for setting up the MPLS TE tunnel, you must enable both MPLS TE and RSVP-TE on the interface for the tunnel to use.

To establish an MPLS TE tunnel with RSVP-TE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Enable RSVP-TE on your device.	**mpls rsvp-te**	Required. Disabled by default.
4. Exit to system view.	**quit**	—
5. Enter interface view of MPLS TE link.	**interface** *interface-type interface-number*	—
6. Enable RSVP-TE on the interface.	**mpls rsvp-te**	Required. Disabled by default.
7. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
8. Set the signaling protocol for setting up the MPLS TE tunnel to RSVP-TE.	**mpls te signal-protocol rsvp-te**	Optional. RSVP-TE applies by default.
9. Submit current tunnel configuration.	**mpls te commit**	Required.

Configuring RSVP-TE advanced features

RSVP-TE adds new objects in Path and Resv messages to support CR-LSP setup. RSVP-TE provides many configurable options with respect to reliability, network resources, and other advanced features of MPLS TE.

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

Prerequisites

Before configuring RSVP-TE advanced features, complete the following tasks:

- Configure MPLS basic capabilities
- Configure MPLS TE basic capabilities
- Establish an MPLS TE tunnel with RSVP-TE

Procedure

Configuring RSVP-TE advanced features involves these tasks:

- Configuring RSVP reservation style
- Configuring RSVP state timers
- Configuring the RSVP refreshing mechanism
- Configuring the RSVP hello extension
- Configuring RSVP-TE resource reservation confirmation
- Configuring RSVP authentication

- Configuring RSVP-TE GR

Configuring RSVP reservation style

Each LSP set up using RSVP-TE is assigned a resource reservation style. During an RSVP session, the receiver decides which reservation style can be used for this session and which LSPs can be used.

Two reservation styles are available:

- **FF style**—where resources are reserved for individual senders and cannot be shared among senders on the same session.
- **SE style**—where resources are reserved for senders on the same session and shared among them.

To configure RSVP reservation style:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Configure the resources reservation style for the tunnel.	**mpls te resv-style** { **ff** | **se** }	Optional. The default resource reservation style is SE.
4. Submit current tunnel configuration.	**mpls te commit**	Required.

In current MPLS TE applications, the SE style is mainly used for make-before-break. The FF style is rarely used.

Configuring RSVP state timers

To configure RSVP state timers:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Configure the path/reservation state refresh interval of the node.	**mpls rsvp-te timer refresh** *timevalue*	Optional. The default path/reservation state refresh interval is 30 seconds.
4. Configure the keep multiplier for PSB and RSB.	**mpls rsvp-te keep-multiplier** *number*	Optional. The default is 3.
5. Configure the blockade timeout multiplier.	**mpls rsvp-te blockade-multiplier** *number*	Optional. The default blockade timeout multiplier is 4.

Configuring the RSVP refreshing mechanism

To enhance reliability of RSVP message transmission, the Message_ID extension mechanism is used to acknowledge RSVP messages. The Message_ID extension mechanism is also referred to as “the reliability mechanism” throughout this document.

After you enable RSVP message acknowledgement on an interface, you may enable retransmission.

To use Srefresh, you must use the Message_ID extension. Only states advertised using MESSAGE_ID included Path and Resv messages can be refreshed using summary refreshes.

To configure RSVP refreshing mechanism:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view of MPLS TE link.	**interface** *interface-type interface-number*	—
3. Enable the reliability mechanism of RSVP-TE.	**mpls rsvp-te reliability**	Optional.
4. Enable retransmission.	**mpls rsvp-te timer retransmission** { **increment-value** [*increment-value*] | **retransmit-value** [*retrans-timer-value*] } *	Optional. Disabled by default.
5. Enable summary refresh.	**mpls rsvp-te srefresh**	Optional. Disabled by default.

Configuring the RSVP hello extension

To configure the RSVP hello extension:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Enable global RSVP hello extension.	**mpls rsvp-te hello**	Required. Disabled by default.
4. Configure the maximum number of consecutive hellos that should be lost before the link is considered failed.	**mpls rsvp-te hello-lost** *times*	Optional. By default, the link is considered failed if three consecutive hellos are lost.
5. Configure the hello interval.	**mpls rsvp-te timer hello** *timevalue*	Optional. The default is 3 seconds.
6. Exit to system view.	**quit**	—
7. Enter interface view of MPLS TE link.	**interface** *interface-type interface-number*	—
8. Enable interface RSVP hello extension.	**mpls rsvp-te hello**	Required. Disabled by default.

RSVP hello extension detects the reachability of RSVP neighboring nodes. It is defined in RFC 3209.

Configuring RSVP-TE resource reservation confirmation

To configure RSVP-TE resource reservation confirmation:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Enable resource reservation confirmation.	**mpls rsvp-te resvconfirm**	Required. Disabled by default.

Reservation confirmation is initiated by the receiver, which sends the Resv message with an object requesting reservation confirmation.

Receiving the ResvConf message does not mean that resource reservation is established. It only indicates that resources are reserved on the farthest upstream node where the Resv message arrived and the resources can be preempted.

Configuring RSVP authentication

RSVP adopts hop-by-hop authentication to prevent fake resource reservation requests from occupying network resources.

It requires that the interfaces at the two ends of a link must share the same authentication key to exchange RSVP messages.

To configure RSVP authentication:

Step	Command	Remarks
Enter system view.	**system-view**	—
Enter interface view of MPLS TE link.	**interface** *interface-type interface-number*	—
Enable RSVP authentication.	**mpls rsvp-te authentication** { **cipher** | **plain** } *auth-key*	Required.

FRR and RSVP authentication cannot run at the same time.

Configuring RSVP-TE GR

The RSVP-TE GR function depends on the extended hello capability of RSVP-TE. Be sure to enable the extended hello capability of RSVP-TE before configuring RSVP-TE GR.

To configure RSVP-TE GR on each device to act as the GR restarter or a GR helper:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Enable global RSVP hello extension.	**mpls rsvp-te hello**	Required. Disabled by default.
4. Enable MPLS RSVP-TE GR.	**mpls rsvp-te graceful-restart**	Required. Disabled by default.
5. Set the RSVP-TE GR restart timer.	**mpls rsvp-te timer graceful-restart restart** *restart-time*	Optional. 120 seconds by default.
6. Set the RSVP-TE GR recovery timer.	**mpls rsvp-te timer graceful-restart recovery** *recovery-time*	Optional. 300 seconds by default.
7. Enter interface view of MPLS TE link.	**interface** *interface-type interface-number*	—
8. Enable RSVP hello extension for the interface.	**mpls rsvp-te hello**	Required. Disabled by default.

Tuning CR-LSP setup

A CR-LSP is established through the signaling protocol based on the path calculated by CSPF using TEDB and constraints. MPLS TE can affect CSPF calculation in many ways to determine the path that a CR-LSP can traverse.

Prerequisites

The configuration tasks described in this section are about CSPF of MPLS TE. They must be used in conjunction with CSPF and the dynamic signal protocol (CR-LDP or RSVP-TE). Before performing them, be aware of each configuration objective and its impact on your system.

Procedure

Tuning CR-LSP setup involves these tasks:

- Configuring route pinning
- Configuring administrative group and affinity attribute
- Configuring CR-LSP reoptimization

Configuring route pinning

Route pinning cannot be used together with reoptimization or automatic bandwidth adjustment.

To configure route pinning:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Enable route pinning.	**mpls te route-pinning**	Required. Disabled by default.
4. Submit current tunnel configuration.	**mpls te commit**	Required.

Configuring administrative group and affinity attribute

The affinity attribute of an MPLS TE tunnel identifies the properties of the links that the tunnel can use. Together with the link administrative group, it decides which links the MPLS TE tunnel can use. This is done by ANDing the 32-bit affinity attribute with the 32-bit link administrative group attribute. When doing that, a 32-bit mask is used. The affinity bits corresponding to the 1s in the mask are “do care” bits which must be considered. Those corresponding to the 0s in the mask are “don’t care” bits.

For a link to be used by a TE tunnel, at least one considered affinity bit and its corresponding administrative group bit must be set to 1.

Suppose the affinity of an MPLS TE tunnel is 0xFFFFFFFF and the mask is 0x0000FFFF. For a link to be used by the tunnel, the leftmost 16 bits of its administrative group attribute can be 0s or 1s, but at least one of the rest bits must be 1.

The affinity of an MPLS TE tunnel is configured at the first node on the tunnel and then signaled to the rest nodes through CR-LDP or RSVP-TE.

The associations between administrative groups and affinities may vary by vendor. To ensure the successful establishment of a tunnel between two devices from different vendors, correctly configure their respective administrative groups and affinities.

To configure the administrative group and affinity attribute:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view of MPLS TE link.	**interface** *interface-type interface-number*	—
3. Assign the link to a link administrative group.	**mpls te link administrative group** *value*	Optional. The default is 0x00000000.
4. Exit to system view.	**quit**	—
5. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
6. Configure the affinity attribute of the MPLS TE tunnel.	**mpls te affinity property** *properties* [**mask** *mask-value*]	Optional. The default affinity attribute is 0x00000000, and the default mask is 0x00000000.
7. Submit current tunnel configuration.	**mpls te commit**	Required.

Configuring CR-LSP reoptimization

Dynamic CR-LSP optimization involves periodic calculation of paths that traffic trunks should traverse. If a better route is found for an existing CR-LSP, a new CR-LSP is established to replace the old one, and services are switched to the new CR-LSP.

To configure CR-LSP reoptimization:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Enable reoptimization for the MPLS TE tunnel.	**mpls te reoptimization** [**frequency** *seconds*]	Required. Disabled by default.
4. Submit current tunnel configuration.	**mpls te commit**	Required.
5. Exit to user view.	**return**	—
6. Perform reoptimization on all MPLS TE tunnels with reoptimization enabled.	**mpls te reoptimization**	Optional.

Tuning MPLS TE tunnel setup

This section only covers the configuration tasks for tuning MPLS TE tunnel setup.

Prerequisites

The configurations described in this section must be used together with the dynamic signaling protocol RSVP-TE.

Before performing the configurations, be aware of each configuration objective and its impact on your system.

Procedures

Tuning MPLS TE tunnel setup involves these tasks:

- Configuring loop detection
- Configuring route and label recording
- Configuring tunnel setup retry
- Assigning priorities to a tunnel

Configuring loop detection

To configure loop detection:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Enable the system to perform loop detection when setting up a tunnel.	**mpls te loop-detection**	Required. Disabled by default.
4. Submit current tunnel configuration.	**mpls te commit**	Required.

Configuring route and label recording

To configure route and label recording:

Step		Command	Remarks
1. Enter system view.		**system-view**	—
2. Enter MPLS TE tunnel interface view.		**interface tunnel** *tunnel-number*	—
3. Enable the system to record routes or label bindings when setting up the tunnel.	Record routes.	**mpls te record-route**	Required. Use either of the commands. Both route recording and label binding recording are disabled by default.
	Record routes and label bindings.	**mpls te record-route label**	
4. Submit current tunnel configuration		**mpls te commit**	Required.

Configuring tunnel setup retry

You may configure the system to attempt setting up a tunnel multiple times until it is established successfully or until the number of attempts reaches the upper limit.

To configure tunnel setup retry:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Configure maximum number of tunnel setup retries.	**mpls te retry** *times*	Optional. The default is 10.
4. Configure the tunnel setup retry interval.	**mpls te timer retry** *seconds*	Optional. The default is 2 seconds.
5. Submit current tunnel configuration.	**mpls te commit**	Required.

Assigning priorities to a tunnel

Two priorities, setup priority and holding priority, are assigned to paths for MPLS TE to make preemption decision. For a new path to preempt an existing path, the setup priority of the new path must be greater than the holding priority of the existing path.

To avoid flapping caused by improper preemptions between CR-LSPs, the setup priority of a CR-LSP should not be set higher than its holding priority.

To assign priorities to a tunnel:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Assign priorities to the tunnel.	**mpls te priority** *setup-priority* [*hold-priority*]	Optional. The default setup and holding priorities are 7.
4. Submit current tunnel configuration.	**mpls te commit**	Required.

Configuring traffic forwarding

Prerequisites

Before you configure traffic forwarding, complete the following tasks:

- Configure MPLS basic capabilities
- Configure MPLS TE basic capabilities
- Configure MPLS TE tunnels

Procedures

Configuring traffic forwarding involves these tasks:

- Forwarding traffic along MPLS TE tunnels using static routes

- Forwarding traffic along MPLS TE tunnels through automatic route advertisement

Forwarding traffic along MPLS TE tunnels using static routes

To create static routes for routing traffic along an MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a static route for forwarding traffic along an MPLS TE tunnel.	**ip route-static** *dest-address* { *mask* | *mask-length* } *interface-type interface-number* [*gateway-address*] | **vpn-instance** *d-vpn-instance-name gateway-address* } [**preference** *preference-value*] [**tag** *tag-value*] [**description** *description-text*]	Required.

The *interface-type* argument in **ip route-static** must be tunnel. In addition, the preference value must be set.

For more information about static routing, see *Layer 3—IP Routing Command Reference*.

Forwarding traffic along MPLS TE tunnels through automatic route advertisement

Two approaches, IGP shortcut and forwarding adjacency, are available to automatic route advertisement to advertise MPLS TE tunnel interface routes to IGPs, allowing traffic to be routed down MPLS TE tunnels.

In either approach, TE tunnels are considered point-to-point links and TE tunnel interfaces can be set as outgoing interfaces.

Routes with TE tunnel interfaces as outgoing interfaces are advertised to neighboring devices in the forwarding adjacency approach but not in the IGP shortcut approach. TE tunnels are visible to other devices in the forwarding adjacency approach but not in the IGP shortcut approach.

You may assign a metric, either absolute or relative, to TE tunnels for the purpose of path calculation in either approach. If it is absolute, the metric is directly used for path calculation. If it is relative, the cost of the corresponding IGP path must be added to the metric before it can be used for path calculation.

Enable OSPF or IS-IS on the tunnel interface of the MPLS TE tunnel before configuring automatic route advertisement.

1. Configure IGP shortcut

To configure IGP shortcut:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Configure the IGP to take the MPLS TE tunnels in an **up** state into account when performing enhanced SPF calculation.	**mpls te igp shortcut** [**isis** | **ospf**]	Required. MPLS TE tunnels are not considered in the enhanced SPF calculation of IGP. If no IGP type is specified, the configuration applies to both OSPF and ISIS by default.
4. Assign a metric to the MPLS TE tunnel.	**mpls te igp metric** { **absolute** | **relative** } *value*	Optional. The metrics of TE tunnels equal the metrics of their corresponding IGP routes by default.

Step	Command	Remarks
5. Submit current tunnel configuration.	**mpls te commit**	Required.
6. Exit to system view.	**quit**	—
7. Enter OSPF view.	**ospf** [*process-id*]	—
8. Enable the IGP shortcut function.	**enable traffic-adjustment**	Required. Disabled by default.

2. Configure forwarding adjacency

You must create a bi-directional MPLS TE tunnel and enable forwarding adjacency at both ends of the tunnel to make forwarding adjacency take effect.

To configure forwarding adjacency:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Enable IGP to advertise the route of the MPLS TE tunnel to IGP neighbors.	**mpls te igp advertise** [**hold-time** *value*]	Required. Routes of MPLS TE tunnels are not advertised to IGP neighbors by default.
4. Assign a metric to the MPLS TE tunnel.	**mpls te igp metric** { **absolute** | **relative** } *value*	Optional. The metrics of TE tunnels equal the metrics of their corresponding IGP routes by default.
5. Submit current tunnel configuration.	**mpls te commit**	Required.
6. Exit to system view.	**quit**	—
7. Enter OSPF view.	**ospf** [*process-id*]	—
8. Enable forwarding adjacency.	**enable traffic-adjustment advertise**	Required. Disabled by default

If you use automatic route advertisement, you must specify the destination address of the TE tunnel as the LSR ID of the peer and advertise the tunnel interface address to IGPs, such as OSPF and ISIS.

Configuring traffic forwarding tuning parameters

In MPLS TE, you may configure traffic forwarding tuning parameters such as the failed link timer and flooding thresholds to change paths that IP or MPLS traffic flows traverse or to define type of traffic that may travel down a TE tunnel.

Prerequisites

The configurations described in this section are used in conjunction with CSPF and the dynamic signaling protocol RSVP-TE.

Procedure

Configuring traffic forwarding tuning parameters involves these tasks:

- Configuring the failed link timer
- Configuring the link metric used for routing a tunnel
- Configuring the traffic flow type of a tunnel

Configuring the failed link timer

A CSPF failed link timer starts once a link goes down. If IGP removes or modifies the link before the timer expires, CSPF updates information about the link in TEDB and stops the timer. If IGP does not remove or modify the link before the timer expires, the state of the link in TEDB changes to up.

To configure failed link timer:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Configure the CSPF failed link timer.	**mpls te cspf timer failed-link** *timer-interval*	Optional. The default is 10 seconds.

Configuring the link metric used for routing a tunnel

For an MPLS TE link, you may assign it a TE metric. This TE metric or the IGP metric of the link is used for routing MPLS TE tunnels, depending on which metric type is specified.

To configure the link metric used for routing a tunnel:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Configure the link metric type used for routing TE tunnels without metric type.	**mpls te path metric-type** { **igp** | **te** }	Optional. TE metrics of links are used by default.
4. Exit to system view.	**quit**	—
5. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
6. Configure the link metric type used for routing the tunnel.	**mpls te path metric-type** { **igp** | **te** }	Optional. By default, no link metric type is specified and the one specified in MPLS view is used.
7. Submit current tunnel configuration.	**mpls te commit**	Optional.
8. Return to system view.	**quit**	—
9. Enter interface view of MPLS TE link.	**interface** *interface-type interface-number*	—

Step	Command	Remarks
10. Assign a TE metric to the link.	**mpls te metric** *value*	Optional. If no TE metric is assigned to the link, IGP metric is used as the TE metric by default.

The metric type configured in MPLS TE tunnel interface view takes priority over the one configured in MPLS view.

If you do not configure **mpls te path metric-type** in MPLS TE tunnel interface view, the configuration in MPLS view takes effect.

Configuring the traffic flow type of a tunnel

To configure the traffic flow type of a tunnel:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—
3. Configure the traffic flow type of the TE tunnel.	**mpls te vpn-binding** { **acl** *acl-number* | **vpn-instance** *vpn-instance-name* }	Optional. Traffic flow types of TE tunnels are not restricted by default.
4. Submit current tunnel configuration.	**mpls te commit**	Required.

Configuring CR-LSP backup

CR-LSP backup provides end-to-end path protection to protect the entire LSP.

Prerequisites

Before you configure CR-LSP backup, complete the following tasks:

- Configure MPLS basic capabilities
- Configure MPLS TE basic capabilities
- Configure MPLS TE tunnels

Procedure

To configure CR-LSP backup:

Step	Command	Remarks
1. Enter system view. of the ingress node.	**system-view**	—
2. Enter MPLS TE tunnel interface view.	**interface tunnel** *tunnel-number*	—

Step	Command	Remarks
3. Configure the backup mode used by the TE tunnel.	**mpls te backup** { **hot-standby** | **ordinary** }	Required. Tunnel backup is disabled by default.
4. Submit current tunnel configuration.	**mpls te commit**	Required.

CR-LSP backup should be configured at the ingress node of a tunnel. The system routes the primary LSP and backup LSP automatically. You do not need to configure them.

Configuring FRR

As previously mentioned, FRR provides quick but temporary per-link or per-node local protection on an LSP.

FRR uses bypass tunnels to protect primary tunnels. As bypass tunnels are pre-established, they require extra bandwidth and usually only protect crucial interfaces or links.

You can define which type of LSP can use bypass LSPs, and whether a bypass LSP provides bandwidth protection as well as the sum of protected bandwidth.

The bandwidth of a bypass LSP is to protect its primary LSPs. To guarantee that a primary LSP can always bind with the bypass LSP successfully, make sure that the bandwidth assigned to the bypass LSP is not less than the total bandwidth needed by all protected LSPs.

Normally, bypass tunnels only forward data traffic when protected primary tunnels fail. To allow a bypass tunnel to forward data traffic while protecting the primary tunnel, you must make sure that bypass tunnels are available with adequate bandwidth.

A bypass tunnel cannot be used for services like VPN at the same time.

Prerequisites

Before you configure FRR, complete the following tasks:

- Configure IGP, ensuring that all LSRs are reachable
- Configure MPLS basic capabilities
- Configure MPLS TE basic capabilities
- Establish an MPLS TE tunnel with RSVP-TE
- Set up primary LSPs

Procedure

Configuring FRR involves these tasks:

- Enabling FRR on the headend of a primary LSP
- Configuring a bypass tunnel on its PLR
- Configuring node protection
- Configuring the FRR polling timer

Enabling FRR on the headend of a primary LSP

To enable FRR on the headend of a primary LSP:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter tunnel interface view of the primary LSP.	**interface tunnel** *tunnel-number*	—
3. Enable FRR.	**mpls te fast-reroute**	Required. Disabled by default.
4. Submit current tunnel configuration.	**mpls te commit**	Required.

Configuring a bypass tunnel on its PLR

CAUTION:

Bypass tunnels do not protect bandwidth by default. This can defeat your attempts to binding a primary LSP to a bypass tunnel. Therefore, when configuring a bypass tunnel, you must configure the bandwidth that it is intended to protect with the **mpls te backup bandwidth** command.

After a tunnel is specified to protect an interface, its corresponding LSP becomes a bypass LSP. The setup of a bypass LSP must be manually performed on the PLR. The configuration of a bypass LSP is similar to that of a common LSP. However, a bypass LSP cannot act as a primary LSP to be protected by another LSP at the same time.

When specifying a bypass tunnel for an interface, make sure that:

- The bypass tunnel is up.
- The protected interface is not the outgoing interface of the bypass tunnel.

Up to three bypass tunnels can be specified for a protected interface. The best-fit algorithm determines which of them should be used in case failure occurs.

Your device has restriction on links that use the same bypass tunnel so that their total bandwidth does not exceeds a specified value.

To configure a bypass tunnel on its PLR:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view of the bypass tunnel.	**interface tunnel** *tunnel-number*	—
3. Specify the destination address of the bypass tunnel.	**destination** *ip-address*	Required. • For node protection, this is the LSR ID of the next hop router of PLR. • For link protection, this is the LSR ID of the next hop device of PLR.
4. Configure the bandwidth and type of LSP that the bypass tunnel can protect.	**mpls te backup bandwidth** *bandwidth*	Required. Bandwidth is not protected by default.
5. Submit current tunnel configuration.	**mpls te commit**	Required.

Step	Command	Remarks
6. Exit to system view.	**quit**	—
7. Enter interface view of the outgoing interface of the protected LSP.	**interface** *interface-type interface-number*	—
8. Bind the bypass tunnel with the protected interface.	**mpls te fast-reroute bypass-tunnel tunnel** *tunnel-number*	Required.

Configuring node protection

To use FRR for node protection, you must perform the tasks in this section on the PLR and the protected node. If you only must protect links, skip this section.

To configure node protection:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Enable RSVP hello extension on current node.	**mpls rsvp-te hello**	Required. Disabled by default.
4. Exit to system view.	**quit**	—
5. Enter the view of the interface directly connected to the protected node or PLR.	**interface** *interface-type interface-number*	—
6. Enable RSVP hello extension on the interface.	**mpls rsvp-te hello**	Required. Disabled by default.

RSVP hello extension is configured to detect node failures caused by problems such as signaling error other than failures caused by link failures.

Configuring cooperation of MPLS RSVP-TE and BFD

Cooperation of MPLS RSVP-TE and BFD is mainly used for link status detection of FRR. When BFD detects a link failure, it immediately notifies the failure to FRR, which then reroutes traffic from the primary LSP to the bypass LSP.

To configure cooperation of MPLS RSVP-TE and BFD:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter view of the interface enabled with MPLS RSVP-TE.	**interface** *interface-type interface-number*	—
3. Enable BFD for MPLS RSVP-TE.	**mpls rsvp-te bfd enable**	Required. Disabled by derfault.

Configuring the FRR polling timer

The protection provided by FRR is temporary. Once a protected LSP becomes available again or a new LSP is established, traffic is switched to the protected or new LSP. After this switchover, the PLR polls available bypass tunnels for the best one at the regular interval specified by the FRR polling timer:

To configure the FRR polling timer:

Step	Command	Remarks
1. Enter system view. of the PLR node.	**system-view**	—
2. Enter MPLS view.	**mpls**	—
3. Configure the FRR polling timer.	**mpls te timer fast-reroute** [*second*]	Optional. The FRR polling timer is 300 seconds by default.

Inspecting an MPLS TE tunnel

On an MPLS TE network, when an MPLS TE tunnel fails, the control plane cannot detect the failure or cannot do so in time. This brings difficulty to network maintenance. To detect MPLS TE tunnel failures in time and locate the failed node, the device provides the following mechanisms:

- MPLS LSP ping
- MPLS LSP tracert
- BFD for an MPLS TE tunnel
- Periodic tracert of an MPLS TE tunnel

Using MPLS LSP ping

You can use MPLS LSP ping to check the validity and availability of an MPLS TE tunnel. At the ingress, it adds the label for the MPLS TE tunnel to be inspected into an MPLS echo request, which then is forwarded along the MPLS TE tunnel to the egress. The ingress determines whether the MPLS TE tunnel is normal according to whether it can receive a reply from the egress.

Use the following command to check the validity and reachability of an MPLS TE tunnel:

Task	Command
Use MPLS LSP ping to check the validity and reachability of an MPLS TE tunnel.	**ping lsp** [**-a** *source-ip* | **-c** *count* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *wait-time* | **-r** *reply-mode* | **-s** *packet-size* | **-t** *time-out* | **-v**] * **te** *interface-type interface-number*

Using MPLS LSP tracert

You can use MPLS LSP tracert to locate errors of an MPLS TE tunnel. It sends MPLS echo requests to the nodes along the MPLS TE tunnel to be inspected, with the TTL increasing from 1 to a specified value. Each node along the MPLS TE tunnel returns an MPLS echo reply to the ingress due to TTL timeout. As a result, the ingress can collect the information of each hop along the MPLS TE tunnel, so as to locate the failed node. You can also use MPLS LSP tracert to collect important information of each hop along the MPLS TE tunnel, such as the label allocated.

Use the following command to locate errors of an MPLS TE tunnel:

Task	Command
Use MPLS LSP tracert to locate errors of an MPLS TE tunnel.	**tracert lsp** [**-a** *source-ip* | **-exp** *exp-value* | **-h** *ttl-value* | **-r** *reply-mode* |**-t** *time-out*] * **te** *interface-type interface-number*

Configuring BFD for an MPLS TE tunnel

You can configure BFD for an MPLS TE tunnel to implement fast detection of the connectivity of the tunnel. After you configure BFD for an MPLS TE tunnel, a BFD session is established between the ingress and egress of the tunnel, and the ingress adds the label for the tunnel into a BFD control packet, forward the BFD control packet along the tunnel, and determine the status of the tunnel according to the BFD control packet received from the egress. Once detecting an MPLS TE tunnel failure, BFD triggers protection switching to switch traffic to another tunnel.

A BFD session for MPLS TE tunnel detection can be static or dynamic:

- Static: If you specify the local and remote discriminator values by using the **discriminator** keyword when configuring **mpls te bfd enable**, the BFD session is established with the specified discriminator values. Such a BFD session can detect the connectivity of a pair of MPLS TE tunnels in opposite directions (one from local to remote, and the other from remote to local) between two devices.
- Dynamic: If you do not specify the local and remote discriminator values when configuring **mpls te bfd enable**, the MPLS LSP ping runs automatically to negotiate the discriminator values and then the BFD session is established based on the negotiated discriminator values. Such a BFD session can detect the connectivity of a unidirectional (from the local device to the remote device) MPLS TE tunnel between two devices.

After you enable BFD and configure **mpls te failure-action teardown** for an MPLS TE tunnel, once an RSVP-TE tunnel failure occurs, BFD can detect the failure, and if RSVP does not re-establish the tunnel within a specified period of time, MPLS TE removes the failed RSVP-TE tunnel and then re-establish it.

To configure BFD for an MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enable LSP verification and enter MPLS LSPV view.	**mpls lspv**	Required. By default, LSP verification is disabled.
3. Return to system view.	**quit**	—
4. Enter the tunnel interface view of an MPLS TE tunnel.	**interface tunnel** *tunnel-number*	—
5. Configure BFD to check the connectivity of the MPLS TE tunnel.	**mpls te bfd enable** [**discriminator local** *local-id* **remote** *remote-id*]	Required. By default, BFD is not configured to check connectivity of MPLS TE tunnels.
6. Configure MPLS TE to tear down a failed RSVP TE tunnel and reestablish it.	**mpls te failure-action teardown**	Optional. Not configured by default.

For information about **mpls lspv**, see *MPLS Command Reference*.

The BFD session parameters are those configured on the MPLS TE tunnel interface. The source address of the BFD session is the MPLS LSR ID. Therefore, before configuring BFD to inspect an MPLS TE tunnel, make sure that the peer device has a route to the MPLS LSR ID, and you can also configure the BFD session parameters on the tunnel interface as needed. For information about BFD parameter configuration, see *High Availability Configuration Guide*.

You cannot establish both a static BFD session and a dynamic BFD session for the same MPLS TE tunnel.

Before establishing a static BFD session to detect the reachability of a pair of MPLS TE tunnels in opposite directions between the local device and the remote device, make sure that the two MPLS TE tunnels already exist.

After establishing a static BFD session for an MPLS TE tunnel, you are not allowed to modify the discriminator values of the BFD session.

If you enable both FRR and BFD for an MPLS TE tunnel, you must give the BFD detection interval a greater value than the FRR detection interval to ensure that the BFD session is not down during an FRR switching.

Configuring periodic LSP tracert for an MPLS TE tunnel

The periodic LSP tracert function for an MPLS TE tunnel is for locating faults of the MPLS TE tunnel periodically. It detects the consistency of the forwarding and control plane and records detection results into logs. You can know whether an MPLS TE tunnel has failed by checking the logs.

If you configure BFD as well as periodical tracert for an MPLS TE tunnel, once the periodical LSP tracert function detects a fault or inconsistency of the forwarding plane and control plane of the MPLS TE tunnel, the BFD session for the tunnel is deleted and a new BFD session is established according to the control plane.

After you configure periodic LSP tracert and **mpls te failure-action teardown** for an MPLS TE tunnel, once an RSVP-TE tunnel failure occurs, the periodic LSP tracert function can detect the failure, and if RSVP does not re-establish the RSVP-TE tunnel within a specified period of time, MPLS TE removes the failed RSVP-TE tunnel and then re-establish it.

To configure periodic LSP tracert for an MPLS TE tunnel:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enable LSP verification and enter MPLS LSPV view.	**mpls lspv**	Required. By default, LSP verification is disabled.
3. Return to system view.	**quit**	—
4. Enter the tunnel interface view of an MPLS TE tunnel.	**interface tunnel** *tunnel-number*	—
5. Enable periodic LSP tracert for the MPLS TE tunnel.	**mpls te periodic-tracert** [**-a** *source-ip* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *wait-time* | **-t** *time-out* | **-u** *retry-attempt*] *	Required. By default, periodic LSP tracert is disabled for MPLS TE tunnels.
6. Configure MPLS TE to tear down a failed RSVP TE tunnel and reestablish it.	**mpls te failure-action teardown**	Optional. Not configured by default.

For more information about **mpls lspv**, see *MPLS Command Reference*.

Displaying and maintaining MPLS TE

Displaying and maintaining MPLS TE

Task	Command	Remarks
Display information about explicit paths.	**display explicit-path** [*pathname*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]]	Available in any view.
Display information about static CR-LSPs.	**display mpls static-cr-lsp** [**lsp-name** *lsp-name*] [{ **include** | **exclude** } *ip-address prefix-length*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]]	Available in any view.
Display RSVP-TE configuration.	**display mpls rsvp-te** [**interface** [*interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]]	Available in any view.
Display RSVP-TE information.	**display mpls rsvp-te established** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display RSVP-TE neighbors.	**display mpls rsvp-te peer** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about RSVP requests.	**display mpls rsvp-te request** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about RSVP resource reservation.	**display mpls rsvp-te reservation** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about RSVP-TE PSB.	**display mpls rsvp-te psb-content** { *ingress-lsr-id lspid tunnel-id egress-lsr-id* } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about RSVP-TE RSB.	**display mpls rsvp-te rsb-content** { *ingress-lsr-id lspid tunnel-id egress-lsr-id nexthop-address* } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about RSVP sender messages .	**display mpls rsvp-te sender** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display statistics about RSVP-TE.	**display mpls rsvp-te statistics** { **global** | **interface** [*interface-type interface-number*] } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display criteria-compliant information about CSPF-based TEDB.	**display mpls te cspf tedb** { **all** | **area** *area-id* | **interface** *ip-address* | **network-lsa** | **node** [*mpls-lsr-id*] } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.

Task	Command	Remarks
Display information about the CR-LSPs carried on the specified or all links.	display mpls te link-administration admission-control [interface *interface-type interface-number*] [| { begin | exclude | include } *regular-expression*]	Available in any view.
Display information about MPLS TE tunnels.	display mpls te tunnel [destination *dest-addr*] [lsp-id *lsr-id lsp-id*] [lsr-role { all | egress | ingress | remote | transit }] [name *name*] [{ incoming-interface | outgoing-interface | interface } *interface-type interface-number*] [verbose] [| { begin | exclude | include } *regular-expression*]	Available in any view.
Display the path attributes of MPLS TE tunnels on this node.	**display mpls te tunnel path** [**lsp-id** *lsr-id lsp-id* | **tunnel-name** *tunnel-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display tunnel statistics.	display mpls te tunnel statistics [| { begin | exclude | include } *regular-expression*]	Available in any view.
Display statistics about MPLS TE tunnels.	display mpls te tunnel-interface tunnel *number* [| { begin | exclude | include } *regular-expression*]	Available in any view.
Display the information of the specified or all OSPF processes about traffic tuning.	display ospf [*process-id*] traffic-adjustment [| { begin | exclude | include } *regular-expression*]	Available in any view.
Display information about OSPF TE.	**display ospf** [*process-id*] **mpls-te** [**area** *area-id*] [**self-originated**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display the latest TE information advertised by IS-IS TE.	display isis traffic-eng advertisements [[level-1 | level-1-2 | level-2] | [lsp-id *lsp-id* | local]] * [*process-id* | vpn-instance *vpn-instance-name*] [| { begin | exclude | include } *regular-expression*]	Available in any view.
Display information about TE links for IS-IS.	**display isis traffic-eng link** [[**level-1** | **level-1-2** | **level-2**] | **verbose**] * [*process-id* | **vpn-instance** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about TE networks for IS-IS.	**display isis traffic-eng network** [**level-1** | **level-1-2** | **level-2**] [*process-id* | **vpn-instance** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display statistics about TE for IS-IS.	**display isis traffic-eng statistics** [*process-id* | **vpn-instance** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.

Task	Command	Remarks
Display information about tunnels.	**display tunnel-info** { *tunnel-id* | **all** | **statistics** } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display the BFD information for an MPLS TE tunnel.	**display mpls lsp bfd** [**te tunnel** *tunnel-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Clear the statistics about RSVP-TE.	**reset mpls rsvp-te statistics** { **global** | **interface** [*interface-type interface-number*]	Available in user view.

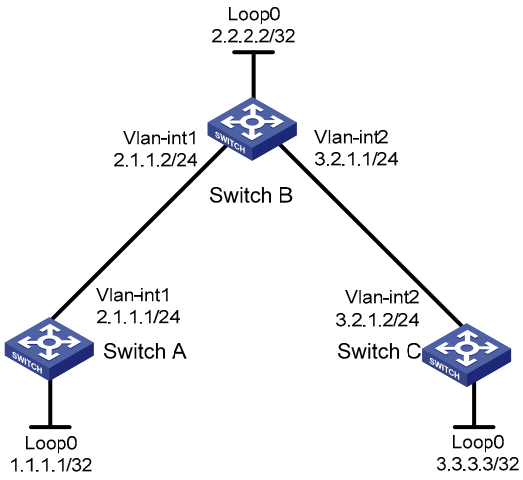
MPLS TE configuration examples

MPLS TE using static CR-LSP configuration example

Network requirements

- Switch A, Switch B, and Switch C run IS-IS.
- Establish a TE tunnel using a static CR-LSP between Switch A and Switch C.

Figure 31 Set up MPLS TE tunnels using static CR-LSPs



Procedure

1. Assign IP addresses and masks to interfaces (see Figure 31)

Omitted

2. Enable IS-IS to advertise host routes with LSR IDs as destinations

Configure Switch A.

```
<SwitchA> system-view
[SwitchA] isis 1
[SwitchA-isis-1] network-entity 00.0005.0000.0000.0001.00
[SwitchA-isis-1] quit
[SwitchA] interface vlan-interface 1
[SwitchA-Vlan-interface1] isis enable 1
```

```
[SwitchA-Vlan-interface1] quit
[SwitchA] interface loopback 0
[SwitchA-LoopBack0] isis enable 1
[SwitchA-LoopBack0] quit
```

Configure Switch B.

```
<SwitchB> system-view
[SwitchB] isis 1
[SwitchB-isis-1] network-entity 00.0005.0000.0000.0002.00
[SwitchB-isis-1] quit
[SwitchB] interface vlan-interface 1
[SwitchB-Vlan-interface1] isis enable 1
[SwitchB-Vlan-interface1] quit
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] isis enable 1
[SwitchB-Vlan-interface2] quit
[SwitchB] interface loopback 0
[SwitchB-LoopBack0] isis enable 1
[SwitchB-LoopBack0] quit
```

Configure Switch C.

```
<SwitchC> system-view
[SwitchC] isis 1
[SwitchC-isis-1] network-entity 00.0005.0000.0000.0003.00
[SwitchC-isis-1] quit
[SwitchC] interface vlan-interface 2
[SwitchC-Vlan-interface2] isis enable 1
[SwitchC-Vlan-interface2] quit
[SwitchC] interface loopback 0
[SwitchC-LoopBack0] isis enable 1
[SwitchC-LoopBack0] quit
```

Perform **display ip routing-table** on each switch. The output shows that all nodes have learned the host routes of other nodes with LSR IDs as destinations. Take Switch A for example:

```
[SwitchA] display ip routing-table
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto  Pre  Cost      NextHop         Interface
        1.1.1.1/32  Direct 0    0         127.0.0.1       InLoop0
        2.1.1.0/24  Direct 0    0         2.1.1.1         Vlan1
        2.1.1.1/32  Direct 0    0         127.0.0.1       InLoop0
        2.2.2.2/32  ISIS   15   10        2.1.1.2         Vlan1
        3.2.1.0/24  ISIS   15   20        2.1.1.2         Vlan1
        3.3.3.3/32  ISIS   15   20        2.1.1.2         Vlan1
      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 MPLS TE basic capabilities

Configure Switch A.

```
[SwitchA] mpls lsr-id 3.3.3.3
```

```
[SwitchA] mpls
[SwitchA-mpls] mpls te
[SwitchA-mpls] quit
[SwitchA] interface vlan-interface 1
[SwitchA-Vlan-interface1] mpls
[SwitchA-Vlan-interface1] mpls te
[SwitchA-Vlan-interface1] quit
```

Configure Switch B.

```
[SwitchB] mpls lsr-id 2.2.2.2
[SwitchB] mpls
[SwitchB-mpls] mpls te
[SwitchB-mpls] quit
[SwitchB] interface vlan-interface 1
[SwitchB-Vlan-interface1] mpls
[SwitchB-Vlan-interface1] mpls te
[SwitchB-Vlan-interface1] quit
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] mpls
[SwitchB-Vlan-interface2] mpls te
[SwitchB-Vlan-interface2] quit
```

Configure Switch C.

```
[SwitchC] mpls lsr-id 3.3.3.3
[SwitchC] mpls
[SwitchC-mpls] mpls te
[SwitchC-mpls] quit
[SwitchC] interface vlan-interface 2
[SwitchC-Vlan-interface2] mpls
[SwitchC-Vlan-interface2] mpls te
[SwitchC-Vlan-interface2] quit
```

4. Configure an MPLS TE tunnel

Configure an MPLS TE tunnel on Switch A.

```
[SwitchA] interface tunnel 0
[SwitchA-Tunnel0] ip address 6.1.1.1 255.255.255.0
[SwitchA-Tunnel0] tunnel-protocol mpls te
[SwitchA-Tunnel0] destination 3.3.3.3
[SwitchA-Tunnel0] mpls te tunnel-id 10
[SwitchA-Tunnel0] mpls te signal-protocol static
[SwitchA-Tunnel0] mpls te commit
[SwitchA-Tunnel0] quit
```

5. Create a static CR-LSP

Configure Switch A as the ingress node of the static CR-LSP.

```
[SwitchA] static-cr-lsp ingress Tunnel0 destination 3.3.3.3 nexthop 2.1.1.2 out-label 20
```

Configure Switch B as the transit node of the static CR-LSP.

```
[SwitchB] static-cr-lsp transit tunnel0 incoming-interface Vlan-interface1 in-label 20
nexthop 3.2.1.2 out-label 30
```

Configure Switch C as the egress node of the static CR-LSP.

```
[SwitchC] static-cr-lsp egress tunnel0 incoming-interface Vlan-interface2 in-label 30
```

6. Verify the configuration

Perform **display interface tunnel** on Switch A. You can see that the tunnel interface is up.

```
[SwitchA] display interface tunnel
Tunnel0 current state: UP
Line protocol current state: UP
Description: Tunnel0 Interface
The Maximum Transmit Unit is 64000
Internet Address is 6.1.1.1/24 Primary
Encapsulation is TUNNEL, service-loopback-group ID not set
Tunnel source unknown, destination 3.3.3.3
Tunnel protocol/transport CR_LSP
Output queue : (Urgent queuing : Size/Length/Discards)  0/100/0
Output queue : (Protocol queuing : Size/Length/Discards)  0/500/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
    Last 300 seconds input:  0 bytes/sec, 0 packets/sec
    Last 300 seconds output:  0 bytes/sec, 0 packets/sec
    0 packets input,  0 bytes
    0 input error
    0 packets output,  0 bytes
    0 output error
```

Perform **display mpls te tunnel** on each switch to verify information about the MPLS TE tunnel.

```
[SwitchA] display mpls te tunnel
LSP-Id          Destination     In/Out-If               Name
1.1.1.1:1       3.3.3.3         -/Vlan1                 Tunnel0
[SwitchB] display mpls te tunnel
LSP-Id          Destination     In/Out-If               Name
-               -               Vlan1/Vlan2             Tunnel0
[SwitchC] display mpls te tunnel
LSP-Id          Destination     In/Out-If               Name
-               -               Vlan2/-                 Tunnel0
```

Perform **display mpls lsp** or **display mpls static-cr-lsp** on each switch to verify information about the static CR-LSP.

```
[SwitchA] display mpls lsp
--------------------------------------------------------------------
                LSP Information: STATIC CRLSP
--------------------------------------------------------------------
FEC             In/Out Label  In/Out IF             Vrf Name
3.3.3.3/32      NULL/20       -/Vlan1
[SwitchB] display mpls lsp
--------------------------------------------------------------------
                LSP Information: STATIC CRLSP
--------------------------------------------------------------------
FEC             In/Out Label  In/Out IF              Vrf Name
-/-             20/30         Vlan1/Vlan2
[SwitchC] display mpls lsp
--------------------------------------------------------------------
```

```
                LSP Information: STATIC CRLSP
--------------------------------------------------------------------
FEC              In/Out Label  In/Out IF              Vrf Name
-/-              30/NULL       Vlan1/-
[SwitchA] display mpls static-cr-lsp
total static-cr-lsp : 1
Name        FEC              I/O Label  I/O If               State
Tunnel0     3.3.3.3/32       NULL/20    -/Vlan1              Up
[SwitchB] display mpls static-cr-lsp
total statics-cr-lsp : 1
Name        FEC              I/O Label  I/O If               State
Tunnel0     -/-              20/30      Vlan1/Vlan2          Up
[SwitchC] display mpls static-cr-lsp
total statics-cr-lsp : 1
Name        FEC              I/O Label  I/O If                State
Tunnel0     -/-              30/NULL    Vlan2/-               Up
```

On an MPLS TE tunnel configured using a static CR-LSP, traffic is forwarded directly based on label at the transit nodes and egress node. Therefore, it is normal that the FEC field in the sample output is empty on Switch B and Switch C.

7. Create a static route for routing MPLS TE tunnel traffic.

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

Perform **display ip routing-table** on Switch A. You can see a static route entry with interface Tunnel 0 as the outgoing interface.

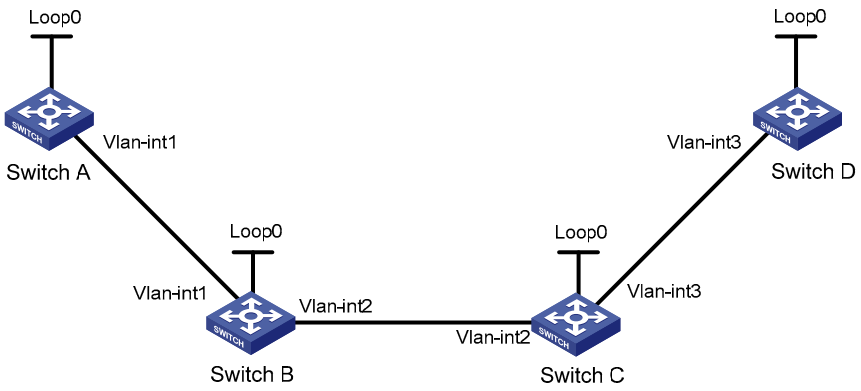
MPLS TE using RSVP-TE configuration example

Network requirements

Switch A, Switch B, Switch C, and Switch D are running IS-IS and all of them are Level-2 devices.

Use RSVP-TE to create a TE tunnel from Switch A to Switch D, ensuring that the maximum bandwidth of each link that the tunnel traverses is 10000 kbps.

Figure 32 Set up MPLS TE tunnels using RSVP-TE



Device	Interface	IP address	Device	Interface	IP address
Switch A	Loop0	1.1.1.9/32	Switch D	Loop0	4.4.4.9/32
	Vlan-int1	10.1.1.1/24		Vlan-int3	30.1.1.2/24

Switch B	Loop0	2.2.2.9/32	Switch C	Loop0	3.3.3.9/32
	Vlan-int1	10.1.1.2/24		Vlan-int3	30.1.1.1/24
	Vlan-int2	20.1.1.1/24		Vlan-int2	20.1.1.2/24

Procedure

1. Assign IP addresses and masks to interfaces (see Figure 32)

Omitted

2. Enable IS-IS to advertise host routes with LSR IDs as destinations

Configure Switch A.

```
<SwitchA> system-view
[SwitchA] isis 1
[SwitchA-isis-1] network-entity 00.0005.0000.0000.0001.00
[SwitchA-isis-1] quit
[SwitchA] interface vlan-interface 1
[SwitchA-Vlan-interface1] isis enable 1
[SwitchA-Vlan-interface1] isis circuit-level level-2
[SwitchA-Vlan-interface1] quit
[SwitchA] interface loopback 0
[SwitchA-LoopBack1] isis enable 1
[SwitchA-LoopBack1] isis circuit-level level-2
[SwitchA-LoopBack1] quit
```

Configure Switch B.

```
<SwitchB> system-view
[SwitchB] isis 1
[SwitchB-isis-1] network-entity 00.0005.0000.0000.0002.00
[SwitchB-isis-1] quit
[SwitchB] interface vlan-interface 1
[SwitchB-Vlan-interface1] isis enable 1
[SwitchB-Vlan-interface1] isis circuit-level level-2
[SwitchB-Vlan-interface1] quit
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] isis enable 1
[SwitchB-Vlan-interface2] isis circuit-level level-2
[SwitchB-Vlan-interface2] quit
[SwitchB] interface loopback 0
[SwitchB-LoopBack0] isis enable 1
[SwitchB-LoopBack0] isis circuit-level level-2
[SwitchB-LoopBack0] quit
```

Configure Switch C.

```
<SwitchC> system-view
[SwitchC] isis 1
[SwitchC-isis-1] network-entity 00.0005.0000.0000.0003.00
[SwitchC-isis-1] quit
[SwitchC] interface vlan-interface 3
[SwitchC-Vlan-interface3] isis enable 1
[SwitchC-Vlan-interface3] isis circuit-level level-2
```

```
[SwitchC-Vlan-interface3] quit
[SwitchC] interface vlan-interface 2
[SwitchC-Vlan-interface2] isis enable 1
[SwitchC-Vlan-interface2] isis circuit-level level-2
[SwitchC-Vlan-interface2] quit
[SwitchC] interface loopback 0
[SwitchC-LoopBack0] isis enable 1
[SwitchC-LoopBack0] isis circuit-level level-2
[SwitchC-LoopBack0] quit
```

Configure Switch D.

```
<SwitchD> system-view
[SwitchD] isis 1
[SwitchD-isis-1] network-entity 00.0005.0000.0000.0004.00
[SwitchD-isis-1] quit
[SwitchD] interface vlan-interface 3
[SwitchD-Vlan-interface3] isis enable 1
[SwitchD-Vlan-interface3] isis circuit-level level-2
[SwitchD-Vlan-interface3] quit
[SwitchD] interface loopback 0
[SwitchD-LoopBack0] isis enable 1
[SwitchD-LoopBack0] isis circuit-level level-2
[SwitchD-LoopBack0] quit
```

Perform **display ip routing-table** on each switch. The output shows that all nodes have learned the host routes of other nodes with LSR IDs as destinations. Take Switch A for example:

```
[SwitchA] display ip routing-table
Routing Tables: Public
         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  ISIS   15   10        10.1.1.2        Vlan1
      3.3.3.9/32  ISIS   15   20        10.1.1.2        Vlan1
      4.4.4.9/32  ISIS   15   30        10.1.1.2        Vlan1
     10.1.1.0/24  Direct 0    0         10.1.1.1        Vlan1
     10.1.1.1/32  Direct 0    0         127.0.0.1       InLoop0
     20.1.1.0/24  ISIS   15   20        10.1.1.2        Vlan1
     30.1.1.0/24  ISIS   15   30        10.1.1.2        Vlan1
    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 MPLS TE basic capabilities, and enable RSVP-TE and CSPF.

Configure Switch A.

```
[SwitchA] mpls lsr-id 1.1.1.9
[SwitchA] mpls
[SwitchA-mpls] mpls te
[SwitchA-mpls] mpls rsvp-te
[SwitchA-mpls] mpls te cspf
[SwitchA-mpls] quit
[SwitchA] interface vlan-interface 1
```

```
[SwitchA-Vlan-interface1] mpls
[SwitchA-Vlan-interface1] mpls te
[SwitchA-Vlan-interface1] mpls rsvp-te
[SwitchA-Vlan-interface1] quit
```

Configure Switch B.

```
[SwitchB] mpls lsr-id 2.2.2.9
[SwitchB] mpls
[SwitchB-mpls] mpls te
[SwitchB-mpls] mpls rsvp-te
[SwitchB-mpls] mpls te cspf
[SwitchB-mpls] quit
[SwitchB] interface vlan-interface 1
[SwitchB-Vlan-interface1] mpls
[SwitchB-Vlan-interface1] mpls te
[SwitchB-Vlan-interface1] mpls rsvp-te
[SwitchB-Vlan-interface1] quit
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] mpls
[SwitchB-Vlan-interface2] mpls te
[SwitchB-Vlan-interface2] mpls rsvp-te
[SwitchB-Vlan-interface1] quit
```

Configure Switch C.

```
[SwitchC] mpls lsr-id 3.3.3.9
[SwitchC] mpls
[SwitchC-mpls] mpls te
[SwitchC-mpls] mpls rsvp-te
[SwitchC-mpls] mpls te cspf
[SwitchC-mpls] quit
[SwitchC] interface vlan-interface 3
[SwitchC-Vlan-interface3] mpls
[SwitchC-Vlan-interface3] mpls te
[SwitchC-Vlan-interface3] mpls rsvp-te
[SwitchC-Vlan-interface3] quit
[SwitchC] interface vlan-interface 2
[SwitchC-Vlan-interface2] mpls
[SwitchC-Vlan-interface2] mpls te
[SwitchC-Vlan-interface2] mpls rsvp-te
[SwitchC-Vlan-interface2] quit
```

Configure Switch D.

```
[SwitchD] mpls lsr-id 4.4.4.9
[SwitchD] mpls
[SwitchD-mpls] mpls te
[SwitchD-mpls] mpls rsvp-te
[SwitchD-mpls] mpls te cspf
[SwitchD-mpls] quit
[SwitchD] interface vlan-interface 3
[SwitchD-Vlan-interface3] mpls
```

```
[SwitchD-Vlan-interface3] mpls te
[SwitchD-Vlan-interface3] mpls rsvp-te
[SwitchD-Vlan-interface3] quit
```

4. Configure IS-IS TE

Configure Switch A.

```
[SwitchA] isis 1
[SwitchA-isis-1] cost-style wide
[SwitchA-isis-1] traffic-eng level-2
[SwitchA-isis-1] quit
```

Configure Switch B.

```
[SwitchB] isis 1
[SwitchB-isis-1] cost-style wide
[SwitchB-isis-1] traffic-eng level-2
[SwitchB-isis-1] quit
```

Configure Switch C.

```
[SwitchC] isis 1
[SwitchC-isis-1] cost-style wide
[SwitchC-isis-1] traffic-eng level-2
[SwitchC-isis-1] quit
```

Configure Switch D.

```
[SwitchD] isis 1
[SwitchD-isis-1] cost-style wide
[SwitchD-isis-1] traffic-eng level-2
[SwitchD-isis-1] quit
```

5. Create an MPLS TE tunnel

Create an MPLS TE tunnel on Switch A.

```
[SwitchA] interface tunnel 1
[SwitchA-Tunnel1] ip address 7.1.1.1 255.255.255.0
[SwitchA-Tunnel1] tunnel-protocol mpls te
[SwitchA-Tunnel1] destination 4.4.4.9
[SwitchA-Tunnel1] mpls te tunnel-id 10
[SwitchA-Tunnel1] mpls te signal-protocol rsvp-te
[SwitchA-Tunnel1] mpls te bandwidth 2000
[SwitchA-Tunnel1] mpls te commit
[SwitchA-Tunnel1] quit
```

6. Verify the configuration

Perform **display interface tunnel** on Switch A. You can see that the tunnel interface is up.

```
[SwitchA] display interface tunnel
Tunnel1 current state: UP
Line protocol current state: UP
Description: Tunnel1 Interface
The Maximum Transmit Unit is 64000
Internet Address is 7.1.1.1/24 Primary
Encapsulation is TUNNEL, service-loopback-group ID not set
Tunnel source unknown, destination 4.4.4.9
```

```
Tunnel protocol/transport CR_LSP
Output queue : (Urgent queuing : Size/Length/Discards)  0/100/0
Output queue : (Protocol queuing : Size/Length/Discards)  0/500/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
    Last 300 seconds input:  0 bytes/sec, 0 packets/sec
    Last 300 seconds output:  0 bytes/sec, 0 packets/sec
    0 packets input,  0 bytes
    0 input error
    0 packets output,  0 bytes
    0 output error
```

Perform **display mpls te tunnel-interface** on Switch A to verify information about the MPLS TE tunnel.

```
[SwitchA] display mpls te tunnel-interface
  Tunnel Name       :  Tunnel1
  Tunnel Desc       :  Tunnel1 Interface
  Tunnel State Desc :  CR-LSP is Up
  Tunnel Attributes :
    LSP ID              :  1.1.1.9:3
    Session ID          :  10
    Admin State         :  UP                Oper State   :  UP
    Ingress LSR ID      :  1.1.1.9           Egress LSR ID:  4.4.4.9
    Signaling Prot      :  RSVP              Resv Style   :  SE
    Class Type          :  CT 0              Tunnel BW    :
    Reserved BW         :  2000 kbps
    Setup Priority      :  7                    Hold Priority:  7
    Affinity Prop/Mask  :  0x0/0x0
    Explicit Path Name  :  -
    Tie-Breaking Policy :  None
    Metric Type         :  None
    Record Route        :  Disabled    Record Label :  Disabled
    FRR Flag            :  Disabled    BackUpBW Flag:  Not Supported
    BackUpBW Type       :  -           BackUpBW     :  -
    Route Pinning       :  Disabled
    Retry Limit         :  10          Retry Interval:  10 sec
    Reopt               :  Disabled    Reopt Freq   :  -
    Back Up Type        :  None
    Back Up LSPID       :  -
    Auto BW             :  Disabled    Auto BW Freq :  -
    Min BW              :  -           Max BW       :  -
    Current Collected BW:  -
    Interfaces Protected:  -
    VPN Bind Type       :  NONE
    VPN Bind Value      :  -
    Car Policy          :  Disabled
    Tunnel Group        :  Primary
    Primary Tunnel      :  -
    Backup Tunnel       :  -
    Group Status        :  -
    Oam Status          :  -
```

Perform **display mpls te cspf tedb all** on Switch A to view information about links in TEDB.

```
[SwitchA] display mpls te cspf tedb all
Maximum Node Supported: 128              Maximum Link Supported: 256
Current Total Node Number: 4             Current Total Link Number: 6
Id      MPLS LSR-Id  IGP      Process-Id   Area          Link-Count
1       3.3.3.9      ISIS     1            Level-2       2
2       2.2.2.9      ISIS     1            Level-2       2
3       4.4.4.9      ISIS     1            Level-2       1
4       1.1.1.9      ISIS     1            Level-2       1
```

7. Create a static route for routing MPLS TE tunnel traffic

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

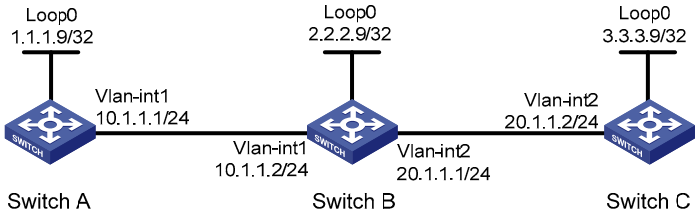
Perform **display ip routing-table** on Switch A. You can see a static route entry with interface Tunnel1 as the outgoing interface.

RSVP-TE GR configuration example

Network requirements

- Switch A, Switch B and Switch C are running IS-IS. All of them are Level-2 devices and support RSVP hello extension.
- Use RSVP-TE to create a TE tunnel from Switch A to Switch C.
- Switch A, Switch B and Switch C are RSVP-TE neighbors. With GR capability, each of them can provide GR helper support when another is GR restarting.

Figure 33 Configure RSVP-TE GR



Procedure

1. Assign IP addresses and masks to interfaces (see Figure 33)

Omitted

2. Enable IS-IS to advertise host routes with LSR IDs as destinations

Omitted

3. Configure MPLS TE basic capabilities, and enable RSVP-TE and RSVP hello extension

Configure Switch A.

```
<SwitchA> system-view
[SwitchA] mpls lsr-id 1.1.1.9
[SwitchA] mpls
[SwitchA-mpls] mpls te
[SwitchA-mpls] mpls rsvp-te
[SwitchA-mpls] mpls rsvp-te hello
[SwitchA-mpls] interface vlan-interface 1
```

```
[SwitchA-Vlan-interface1] mpls
[SwitchA-Vlan-interface1] mpls te
[SwitchA-Vlan-interface1] mpls rsvp-te
[SwitchA-Vlan-interface1] mpls rsvp-te hello
[SwitchA-Vlan-interface1] quit
```

Configure Switch B.

```
<SwitchB> system-view
[SwitchB] mpls lsr-id 2.2.2.9
[SwitchB] mpls
[SwitchB-mpls] mpls te
[SwitchB-mpls] mpls rsvp-te
[SwitchB-mpls] mpls rsvp-te hello
[SwitchB-mpls] interface vlan-interface 1
[SwitchB-Vlan-interface1] mpls
[SwitchB-Vlan-interface1] mpls te
[SwitchB-Vlan-interface1] mpls rsvp-te
[SwitchB-Vlan-interface1] mpls rsvp-te hello
[SwitchB-Vlan-interface1] quit
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] mpls
[SwitchB-Vlan-interface2] mpls te
[SwitchB-Vlan-interface2] mpls rsvp-te
[SwitchB-Vlan-interface2] mpls rsvp-te hello
[SwitchB-Vlan-interface2] quit
```

Configure Switch C.

```
<SwitchC> system-view
[SwitchC] mpls lsr-id 3.3.3.9
[SwitchC] mpls
[SwitchC-mpls] mpls te
[SwitchC-mpls] mpls rsvp-te
[SwitchC-mpls] mpls rsvp-te hello
[SwitchC-mpls] quit
[SwitchC] interface vlan-interface 2
[SwitchC-Vlan-interface2] mpls
[SwitchC-Vlan-interface2] mpls te
[SwitchC-Vlan-interface2] mpls rsvp-te
[SwitchC-Vlan-interface2] mpls rsvp-te hello
[SwitchC-Vlan-interface2] quit
```

4. Configure IS-IS TE

Omitted

5. Configure the MPLS TE tunnel

Omitted

6. Configure RSVP-TE GR

Configure Switch A.

```
<SwitchA> system-view
[SwitchA] mpls
```

```
[SwitchA-mpls] mpls rsvp-te graceful-restart
```

Configure Switch B.

```
<SwitchB> system-view
[SwitchB] mpls
[SwitchB-mpls] mpls rsvp-te graceful-restart
```

Configure Switch C.

```
<SwitchC> system-view
[SwitchC] mpls
[SwitchC-mpls] mpls rsvp-te graceful-restart
```

7. Verify the configuration

After the configuration, a tunnel is created between Switch A and Switch C. Issuing the following command, you should see that the neighbor's GR status is Ready.

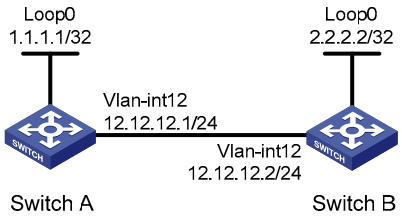
```
<SwitchA> display mpls rsvp-te peer
Interface Vlan-interface1
 Neighbor Addr: 10.1.1.2
 SrcInstance: 880              NbrSrcInstance: 5017
 PSB Count: 0                  RSB Count: 1
 Hello Type Sent: REQ          Neighbor Hello Extension: ENABLE
 SRefresh Enable: NO
 Graceful Restart State: Ready
 Restart Time: 120 Sec         Recovery Time: 300 Sec
```

MPLS RSVP-TE and BFD cooperation configuration example

Network requirements

- Switch A and Switch B are connected directly. Enable MPLS RSVP-TE BFD on the VLAN interfaces connecting the two switches, and run OSPF on the switches to ensure reachability at the network layer.
- If the link between Switch A and Switch B fails, BFD can detect the failure quickly and inform MPLS RSVP-TE of the failure.

Figure 34 Network diagram for MPLS RSVP-TE and BFD cooperation configuration



Procedure

1. Configure basic MPLS RSVP-TE

Configure Switch A.

```
<SwitchA> system-view
[SwitchA] mpls lsr-id 1.1.1.1
[SwitchA] mpls
[SwitchA-mpls] mpls te
```

```
[SwitchA-mpls] mpls rsvp-te
[SwitchA-mpls] quit
[SwitchA] interface vlan-interface 12
[SwitchA-Vlan-interface12] mpls
[SwitchA-Vlan-interface12] mpls te
[SwitchA-Vlan-interface12] mpls rsvp-te
[SwitchA-Vlan-interface12] mpls rsvp-te bfd enable
[SwitchA-Vlan-interface12] quit
```

Configure Switch B.

```
<SwitchB> system-view
[SwitchB] mpls lsr-id 2.2.2.2
[SwitchB] mpls
[SwitchB-mpls] mpls te
[SwitchB-mpls] mpls rsvp-te
[SwitchB-mpls] quit
[SwitchB] interface vlan-interface 12
[SwitchB-Vlan-interface12] mpls
[SwitchB-Vlan-interface12] mpls te
[SwitchB-Vlan-interface12] mpls rsvp-te
[SwitchB-Vlan-interface12] mpls rsvp-te bfd enable
[SwitchB-Vlan-interface12] quit
```

2. Configure OSPF

Configure Switch A.

```
[SwitchA] ospf
[SwitchA-ospf-1] area 0
[SwitchA-ospf-1-area-0.0.0.0] network 12.12.12.1 0.0.0.255
[SwitchA-ospf-1-area-0.0.0.0] network 1.1.1.1 0.0.0.0
[SwitchA-ospf-1-area-0.0.0.0] quit
[SwitchA-ospf-1] quit
```

Configure Switch B.

```
[SwitchB] ospf
[SwitchB-ospf-1] area 0
[SwitchB-ospf-1-area-0.0.0.0] network 12.12.12.2 0.0.0.255
[SwitchB-ospf-1-area-0.0.0.0] network 2.2.2.2 0.0.0.0
[SwitchB-ospf-1-area-0.0.0.0] quit
[SwitchB-ospf-1] quit
```

3. Configure related interfaces on switches

Configure Switch A.

```
[SwitchA] interface vlan-interface 12
[SwitchA-Vlan-interface12] ip address 12.12.12.1 24
[SwitchA-Vlan-interface12] quit
```

Configure Switch B.

```
[SwitchB] interface vlan-interface 12
[SwitchB-Vlan-interface12] ip address 12.12.12.2 24
```

Configure an RSVP-TE tunnel between Switch A and Switch B.

```
[SwitchA] interface tunnel 1
[SwitchA-Tunnel1] ip address 10.10.10.1 24
[SwitchA-Tunnel1] tunnel-protocol mpls te
[SwitchA-Tunnel1] destination 2.2.2.2
[SwitchA-Tunnel1] mpls te tunnel-id 10
[SwitchA-Tunnel1] mpls te signal-protocol rsvp-te
[SwitchA-Tunnel1] mpls te commit
[SwitchA-Tunnel1] return
```

4. Verify the configuration

On Switch A, display the detailed information about the BFD session between Switch A and Switch B.

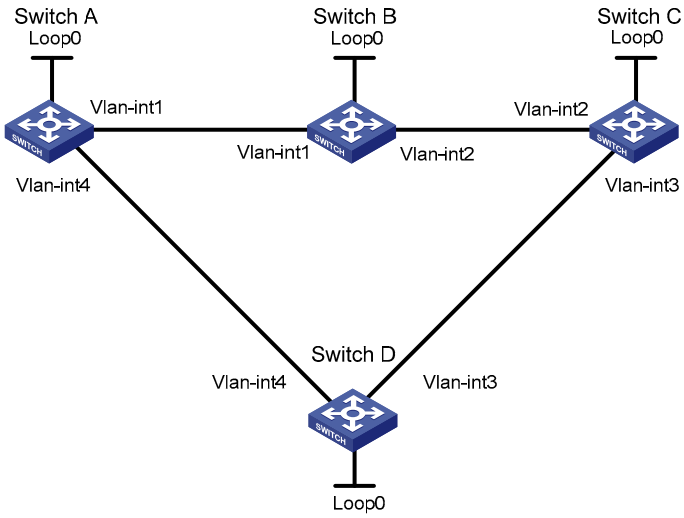
```
<SwitchA> display bfd session verbose

 Total Session Num: 1            Init Mode: Active

 Session Working Under Ctrl Mode:

     Local Discr: 21                 Remote Discr: 20
       Source IP: 12.12.12.1        Destination IP: 12.12.12.2
   Session State: Up                      Interface: Vlan-interface12
Min Trans Inter: 400ms            Act Trans Inter: 400ms
 Min Recv Inter: 400ms           Act Detect Inter: 2000ms
 Running Up for: 00:00:01               Auth mode: None
    Connect Type: Direct                Board Num: 6
        Protocol: RSVP
       Diag Info: No Diagnostic
```

CR-LSP backup configuration example

Network requirements

Set up an MPLS TE tunnel from Switch A to Switch C. Use CR-LSP hot backup for it.

Figure 35 CR-LSP backup



Device	Interface	IP address	Device	Interface	IP address
Switch A	Loop0	1.1.1.9/32	Switch D	Loop0	4.4.4.9/32
	Vlan-int1	10.1.1.1/24		Vlan-int4	30.1.1.2/24
	Vlan-int4	30.1.1.1/24		Vlan-int3	40.1.1.1/24
Switch B	Loop0	2.2.2.9/32	Switch C	Loop0	3.3.3.9/32
	Vlan-int1	10.1.1.2/24		Vlan-int2	20.1.1.2/24
	Vlan-int2	20.1.1.1/24		Vlan-int3	40.1.1.2/24

Procedure

1. Assign IP addresses and masks to interfaces (see Figure 35)

Omitted

2. Configure the IGP protocol

Enable IS-IS to advertise host routes with LSR IDs as destinations on each node. (Omitted)

Perform **display ip routing-table** on each switch. You can see that all nodes have learned the host routes of other nodes with LSR IDs as destinations.

3. Configure MPLS TE basic capabilities, and enable RSVP-TE and CSPF

Configure Switch A.

```
<SwitchA> system-view
[SwitchA] mpls lsr-id 1.1.1.9
[SwitchA] mpls
[SwitchA-mpls] mpls te
[SwitchA-mpls] mpls rsvp-te
[SwitchA-mpls] mpls te cspf
[SwitchA-mpls] quit
```

```
[SwitchA] interface vlan-interface 1
[SwitchA-Vlan-interface1] mpls
[SwitchA-Vlan-interface1] mpls te
[SwitchA-Vlan-interface1] mpls rsvp-te
[SwitchA-Vlan-interface1] quit
[SwitchA] interface vlan-interface 4
[SwitchA-Vlan-interface4] mpls
[SwitchA-Vlan-interface4] mpls te
[SwitchA-Vlan-interface4] mpls rsvp-te
[SwitchA-Vlan-interface4] quit
```

Follow the same steps to configure Switch B, Switch C, and Switch D.

4. Create an MPLS TE tunnel on Switch A.

Configure the MPLS TE tunnel carried on the primary LSP.

```
[SwitchA] interface tunnel 1
[SwitchA-Tunnel1] ip address 9.1.1.1 24
[SwitchA-Tunnel1] tunnel-protocol mpls te
[SwitchA-Tunnel1] destination 3.3.3.9
[SwitchA-Tunnel1] mpls te tunnel-id 10
```

Enable hot LSP backup.

```
[SwitchA-Tunnel1] mpls te backup hot-standby
[SwitchA-Tunnel1] mpls te commit
[SwitchA-Tunnel1] quit
```

Perform **display interface tunnel** on Switch A. You can see that Tunnel1 is up.

```
[SwitchA] display interface tunnel
Tunnel1 current state: UP
Line protocol current state: UP
Description: Tunnel1 Interface
The Maximum Transmit Unit is 64000
Internet Address is 9.1.1.1/24 Primary
Encapsulation is TUNNEL, service-loopback-group ID not set
Tunnel source unknown, destination 3.3.3.9
Tunnel protocol/transport CR_LSP
Output queue : (Urgent queuing : Size/Length/Discards)  0/100/0
Output queue : (Protocol queuing : Size/Length/Discards)  0/500/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
    Last 300 seconds input:  0 bytes/sec, 0 packets/sec
    Last 300 seconds output:  0 bytes/sec, 0 packets/sec
    0 packets input,  0 bytes
    0 input error
    0 packets output,  0 bytes
    0 output error
```

5. Verify the configuration

Perform **display mpls te tunnel** on Switch A. You can see that two tunnels are present with the outgoing interface being VLAN-interface 1 and VLAN-interface 4 respectively. This indicates that a backup CR-LSP was created upon creation of the primary CR-LSP.

```
[SwitchA] display mpls te tunnel
```

```
LSP-Id            Destination     In/Out-If             Name
1.1.1.9:6         3.3.3.9         -/Vlan1               Tunnel1
1.1.1.9:2054      3.3.3.9         -/Vlan4               Tunnel1
```

Perform **display mpls te tunnel path** on Switch A to identify the paths that the two tunnels traverse:

```
[SwitchA] display mpls te tunnel path
 Tunnel Interface Name : Tunnel1
 Lsp ID : 1.1.1.9 :6
 Hop Information
  Hop 0   10.1.1.1
  Hop 1   10.1.1.2
  Hop 2   2.2.2.9
  Hop 3   20.1.1.1
  Hop 4   20.1.1.2
  Hop 5   3.3.3.9
 Tunnel Interface Name : Tunnel1
 Lsp ID : 1.1.1.9 :2054
 Hop Information
  Hop 0   30.1.1.1
  Hop 1   30.1.1.2
  Hop 2   4.4.4.9
  Hop 3   40.1.1.1
  Hop 4   40.1.1.2
  Hop 5   3.3.3.9
```

Perform **tracert** to draw the picture of the path that a packet must travel to reach the tunnel destination.

```
[SwitchA] tracert -a 1.1.1.9 3.3.3.9
 traceroute to  3.3.3.9(3.3.3.9) 30 hops max,40 bytes packet
 1 10.1.1.2 25 ms 30.1.1.2 25 ms 10.1.1.2 25 ms
 2 40.1.1.2 45 ms 20.1.1.2 29 ms 40.1.1.2 54 ms
```

The sample output shows that the current LSP traverses Switch B but not Switch D.

Shut down VLAN-interface 2 on Switch B. Perform **tracert** on Switch A to draw the path to the tunnel destination. The output shows that the LSP is re-routed to traverse Switch D:

```
[SwitchA] tracert -a 1.1.1.9 3.3.3.9
 traceroute to  3.3.3.9(3.3.3.9) 30 hops max,40 bytes packet
 1 30.1.1.2 28 ms  27 ms  23 ms
 2 40.1.1.2 50 ms  50 ms  49 ms
```

Perform **display mpls te tunnel** on Switch A. You can see that only the tunnel traversing Switch D is present:

```
[SwitchA] display mpls te tunnel
LSP-Id            Destination     In/Out-If             Name
1.1.1.9:2054      3.3.3.9         -/Vlan4               Tunnel1
```

Configuring ordinary CR-LSP backup is almost the same as configuring hot CR-LSP backup except that you must replace **mpls te backup hot-standby** with **mpls te backup ordinary**. Unlike in hot CR-LSP backup where a secondary tunnel is created immediately upon creation of a primary tunnel, in ordinary CR-LSP backup, a secondary CR-LSP is created only after the primary LSP goes down.

6. Create a static route for routing MPLS TE tunnel traffic

```
[SwitchA] ip route-static 20.1.1.2 24 tunnel 3 preference 1
```

Perform **display ip routing-table** on Switch A. You can see a static route entry with Tunnel1 as the outgoing interface.

FRR configuration example

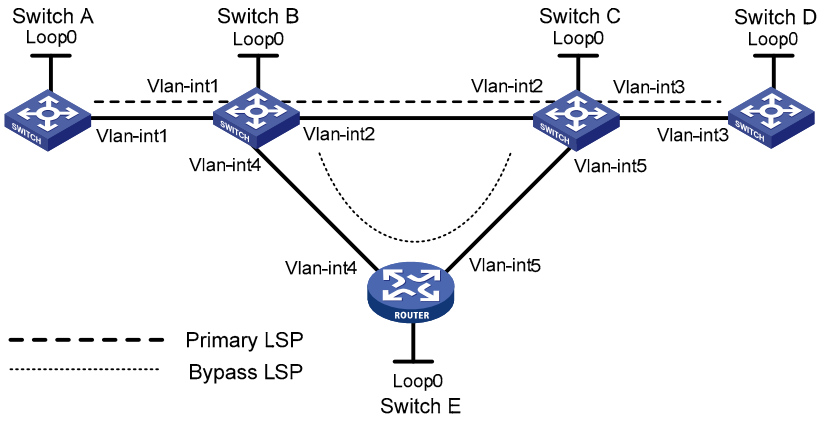
Network requirements

On a primary LSP Switch A →Switch B →Switch C →Switch D, use FRR to protect the link Switch B → Switch C.

Perform the following configurations:

- Create a bypass LSP that traverses the path Switch B →Switch E →Switch C. Switch B is the PLR and Switch C is the MP.
- Explicitly route the primary TE tunnel and the bypass TE tunnel with the signaling protocol being RSVP-TE.

Figure 36 Link protection using the FRR approach



Device	Interface	IP address	Device	Interface	IP address
Switch A	Loop0	1.1.1.1/32	Switch E	Loop0	5.5.5.5/32
	Vlan-int1	2.1.1.1/24		Vlan-int4	3.2.1.2/24
Switch B	Loop0	2.2.2.2/32		Vlan-int5	3.3.1.1/24
	Vlan-int1	2.1.1.2/24	Switch C	Loop0	3.3.3.3/32
	Vlan-int2	3.1.1.1/24		Vlan-int3	4.1.1.1/24
	Vlan-int4	3.2.1.1/24		Vlan-int2	3.1.1.2/24
Switch D	Loop0	4.4.4.4/32		Vlan-int5	3.3.1.2/24
	Vlan-int3	4.1.1.2/24			

Procedure

1. Assign IP addresses and masks to interfaces (see Figure 36)

Omitted

2. Configure the IGP protocol

Enable IS-IS to advertise host routes with LSR IDs as destinations on each node. (Omitted)

Perform **display ip routing-table** on each switch. You can see that all nodes have learned the host routes of other nodes with LSR IDs as destinations. Take Switch A for example:

```
<SwitchA> display ip routing-table
```

```
Routing Tables: Public
         Destinations : 13       Routes : 13
Destination/Mask  Proto  Pre  Cost      NextHop         Interface
      1.1.1.1/32  Direct 0    0         127.0.0.1       InLoop0
      2.1.1.0/24  Direct 0    0         2.1.1.1         Vlan1
      2.1.1.1/32  Direct 0    0         127.0.0.1       InLoop0
      2.2.2.2/32  ISIS   15   10        2.1.1.2         Vlan1
      3.1.1.0/24  ISIS   15   20        2.1.1.2         Vlan1
      3.2.1.0/24  ISIS   15   20        2.1.1.2         Vlan1
      3.3.1.0/24  ISIS   15   30        2.1.1.2         Vlan1
      3.3.3.3/32  ISIS   15   20        2.1.1.2         Vlan1
      4.1.1.0/24  ISIS   15   30        2.1.1.2         Vlan1
      4.4.4.4/32  ISIS   15   30        2.1.1.2         Vlan1
      5.5.5.5/32  ISIS   15   20        2.1.1.2         Vlan1
    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 MPLS TE basic capabilities, and enable RSVP-TE and CSPF.

Configure Switch A.

```
[SwitchA] mpls lsr-id 1.1.1.1
[SwitchA] mpls
[SwitchA-mpls] mpls te
[SwitchA-mpls] mpls rsvp-te
[SwitchA-mpls] mpls te cspf
[SwitchA-mpls] quit
[SwitchA] interface vlan-interface 1
[SwitchA-Vlan-interface1] mpls
[SwitchA-Vlan-interface1] mpls te
[SwitchA-Vlan-interface1] mpls rsvp-te
[SwitchA-Vlan-interface1] quit
```

Configure Switch B.

```
[SwitchB] mpls lsr-id 2.2.2.2
[SwitchB] mpls
[SwitchB-mpls] mpls te
[SwitchB-mpls] mpls rsvp-te
[SwitchB-mpls] mpls te cspf
[SwitchB-mpls] quit
[SwitchB] interface vlan-interface 1
[SwitchB-Vlan-interface1] mpls
[SwitchB-Vlan-interface1] mpls te
[SwitchB-Vlan-interface1] mpls rsvp-te
[SwitchB-Vlan-interface1] quit
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] mpls
[SwitchB-Vlan-interface2] mpls te
[SwitchB-Vlan-interface2] mpls rsvp-te
[SwitchB-Vlan-interface2] quit
[SwitchB] interface vlan-interface 4
```

```
[SwitchB-Vlan-interface4] mpls
[SwitchB-Vlan-interface4] mpls te
[SwitchB-Vlan-interface4] mpls rsvp-te
[SwitchB-Vlan-interface4] quit
```

Follow the same steps to configure Switch C, Switch D, and Switch E.

4. Create an MPLS TE tunnel on Switch A, the headend of the primary LSP

Create an explicit path for the primary LSP.

```
[SwitchA] explicit-path pri-path
[SwitchA-explicit-path-pri-path] next hop 2.1.1.2
[SwitchA-explicit-path-pri-path] next hop 3.1.1.2
[SwitchA-explicit-path-pri-path] next hop 4.1.1.2
[SwitchA-explicit-path-pri-path] next hop 4.4.4.4
[SwitchA-explicit-path-pri-path] quit
```

Configure the MPLS TE tunnel carried on the primary LSP.

```
[SwitchA] interface tunnel 4
[SwitchA-Tunnel4] ip address 10.1.1.1 255.255.255.0
[SwitchA-Tunnel4] tunnel-protocol mpls te
[SwitchA-Tunnel4] destination 4.4.4.4
[SwitchA-Tunnel4] mpls te tunnel-id 10
[SwitchA-Tunnel4] mpls te path explicit-path pri-path preference 1
```

Enable FRR.

```
[SwitchA-Tunnel4] mpls te fast-reroute
[SwitchA-Tunnel4] mpls te commit
[SwitchA-Tunnel4] quit
```

Perform **display interface tunnel** on Switch A. You can see that Tunnel4 is up.

```
[SwitchA] display interface tunnel
Tunnel4 current state: UP
Line protocol current state: UP
Description: Tunnel4 Interface
The Maximum Transmit Unit is 64000
Internet Address is 10.1.1.1/24 Primary
Encapsulation is TUNNEL, service-loopback-group ID not set
Tunnel source unknown, destination 4.4.4.4
Tunnel protocol/transport CR_LSP
Output queue : (Urgent queuing : Size/Length/Discards)  0/100/0
Output queue : (Protocol queuing : Size/Length/Discards)  0/500/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
    Last 300 seconds input:  0 bytes/sec, 0 packets/sec
    Last 300 seconds output:  0 bytes/sec, 0 packets/sec
    0 packets input,  0 bytes
    0 input error
    0 packets output,  0 bytes
    0 output error
```

Perform **display mpls te tunnel-interface** on Switch A to verify the configuration of the tunnel interface.

```
[SwitchA] display mpls te tunnel-interface
  Tunnel Name         :  Tunnel4
```

```
Tunnel Desc         :  Tunnel4 Interface
Tunnel State Desc   :  CR-LSP is Up
Tunnel Attributes   :
  LSP ID              :  1.1.1.1:1
  Session ID          :  10
  Admin State         :  UP               Oper State   :  UP
  Ingress LSR ID      :  1.1.1.1          Egress LSR ID:  4.4.4.4
  Signaling Prot      :  RSVP             Resv Style   :  SE
  Class Type          :  CT0              Tunnel BW    :  0 kbps
  Reserved BW         :  0 kbps
  Setup Priority      :  7                Hold Priority:  7
  Affinity Prop/Mask  :  0/0
  Explicit Path Name  :  pri-path
  Tie-Breaking Policy :  None
  Metric Type         :  None
  Record Route        :  Enabled      Record Label :  Enabled
  FRR Flag            :  Enabled      BackUpBW Flag:  Not Supported
  BackUpBW Type       :  -            BackUpBW     :  -
  Route Pinning       :  Disabled
  Retry Limit         :  10                Retry Interval:  10 sec
  Reopt               :  Disabled          Reopt Freq   :  -
  Back Up Type        :  None
  Back Up LSPID       :  -
  Auto BW             :  Disabled          Auto BW Freq :  -
  Min BW              :  -                 Max BW       :  -
  Current Collected BW:  -
  Interfaces Protected:  -
  VPN Bind Type       :  NONE
  VPN Bind Value      :  -
  Car Policy          :  Disabled
  Tunnel Group        :  Primary
  Primary Tunnel      :  -
  Backup Tunnel       :  -
  Group Status        :  -
  Oam Status          :  -
```

5. Configure a bypass tunnel on Switch B (the PLR)

Create an explicit path for the bypass LSP.

```
[SwitchB] explicit-path by-path
[SwitchB-explicit-path-by-path] next hop 3.2.1.2
[SwitchB-explicit-path-by-path] next hop 3.3.1.2
[SwitchB-explicit-path-by-path] next hop 3.3.3.3
[SwitchB-explicit-path-by-path] quit
```

Create the bypass tunnel.

```
[SwitchB] interface tunnel 5
[SwitchB-Tunnel5] ip address 11.1.1.1 255.255.255.0
[SwitchB-Tunnel5] tunnel-protocol mpls te
[SwitchB-Tunnel5] destination 3.3.3.3
```

```
[SwitchB-Tunnel5] mpls te tunnel-id 15
[SwitchB-Tunnel5] mpls te path explicit-path by-path preference 1
```

Configure the bandwidth that the bypass tunnel protects.

```
[SwitchB-Tunnel5] mpls te backup bandwidth 10000
[SwitchB-Tunnel5] mpls te commit
[SwitchB-Tunnel5] quit
```

Bind the bypass tunnel with the protected interface.

```
[SwitchB] interface Vlan-interface 2
[SwitchB-Vlan-interface2] mpls te fast-reroute bypass-tunnel tunnel 5
[SwitchB-Vlan-interface2] quit
```

Perform **display interface tunnel** on Switch B. You can see that Tunnel5 is up.

Perform **display mpls lsp** on each switch. You can see that two LSPs are traversing Switch B and Switch C.

```
[SwitchA] display mpls lsp
--------------------------------------------------------------------
                LSP Information: RSVP LSP
--------------------------------------------------------------------
FEC                In/Out Label  In/Out IF              Vrf Name
4.4.4.4/32         NULL/1024     -/Vlan1
[SwitchB] display mpls lsp
--------------------------------------------------------------------
                LSP Information: RSVP LSP
--------------------------------------------------------------------
FEC             In/Out Label  In/Out IF                Vrf Name
4.4.4.4/32      1024/1024     Vlan1/Vlan2
3.3.3.3/32      NULL/1024     -/Vla4
[SwitchC] display mpls lsp
--------------------------------------------------------------------
                LSP Information: RSVP LSP
--------------------------------------------------------------------
FEC             In/Out Label  In/Out IF               Vrf Name
4.4.4.4/32      1024/3        Vlan2/Vlan3
3.3.3.3/32      3/NULL        Vlan5/-
[SwitchD] display mpls lsp
--------------------------------------------------------------------
                LSP Information: RSVP LSP
--------------------------------------------------------------------
FEC              In/Out Label  In/Out IF               Vrf Name
4.4.4.4/32       3/NULL        Vlan3/-
[SwitchE] display mpls lsp
--------------------------------------------------------------------
                LSP Information: RSVP LSP
--------------------------------------------------------------------
FEC              In/Out Label  In/Out IF                Vrf Name
3.3.3.3/32       1024/3        Vlan4/Vlan5
```

Perform **display mpls te tunnel** on each switch. You can see that two MPLS TE tunnels are traversing Switch B and Switch C.

```
[SwitchA] display mpls te tunnel
LSP-Id           Destination   In/Out-If          Name
1.1.1.1:1        4.4.4.4       -/Vlan1            Tunnel4
[SwitchB] display mpls te tunnel
LSP-Id           Destination   In/Out-If          Name
1.1.1.1:1        4.4.4.4       Vlan1/Vlan2        Tunnel4
2.2.2.2:1        3.3.3.3       -/Vlan4            Tunnel5
[SwitchC] display mpls te tunnel
LSP-Id           Destination   In/Out-If          Name
1.1.1.1:1        4.4.4.4       Vlan2/Vlan3        Tunnel4
2.2.2.2:1        3.3.3.3       Vlan5/-            Tunnel5
[SwitchD] display mpls te tunnel
LSP-Id           Destination   In/Out-If          Name
1.1.1.1:1        4.4.4.4       Vlan3/-            Tunnel4
[SwitchE] display mpls te tunnel
LSP-Id           Destination   In/Out-If          Name
2.2.2.2:1        3.3.3.3       Vlan4/Vlan5        Tunnel5
```

Perform **display mpls lsp verbose** on Switch B. You can see that the bypass tunnel is bound with the protected interface VLAN-interface 2 and is currently unused.

```
[SwitchB] display mpls lsp verbose
--------------------------------------------------------------------
                LSP Information: RSVP LSP
--------------------------------------------------------------------
  No                  :  1
  IngressLsrID        :  1.1.1.1
  LocalLspID          :  1
  Tunnel-Interface    :  Tunnel4
  Fec                 :  4.4.4.4/32
  Nexthop             :  3.1.1.2
  In-Label            :  1024
  Out-Label           :  1024
  In-Interface        :  Vlan-interface1
  Out-Interface       :  Vlan-interface2
  LspIndex            :  4097
  Tunnel ID           :  0x22001
  LsrType             :  Transit
  Bypass In Use       :  Not Used
  BypassTunnel        :  Tunnel Index[Tunnel5], InnerLabel[1024]
  Mpls-Mtu            :  1500

  No                  :  2
  IngressLsrID        :  2.2.2.2
  LocalLspID          :  1
  Tunnel-Interface    :  Tunnel5
  Fec                 :  3.3.3.3/32
  Nexthop             :  3.2.1.2
  In-Label            :  NULL
  Out-Label           :  1024
```

```
  In-Interface        :  ----------
  Out-Interface       :  Vlan-interface4
  LspIndex            :  4098
  Tunnel ID           :  0x22002
  LsrType             :  Ingress
  Bypass In Use       :  Not Exists
  BypassTunnel        :  Tunnel Index[---]
  Mpls-Mtu            :  1500
```

6. Verify the FRR function

Shut down the protected outgoing interface on PLR.

```
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] shutdown
%Sep  7 08:53:34 2004 SwitchB IFNET/5/UPDOWN:Line protocol on the interface Vlan-interface2
turns into DOWN state
```

Perform **display interface tunnel 4** on Switch A to identify the state of the primary LSP. You can see that the tunnel interface is still up.

Perform **display mpls te tunnel-interface** on Switch A to verify the configuration of the tunnel interface.

```
[SwitchA] display mpls te tunnel-interface
  Tunnel Name         : Tunnel4
  Tunnel Desc         : Tunnel4 Interface
  Tunnel State Desc   : Modifying CR-LSP is setting up
  Tunnel Attributes   :
    LSP ID              :  1.1.1.1:1
    Session ID          :  10
    Admin State         :  UP               Oper State    :  UP
    Ingress LSR ID      :  1.1.1.1          Egress LSR ID:  4.4.4.4
    Signaling Prot      :  RSVP             Resv Style    :  SE
    Class Type          :  CT0              Tunnel BW     :  0 kbps
    Reserved BW         :  0 kbps
    Setup Priority      :  7                Hold Priority:  7
    Affinity Prop/Mask  :  0x0/0x0
    Explicit Path Name  :  pri-path
    Tie-Breaking Policy :  None
    Metric Type         :  None
    Record Route        :  Enabled      Record Label :  Enabled
    FRR Flag            :  Enabled      BackUpBW Flag:  Not Supported
    BackUpBW Type       :  -            BackUpBW     :  -
    Route Pinning       :  Disabled
    Retry Limit         :  10                Retry Interval:  10 sec
    Reopt               :  Disabled          Reopt Freq    :  -
    Back Up Type        :  None
    Back Up LSPID       :  -
    Auto BW             :  Disabled             Auto BW Freq :  -
    Min BW              :  -                    Max BW       :  -
    Current Collected BW:  -
    Interfaces Protected:  -
    VPN Bind Type       :  NONE
```

```
  VPN Bind Value      :  -
  Car Policy          :  Disabled
  Tunnel Group        :  Primary
  Primary Tunnel      :  -
  Backup Tunnel       :  -
  Group Status        :  -
  Oam Status          :  -

Tunnel Name         :  Tunnel4
Tunnel Desc         :  Tunnel4 Interface
Tunnel State Desc   :  Modifying CR-LSP is setting up
Tunnel Attributes   :
  LSP ID              :  1.1.1.1:1025
  Session ID          :  10
  Admin State         :                  Oper State   :  Modified
  Ingress LSR ID      :  1.1.1.1         Egress LSR ID:  4.4.4.4
  Signaling Prot      :  RSVP            Resv Style   :  SE
  Class Type          :  CT0             Tunnel BW    :  0 kbps
  Reserved BW         :  0 kbps
  Setup Priority      :  7               Hold Priority:  7
  Affinity Prop/Mask  :  0x0/0x0
  Explicit Path Name  :  pri-path
  Tie-Breaking Policy :  None
  Metric Type         :  None
  Record Route        :  Enabled      Record Label :  Enabled
  FRR Flag            :  Enabled      BackUpBW Flag:  Not Supported
  BackUpBW Type       :  -            BackUpBW     :  -
  Route Pinning       :  Disabled
  Retry Limit         :  10                Retry Interval:  10 sec
  Reopt               :  Disabled          Reopt Freq   :  -
  Back Up Type        :  None
  Back Up LSPID       :  -
  Auto BW             :  Disabled          Auto BW Freq :  -
  Min BW              :  -                 Max BW       :  -
  Current Collected BW:  -
  Interfaces Protected:  -
  VPN Bind Type       :  NONE
  VPN Bind Value      :  -
  Car Policy          :  Disabled
  Tunnel Group        :  Primary
  Primary Tunnel      :  -
  Backup Tunnel       :  -
  Group Status        :  -
  Oam Status          :  -
```

If you perform the **display mpls te tunnel-interface** command immediately after an FRR protection switch, you are likely to see two CR-LSPs in up state are present. This is normal because the make-before-break mechanism of FRR introduces a delay before removing the old LSP after a new LSP is created.

Perform **display mpls lsp verbose** on Switch B. You can see that the bypass tunnel is in use.

```
[SwitchB] display mpls lsp verbose
--------------------------------------------------------------------
                LSP Information: RSVP LSP
--------------------------------------------------------------------
  No                  :  1
  IngressLsrID        :  1.1.1.1
  LocalLspID          :  1
  Tunnel-Interface    :  Tunnel4
  Fec                 :  4.4.4.4/32
  Nexthop             :  3.1.1.2
  In-Label            :  1024
  Out-Label           :  1024
  In-Interface        :  Vlan-interface1
  Out-Interface       :  Vlan-interface2
  LspIndex            :  4097
  Tunnel ID           :  0x22001
  LsrType             :  Transit
  Bypass In Use       :  In Use
  BypassTunnel        :  Tunnel Index[Tunnel5], InnerLabel[1024]
  Mpls-Mtu            :  1500

  No                  :  2
  IngressLsrID        :  2.2.2.2
  LocalLspID          :  1
  Tunnel-Interface    :  Tunnel5
  Fec                 :  3.3.3.3/32
  Nexthop             :  3.2.1.2
  In-Label            :  NULL
  Out-Label           :  1024
  In-Interface        :  ----------
  Out-Interface       :  Vlan-interface4
  LspIndex            :  4098
  Tunnel ID           :  0x22002
  LsrType             :  Ingress
  Bypass In Use       :  Not Exists
  BypassTunnel        :  Tunnel Index[---]
  Mpls-Mtu            :  1500
```

Set the FRR polling timer to five seconds on PLR.

```
[SwitchB] mpls
[SwitchB-mpls] mpls te timer fast-reroute 5
[SwitchB-mpls] quit
```

Bring the protected outgoing interface up on PLR.

```
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] undo shutdown
%Sep  7 09:01:31 2004 SwitchB IFNET/5/UPDOWN:Line protocol on the interface Vlan-interface2
turns into UP state
```

Perform **display interface tunnel 4** on Switch A to identify the state of the primary LSP. You can see that the tunnel interface is up.

About 5 seconds later, perform **display mpls lsp verbose** on Switch B. You can see that Tunnel5 is still bound with interface VLAN-interface 2 and is unused.

7. Create a static route for routing MPLS TE tunnel traffic

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

Perform **display ip routing-table** on Switch A. You can see a static route entry with Tunnel4 as the outgoing interface.

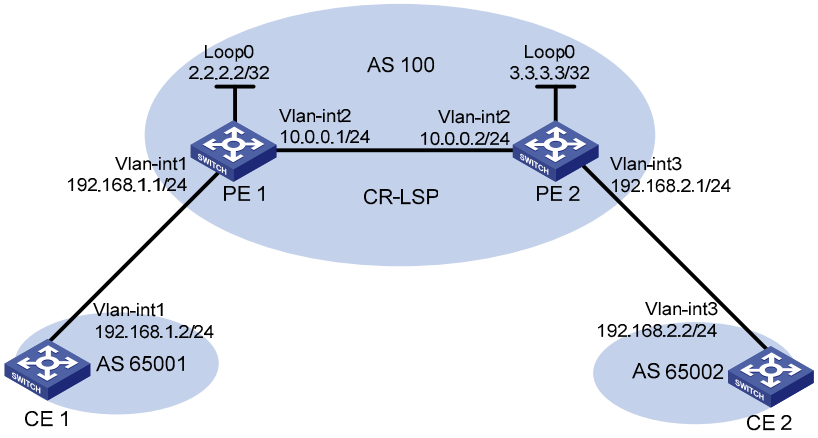
MPLS TE in MPLS L3VPN configuration example

Network requirements

CE 1 and CE 2 belong to VPN 1. They are connected to the MPLS backbone respectively through PE 1 and PE 2. The IGP protocol running on the MPLS backbone is OSPF.

- Set up an MPLS TE tunnel to forward traffic of VPN 1 from PE 1 to PE 2.
- To allow the MPLS L3VPN traffic to travel the TE tunnel, configure a tunneling policy to use a CR-LSP as the VPN tunnel when creating the VPN.

Figure 37 MPLS TE application in VPN



Procedure

1. Configure OSPF, ensuring that PE 1 and PE 2 can learn LSR-ID routes from each other.

Configure PE 1.

```
<PE1> system-view
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 2.2.2.2 255.255.255.255
[PE1-LoopBack0] quit
[PE1] interface vlan-interface 2
[PE1-Vlan-interface2] ip address 10.0.0.1 255.255.255.0
[PE1-Vlan-interface2] quit
[PE1] ospf
[PE1-ospf-1] area 0
[PE1-ospf-1-area-0.0.0.0] network 10.0.0.0 0.0.0.255
[PE1-ospf-1-area-0.0.0.0] network 2.2.2.2 0.0.0.0
```

```
[PE1-ospf-1-area-0.0.0.0] quit
[PE1-ospf-1] quit
```

Configure PE 2.

```
<PE2> system-view
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 3.3.3.3 255.255.255.255
[PE2-LoopBack0] quit
[PE2] interface vlan-interface 2
[PE2-Vlan-interface2] ip address 10.0.0.2 255.255.255.0
[PE2-Vlan-interface2] quit
[PE2] ospf
[PE2-ospf-1] area 0
[PE2-ospf-1-area-0.0.0.0] network 10.0.0.0 0.0.0.255
[PE2-ospf-1-area-0.0.0.0] network 3.3.3.3 0.0.0.0
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit
```

After you complete the configuration, the PEs are able to establish the OSPF neighborship. Perform **display ospf peer**; you can see that the neighborship state is **Full**. Perform **display ip routing-table**; you can see that the PEs have learned the routes to the loopback interfaces of each other. Take PE 1 for example:

```
[PE1] display ospf peer
          OSPF Process 1 with Router ID 2.2.2.2
                  Neighbors
 Area 0.0.0.0 interface 10.0.0.1(Vlan-interface2)'s neighbors
 Router ID: 3.3.3.3          Address: 10.0.0.2         GR State: Normal
   State: Full  Mode:Nbr is  Master  Priority: 1
   DR: None   BDR: None
   Dead timer due in 30  sec
   Neighbor is up for 00:01:00
   Authentication Sequence: [ 0 ]
[PE1] display ip routing-table
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask  Proto  Pre  Cost      NextHop         Interface
      2.2.2.2/32  Direct 0    0         127.0.0.1       InLoop0
      3.3.3.3/32  OSPF   10   1563      10.0.0.2        Vlan2
     10.0.0.0/24  Direct 0    0         10.0.0.1        Vlan2
     10.0.0.1/32  Direct 0    0         127.0.0.1       InLoop0
     10.0.0.2/32  Direct 0    0         10.0.0.2        Vlan2
    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
```

2. Configure MPLS basic capabilities and LDP.

Configure PE 1.

```
[PE1] mpls lsr-id 2.2.2.2
[PE1] mpls
[PE1-mpls] lsp-trigger all
[PE1-mpls] quit
[PE1] mpls ldp
```

```
[PE1-mpls-ldp] quit
[PE1] interface vlan-interface 2
[PE1-Vlan-interface2] mpls
[PE1-Vlan-interface2] mpls ldp
[PE1-Vlan-interface2] quit
```

Configure PE 2.

```
[PE2] mpls lsr-id 3.3.3.3
[PE2] mpls
[PE2-mpls] lsp-trigger all
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface vlan-interface 2
[PE2-Vlan-interface2] mpls
[PE2-Vlan-interface2] mpls ldp
[PE2-Vlan-interface2] quit
```

After you complete the configuration, PEs are able to set up LDP sessions. Perform **display mpls ldp session**; you can see that the session state is operational. Take PE 1 for example:

```
[PE1] display mpls ldp session
               LDP Session(s) in Public Network
Total number of sessions: 1
 ------------------------------------------------------------------
 Peer-ID         Status        LAM  SsnRole  FT   MD5  KA-Sent/Rcv
 ------------------------------------------------------------------
 3.3.3.3:0       Operational   DU   Passive  Off  Off  2/2
 ------------------------------------------------------------------
 LAM : Label Advertisement Mode         FT  : Fault Tolerance
```

3. Enable MPLS TE, CSPF and OSPF TE

Configure PE 1.

```
[PE1] mpls
[PE1-mpls] mpls te
[PE1-mpls] mpls te cspf
[PE1-mpls] quit
[PE1] interface vlan-interface 2
[PE1-Vlan-interface2] mpls te
[PE1-Vlan-interface2] quit
[PE1] ospf
[PE1-ospf-1] opaque-capability enable
[PE1-ospf-1] area 0
[PE1-ospf-1-area-0.0.0.0] mpls-te enable
[PE1-ospf-1-area-0.0.0.0] quit
[PE1-ospf-1] quit
```

Configure PE 2.

```
[PE2] mpls
[PE2-mpls] mpls te
[PE2-mpls] mpls te cspf
```

```
[PE2-mpls] quit
[PE2] interface vlan-interface 2
[PE2-Vlan-interface2] mpls te
[PE2-Vlan-interface2] quit
[PE2] ospf
[PE2-ospf-1] opaque-capability enable
[PE2-ospf-1] area 0
[PE2-ospf-1-area-0.0.0.0] mpls-te enable
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit
```

4. Configure an MPLS TE tunnel

Create a TE tunnel with PE 1 as the headend and PE 2 as the tail. The signaling protocol is CR-LDP.

```
[PE1] interface tunnel 1
[PE1-Tunnel1] ip address 12.1.1.1 255.255.255.0
[PE1-Tunnel1] tunnel-protocol mpls te
[PE1-Tunnel1] destination 3.3.3.3
[PE1-Tunnel1] mpls te tunnel-id 10
[PE1-Tunnel1] mpls te signal-protocol crldp
[PE1-Tunnel1] mpls te commit
[PE1-Tunnel1] quit
```

Perform **display interface tunnel** on PE 1. You can see that the tunnel interface is up.

5. Configure the VPN instance on each PE, and bind it to the interface connected to the CE

Configure on CE 1.

```
<CE1> system-view
[CE1] interface vlan-interface 1
[CE1-Vlan-interface1] ip address 192.168.1.2 255.255.255.0
[CE1-Vlan-interface1] quit
```

Configure the VPN instance on PE 1, and use CR-LSP for VPN setup. Bind the VPN instance with the interface connected to CE 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] tnl-policy policy1
[PE1-vpn-instance-vpn1] quit
[PE1] tunnel-policy policy1
[PE1-tunnel-policy-policy1] tunnel select-seq cr-lsp load-balance-number 1
[PE1-tunnel-policy-policy1] quit
[PE1] interface vlan-interface 1
[PE1-Vlan-interface1] ip binding vpn-instance vpn1
[PE1-Vlan-interface1] ip address 192.168.1.1 255.255.255.0
[PE1-Vlan-interface1] quit
```

Configure on CE 2.

```
<CE2> system-view
[CE2] interface vlan-interface 3
[CE2-Vlan-interface3] ip address 192.168.2.2 255.255.255.0
[CE2-Vlan-interface3] quit
```

Configure the VPN instance on PE 2, and bind it with the interface connected to CE 2.

```
[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] quit
[PE2] interface vlan-interface 3
[PE2-Vlan-interface3] ip binding vpn-instance vpn1
[PE2-Vlan-interface3] ip address 192.168.2.1 255.255.255.0
[PE2-Vlan-interface3] quit
```

Perform **display ip vpn-instance** on the PEs to verify the configuration of the VPN instance. Take PE 1 for example:

```
[PE1] display ip vpn-instance instance-name vpn1
VPN-Instance Name and ID : vpn1, 1
  Create time : 2006/09/27 15:10:29
  Up time : 0 days, 00 hours, 03 minutes and 09 seconds
  Route Distinguisher : 100:1
  Export VPN Targets :  100:1
  Import VPN Targets :  100:1
  Tunnel Policy : policy1
  IPv6 Export VPN Targets :  100:1
  IPv6 Import VPN Targets :  100:1
  IPv6 Import Route Policy : policy1
  Interfaces : Vlan-interface1
```

Ping connected CEs on PEs to test connectivity. For example, ping CE 1 on PE 1:

```
[PE1] ping -vpn-instance vpn1 192.168.1.2
  PING 192.168.1.2: 56  data bytes, press CTRL_C to break
    Reply from 192.168.1.2: bytes=56 Sequence=1 ttl=255 time=47 ms
    Reply from 192.168.1.2: bytes=56 Sequence=2 ttl=255 time=26 ms
    Reply from 192.168.1.2: bytes=56 Sequence=3 ttl=255 time=26 ms
    Reply from 192.168.1.2: bytes=56 Sequence=4 ttl=255 time=26 ms
    Reply from 192.168.1.2: bytes=56 Sequence=5 ttl=255 time=26 ms
  --- 192.168.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 26/30/47 ms
```

The sample output shows that PE 1 can reach CE 1.

6. Configure BGP

Configure CE 1.

```
[CE1] bgp 65001
[CE1-bgp] peer 192.168.1.1 as-number 100
[CE1-bgp] quit
```

Configure PE 1 to establish the EBGP peer relationship with CE 1, and the IBGP peer relationship with PE 2.

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
```

```
[PE1-bgp-vpn1] peer 192.168.1.2 as-number 65001
[PE1-bgp-vpn1] import-route direct
[PE1-bgp-vpn1] quit
[PE1-bgp] peer 3.3.3.3 as-number 100
[PE1-bgp] peer 3.3.3.3 connect-interface loopback1
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 3.3.3.3 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

Configure CE 2.

```
[CE2] bgp 65002
[CE2-bgp] peer 192.168.2.1 as-number 100
[CE2-bgp] quit
```

Configure PE 2 to establish the EGBP peer relationship with CE 2 and the IBGP relationship with PE 1.

```
[PE2] bgp 100
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] peer 192.168.2.2 as-number 65002
[PE2-bgp-vpn1] import-route direct
[PE2-bgp-vpn1] quit
[PE2-bgp] peer 2.2.2.2 as-number 100
[PE2-bgp] peer 2.2.2.2 connect-interface loopback 0
[PE2-bgp] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 2.2.2.2 enable
[PE2-bgp-af-vpnv4] quit
[PE2-bgp] quit
```

Perform **display bgp peer** and **display bgp vpn-instance peer** on PEs. The output shows that the BGP peer relationships have been formed between PEs and between PEs and CEs and have reached the established state. Take PE 1 for example:

```
[PE1-bgp] display bgp peer
 BGP local router ID : 2.2.2.2
 Local AS number : 100
 Total number of peers : 1          Peers in established state : 1
  Peer     V    AS  MsgRcvd  MsgSent  OutQ  Up/Down   State        PrefRcv
  3.3.3.3  4    100       3        3     0  00:00:11  Established  0
[PE1-bgp] display bgp vpn-instance vpn1 peer
 BGP local router ID : 2.2.2.2
 Local AS number : 100
 Total number of peers : 1          Peers in established state : 1
  Peer        V AS   MsgRcvd  MsgSent  OutQ  Up/Down  State        PrefRcv
  192.168.1.2 4 65001      4       5      0  00:02:13 Established  0
```

Ping CE 2 on CE 1 and vice versa to test connectivity.

```
[CE1] ping 192.168.2.2
  PING 192.168.2.2: 56  data bytes, press CTRL_C to break
    Reply from 192.168.2.2: bytes=56 Sequence=1 ttl=253 time=61 ms
    Reply from 192.168.2.2: bytes=56 Sequence=2 ttl=253 time=54 ms
    Reply from 192.168.2.2: bytes=56 Sequence=3 ttl=253 time=53 ms
    Reply from 192.168.2.2: bytes=56 Sequence=4 ttl=253 time=57 ms
```

```
    Reply from 192.168.2.2: bytes=56 Sequence=5 ttl=253 time=36 ms
  --- 192.168.2.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 36/52/61 ms
[CE2] ping 192.168.1.2
  PING 192.168.1.2: 56  data bytes, press CTRL_C to break
    Reply from 192.168.1.2: bytes=56 Sequence=1 ttl=253 time=38 ms
    Reply from 192.168.1.2: bytes=56 Sequence=2 ttl=253 time=61 ms
    Reply from 192.168.1.2: bytes=56 Sequence=3 ttl=253 time=74 ms
    Reply from 192.168.1.2: bytes=56 Sequence=4 ttl=253 time=36 ms
    Reply from 192.168.1.2: bytes=56 Sequence=5 ttl=253 time=35 ms
  --- 192.168.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 35/48/74 ms
```

The sample output shows that CE 1 and CE 2 can reach each other.

7. Verify the configuration

Perform **display mpls lsp verbose** on PE 1. You can see an LSP with LspIndex 2050. This is the LSP, or, the MPLS TE tunnel, established using CR-LDP.

```
[PE1] display mpls lsp verbose
------------------------------------------------------------------------
                 LSP Information: CRLDP LSP
------------------------------------------------------------------------
  No                  :  1
  IngressLsrID        :  2.2.2.2
  LocalLspID          :  1
  Tunnel-Interface    :  Tunnel1
  Fec                 :  3.3.3.3/32
  Nexthop             :  10.0.0.2
  In-Label            :  NULL
  Out-Label           :  1024
  In-Interface        :  ----------
  Out-Interface       :  Vlan-interface2
  LspIndex            :  2050
  Tunnel ID           :  0x22004
  LsrType             :  Ingress
  Bypass In Use       :  Not Exists
  BypassTunnel        :  Tunnel Index[---]
  Mpls-Mtu            :  1500
------------------------------------------------------------------------
                 LSP Information: BGP  LSP
------------------------------------------------------------------------
  No                  :  2
  VrfIndex            :  vpn1
```

```
  Fec                 :  192.168.1.0/24
  Nexthop             :  192.168.1.1
  In-Label            :  1024
  Out-Label           :  NULL
  In-Interface        :  ----------
  Out-Interface       :  ----------
  LspIndex            :  8193
  Tunnel ID           :  0x0
  LsrType             :  Egress
  Outgoing Tunnel ID  :  0x0
  Label Operation     :  POP
------------------------------------------------------------------
                LSP Information: LDP LSP
------------------------------------------------------------------
  No                  :  3
  VrfIndex            :
  Fec                 :  2.2.2.2/32
  Nexthop             :  127.0.0.1
  In-Label            :  3
  Out-Label           :  NULL
  In-Interface        :  Vlan-interface2
  Out-Interface       :  ----------
  LspIndex            :  10241
  Tunnel ID           :  0x0
  LsrType             :  Egress
  Outgoing Tunnel ID  :  0x0
  Label Operation     :  POP

  No                  :  4
  VrfIndex            :
  Fec                 :  3.3.3.3/32
  Nexthop             :  10.0.0.2
  In-Label            :  NULL
  Out-Label           :  3
  In-Interface        :  ----------
  Out-Interface       :  Vlan-interface2
  LspIndex            :  10242
  Tunnel ID              0x22000
  LsrType             :  Ingress
  Outgoing Tunnel ID  :  0x0
  Label Operation     :  PUSH
```

Perform **display interface tunnel** on PE 1. The output shows that traffic is being forwarded along the CR-LSP of the TE tunnel.

```
[PE1] display interface tunnel 1
Tunnel1 current state: UP
Line protocol current state: UP
Description: Tunnel1 Interface
The Maximum Transmit Unit is 1500
```

```
Internet Address is 12.1.1.1/24 Primary
Encapsulation is TUNNEL, service-loopback-group not set
Tunnel source unknown, destination 3.3.3.3
Tunnel protocol/transport CR_LSP
Output queue : (Urgent queuing : Size/Length/Discards)  0/100/0
Output queue : (Protocol queuing : Size/Length/Discards)  0/500/0
Output queue : (FIFO queuing : Size/Length/Discards)  0/75/0
    Last 300 seconds input:  5 bytes/sec, 0 packets/sec
    Last 300 seconds output:  5 bytes/sec, 0 packets/sec
    34 packets input,  2856 bytes
    0 input error
    34 packets output,  2856 bytes
    0 output error
```

Troubleshooting MPLS TE

No TE LSA generated

Symptom:

OSPF TE is configured but no TE LSAs can be generated to describe MPLS TE attributes.

Analysis:

For TE LSAs to be generated, at least one OSPF neighbor must reach the FULL state.

Solution:

1. Perform **display current-configuration** to check that MPLS TE is configured on involved interfaces.
2. Perform **debugging ospf mpls-te** to observe whether OSPF can receive the TE LINK establishment message.
3. Perform **display ospf peer** to check that OSPF neighbors are established correctly.

Configuring VPLS

The A5820X switch series do not support VPLS.

VPLS, also called TLS or "virtual private switched network service", can deliver a point-to-multipoint L2VPN service over public networks. With VPLS, geographically-dispersed sites can interconnect and communicate over MAN or WAN as if they were on the same LAN.

VPLS provides Layer 2 VPN services. However, it supports multipoint services, rather than the point-to-point services that traditional VPN supports. With VPLS, service providers can create on the PEs a series of virtual switches for customers, allowing customers to build their LANs across the MAN or WAN.

VPLS operation

CE—Customer edge device directly connected with the service provider network.

PE—Provider edge device that connects one or more CEs to the service provider network. A PE maps and forwards packets between private networks and public network tunnels. A PE can be a UPE or NPE.

UPE—User facing provider edge device that functions as the user access convergence device.

NPE—Network provider edge device that functions as the network core PE. An NPE resides at the edge of a VPLS network core domain and provides transparent VPLS transport services between core networks.

VSI—Virtual switch instance that maps actual access links to virtual links.

PW—Pseudo wire is a bidirectional virtual connection between VSIs. A PW consists of two unidirectional MPLS VCs.

AC—Attachment circuit that connects the CE to the PE. It can use physical interfaces or virtual interfaces. Usually, all user packets on an AC, including Layer 2 and Layer 3 protocol messages, must be forwarded to the peer site without being changed.

QinQ—802.1Q in 802.1Q, a tunneling protocol based on 802.1Q. It offers a point-to-multipoint L2VPN service mechanism. With QinQ, the private network VLAN tags of packets are encapsulated into the public network VLAN tags, allowing packets to be transmitted with two layers of tags across the service provider network. This provides a simpler Layer 2 VPN tunneling service.

Forwarders—A forwarder functions as the VPLS forwarding table. Once a PE receives a packet from an AC, the forwarder selects a PW for forwarding the packet.

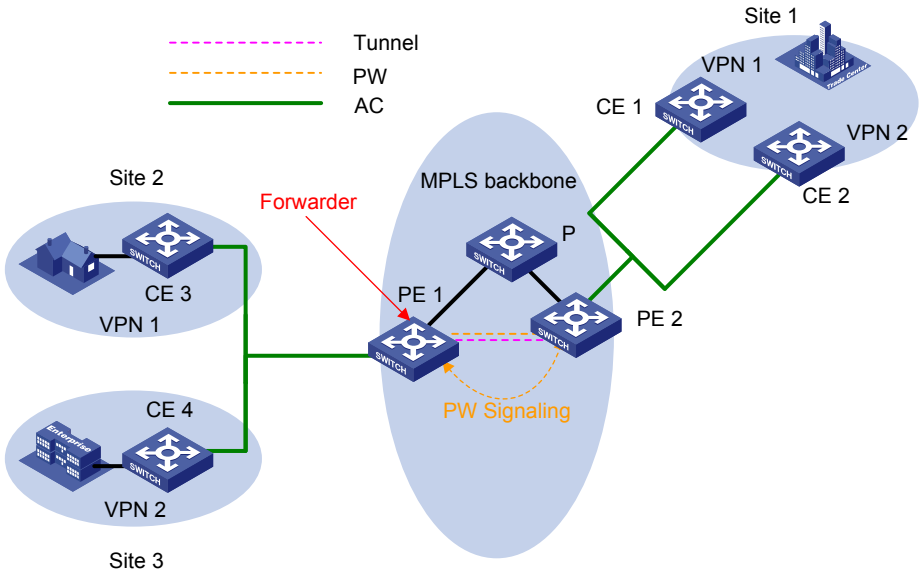
Tunnel—A tunnel, usually an MPLS tunnel, is a direct channel between a local PE and the peer PE for transparent data transmission in-between. It is used to carry PWs. A tunnel can carry multiple PWs.

Encapsulation—Packets transmitted over a PW use the standard PW encapsulation formats and technologies: Ethernet and VLAN.

PW signaling—The PW signaling protocol is the fundament of VPLS. It is used for creating and maintaining PWs and automatically discovering VSI peer PE. Two PW signaling protocols are available: LDP and BGP.

Figure 38 shows a typical VPLS networking scenario.

Figure 38 Network diagram for VPLS



MAC address learning and flooding

VPLS provides reachability by MAC address learning. Each PE maintains a MAC address table.

1. Source MAC address learning

MAC address learning includes two parts:

- Remote MAC address learning associated with PWs

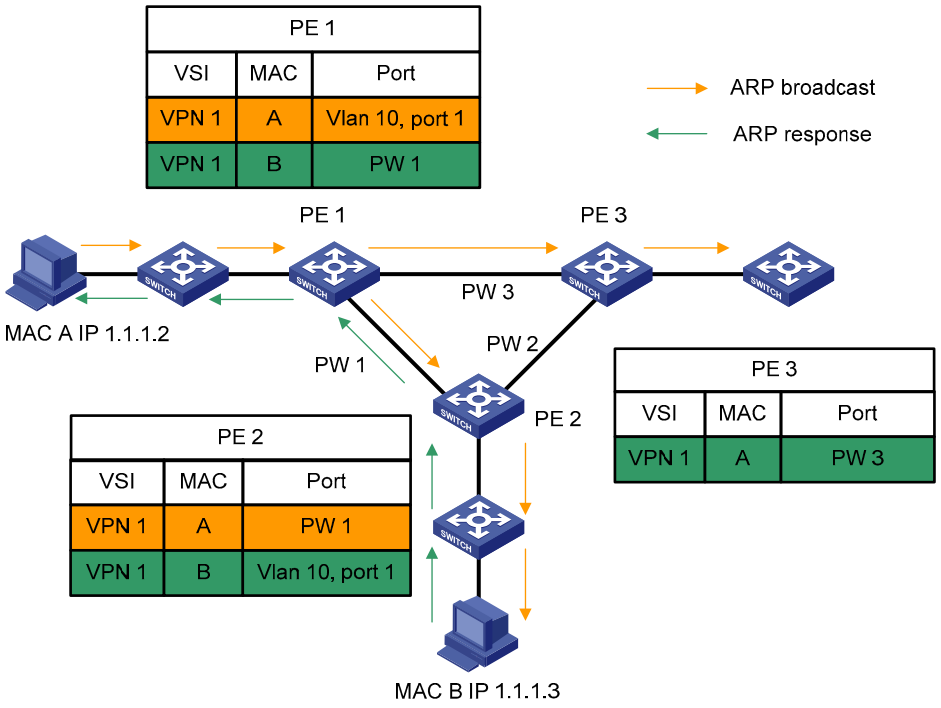
A PW consists of two unidirectional VC LSPs. A PW is up only when both of the VC LSPs are up. When the inbound VC LSP learns a new MAC address, the PW needs to map the MAC address to the outbound VC LSP.

- Local MAC address learning of interfaces directly connected with users

This refers to learning source MAC addresses from Layer 2 packets originated by CEs. This occurs on the corresponding VSI interfaces.

Figure 39 shows the procedure of MAC address learning and flooding on PEs.

Figure 39 MAC learning and flooding on PEs



2. MAC address reclaim

Dynamic address learning must support refreshing and relearning. The VPLS draft defines a dynamic address learning method that uses the address reclaim message, which carries MAC TLV. Upon receiving such a message, a device removes MAC addresses or relearns them according to the specified parameters in the TLV. If NULL is specified, the device removes all MAC addresses of the VSI except for those learned from the PW that received the address reclaim message.

The address reclaim message is very useful when the network topology changes and you must remove the learned MAC addresses quickly. There are two types of address reclaim messages: those with MAC address lists and those without MAC address lists.

After a backup link becomes active and a message with the instruction of relearning MAC entries arrives, a PE updates the corresponding MAC entries in the FIB table of the VPLS instance and sends the message to other PEs that are directly connected through LDP sessions. If the message contains a null MAC address TLV list, these PEs remove all MAC addresses from the specified VSI, except for those learned from the PW that sent the message.

3. MAC address aging

Remote MAC addresses learned by a PE that are related to VC labels but no more in use must be aged out by an aging mechanism. The aging mechanism used here is the aging timer corresponding to the MAC address. When receiving a packet whose source MAC address has an aging timer started, the PE resets the aging timer.

VPLS loop avoidance

To avoid loops in a VPLS network, full mesh and split horizon forwarding are used instead of STP at the private network side.

- Full mesh: PEs are logically fully meshed (so are PWs). Each PE must create for each VPLS forwarding instance a tree to all other PEs of the instance.

- Split horizon forwarding: Each PE must support horizontal split to avoid loops. A PE cannot forward packets through PWs of the same VSI, because all PEs of a VSI are directly connected. Packets from PWs on the public network side cannot be forwarded to other PWs; they can only be forwarded to the private network side.

Peer PE discovery and PW signaling protocol

For PEs in the same VSI, you can configure the peer PE addresses or use an automatic discovery mechanism. LDP and BGP are used to automatically discover VSI peer PEs.

For a PW to be created, a PW signaling protocol is needed to assign a multiplex distinguishing flag (or, VC label) and advertise the assigned VC flag to the peer. In addition, the PW signaling protocol advertises VPLS system parameters such as PW ID, control word, and interface parameters. With the PW signaling protocol, PWs can be established between PEs to form a fully meshed network to provide VPLS services. LDP and BGP can be used as PW signaling protocols.

VPLS can be one of the following based on the PW signaling protocol used:

- **LDP VPLS**—Uses LDP as the signaling protocol. This mode is also called the "Martini mode".
- **BGP VPLS**—Uses BGP extension as the signaling protocol. This mode is also called the "Kompella mode".

For more information about the Martini mode and Kompella mode, see "Configuring MPLS L2VPN."

VPLS packet encapsulation

Packet encapsulation on an AC

The packet encapsulation type of an AC depends on the user VSI access mode, which can be VLAN or Ethernet.

- **VLAN access**—The Ethernet header of a packet sent by a CE to a PE or sent by a PE to a CE includes a VLAN tag that is added in the header as a service delimiter for the service provider network to identify the user. The tag is called a "P-Tag".
- **Ethernet access**—The Ethernet header of a packet upstream from the CE or downstream from the PE does not contain any service delimiter. If a header contains a VLAN tag, it is the internal VLAN tag of the user and means nothing to the PE. This kind of internal VLAN tag of the user is called a "U-Tag".

You can specify the VSI access mode to be used.

Packet encapsulation on a PW

The packet encapsulation type of a PW, also called the "PW transport mode", can be either Ethernet or VLAN.

- In Ethernet mode, P-TAG is not transferred on the PW. For a packet from a CE, if it contains the service delimiter, the PE removes the service delimiter and adds two levels of MPLS labels into the packet before forwarding the packet. Otherwise, the PE directly adds two levels of MPLS labels into the packet and then forwards the packet. For a packet to be sent downstream, whether the PE adds the service delimiter into the packet depends on your configuration. However, rewriting and removing of existing tags are not allowed.
- In VLAN mode, every packet to the PW must carry a P-TAG. For a packet from a CE, if it contains the service delimiter, the PE directly adds two levels of MPLS labels into the packet and sends the packet out. Otherwise, the PE adds a null tag together with two levels of MPLS labels into the packet and sends the packet out. For a packet to be sent downstream, the PE rewrites, removes, or retains the service delimiter depending on your configuration.

According to the protocol, the packet encapsulation type of a PW is VLAN by default.

H-VPLS implementation

H-VPLS can extend the VPLS access range of a service provider and reduce costs.

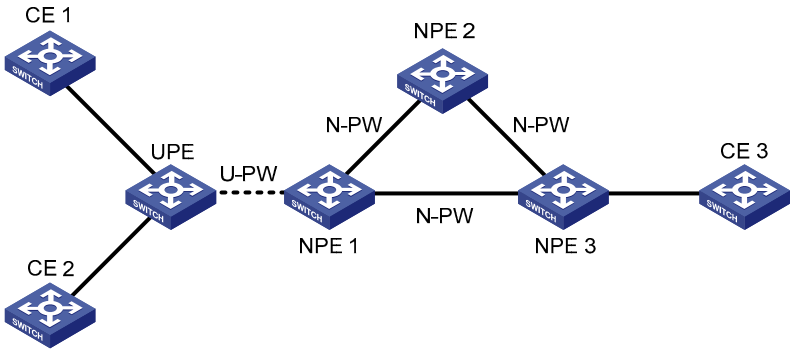
Advantages of H-VPLS access

- H-VPLS has lower requirements on the MTU-s. It has distinct hierarchies which fulfill definite tasks.
- H-VPLS reduces the logical complexity of the fully meshed network consisting of PEs and the configuration complexity.

Two H-VPLS access modes

1. H-VPLS with LSP access

Figure 40 H-VPLS with LSP access



As shown in Figure 40, UPE functions as the convergence device MTU-s and establishes only a virtual link U-PW with NPE 1. It does not establish virtual links with any other peers.

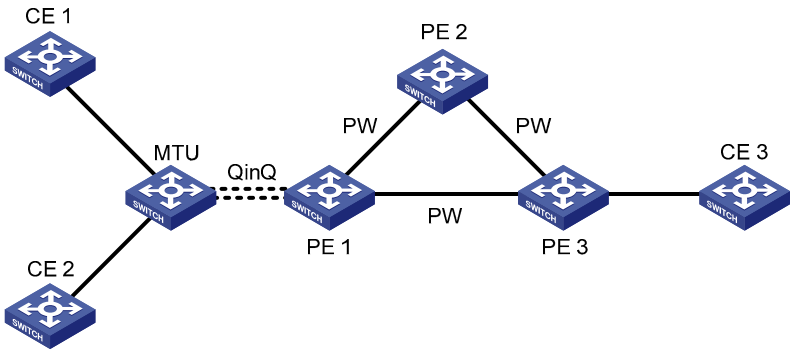
In H-VPLS with LSP access, data is forwarded in the following procedure:

- UPE tags a packet received from a CE with the MPLS label for the U-PW, and then sends the packet to NPE 1.
- When receiving the packet, NPE 1 determines which VSI the packet belongs to by the label and, based on the destination MAC address of the packet, tags the packet with the multiplex distinguishing flag for the N-PW, and forwards the packet.
- When receiving a packet from the N-PW, NPE 1 tags the packet with the multiplex distinguishing flag for the U-PW and sends the packet to UPE, which forwards the packet to the CE.

For packets to be exchanged between CE 1 and CE 2, UPE can forward them directly without NPE 1 because it holds the bridging function by itself. For the first packet with an unknown destination MAC address or a broadcast packet, UPE broadcasts the packet to CE 2 through the bridging function and, at the same time, forwards it through U-PW to NPE 1, which replicates the packet and sends a copy to each peer CE.

2. H-VPLS with QinQ access

Figure 41 H-VPLS with QinQ access



As shown in Figure 41, MTU is a standard bridging device and QinQ is enabled on its interfaces connected with CEs.

In H-VPLS with QinQ access, data is forwarded in the following procedure:

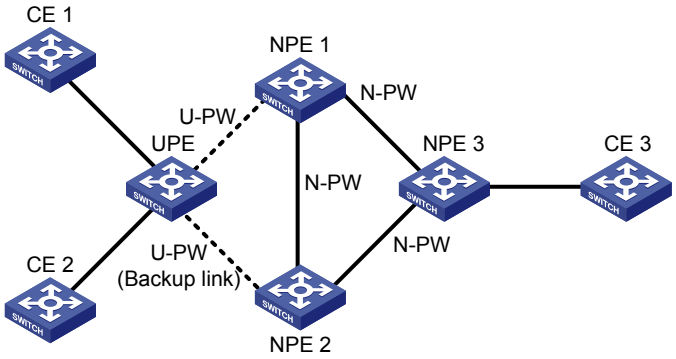
- MTU labels a packet received from a CE with a VLAN tag, and transparently sends the packet to PE 1 through the QinQ tunnel.
- When receiving the packet, PE 1 determines which VSI the packet belongs to by the VLAN tag and, based on the destination MAC address of the packet, tags the packet with the MPLS label for the PW. Then, it forwards the packet.
- When receiving a packet from the PW, PE 1 determines to which VSI the packet belongs by the MPLS label and, based on the destination MAC address of the packet, labels the packet with the VLAN tag. Then, it forwards the packet through the QinQ tunnel to MTU, which in turn forwards the packet to the CE.

For packets to be exchanged between CE 1 and CE 2, MTU can forward them directly without PE 1 because it holds the bridging function by itself. For the first data packet with an unknown destination MAC address or a broadcast packet, MTU broadcasts the packet to CE 2 through the bridging function and, at the same time, forwards it through the QinQ tunnel to PE 1, which replicates the packet and sends a copy to each peer CE.

3. PW switchover

The network design with a single PW between a UPE and an NPE has a distinct drawback: once the PW experiences a failure, all VPNs connected to the aggregate device loses connectivity. The H-VPLS with LSP access provides redundant links for PW backup. Normally, only the primary PW link is used. When the main link fails, the backup link takes over the VPN services, as shown in Figure 42.

Figure 42 Backup link for H-VPLS with LSP access



The H-VPLS with LSP access activates the backup link when:

- The tunnel over which the primary PW is established is deleted, causing the PW to go down.
- BFD detects a main link failure.
- The LDP session between the peers of the primary PW goes down, and the PW is deleted as a result.

VPLS configuration task list

Complete the following tasks to configure VPLS:

Task		Remarks	
Configuring LDP VPLS	Enabling L2VPN and MPLS L2VPN	Required.	Configure either type of VPLS as needed.
	Configuring an LDP VPLS instance	Required.	
	Binding an LDP VPLS instance	Required.	
Configuring BGP VPLS	Configuring the BGP extension	Required.	
	Enabling L2VPN and MPLS L2VPN	Required.	
	Configuring a BGP VPLS instance	Required.	
	Binding a BGP VPLS instance	Required.	
Configuring MAC address learning		Optional.	
Configuring MAC address transition		Optional.	
Configuring VPLS instance attributes		Optional.	
Inspecting PWs		Optional.	

Configuring LDP VPLS

Prerequisites

- Configure IGP on the MPLS backbone devices (PEs and P devices) to guarantee the IP connectivity of the MPLS backbone. For configuration details, see *Layer 3—IP Routing Configuration Guide*.
- Configure MPLS basic capability on the MPLS backbone devices (PEs and P devices) to establish LSP tunnels on the backbone network. For configuration information, see "Configuring MPLS basics"
- Configure LDP remote peers on PEs to establish remote LDP sessions. For configuration information, see "Configuring MPLS basics."

Enabling L2VPN and MPLS L2VPN

You must enable L2VPN and MPLS L2VPN before you can perform VPLS related configurations.

To enable L2VPN and MPLS L2VPN:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enable L2VPN and enter L2VPN view.	**l2vpn**	Required.

Step	Command	Remarks
3. Enable MPLS L2VPN.	**mpls l2vpn**	Required.

For detailed information about **l2vpn** and **mpls l2vpn**, see *MPLS Command Reference*.

Configuring an LDP VPLS instance

When creating an LDP VPLS instance, you must perform the following configurations:

1. Specify a globally unique name for the VPLS instance and set the peer discovery mechanism to manual configuration.
2. Configure LDP as the signaling protocol to be used.
3. Specify the ID of the VPLS instance.
4. Use **peer** to create the VPLS peer PE for the instance, specifying:

- IP address of the peer PE.
- ID of the PW to the peer PE, which must be consistent with that specified on the peer PE.
- Type of the peer PE. If you specify a peer as a UPE, the peer is a user access convergence device in the H-VPLS model. If you specify the **backup-peer** keyword when creating the peer, the local PE is a UPE and you create a primary NPE and a secondary NPE on it. On a UPE, you can configure only one pair of primary and secondary NPEs. The specified remote NPE peers must be fully meshed, while it is not necessary for a UPE to connect with all NPEs.
- PW class template to be referenced. A PW class template defines the PW transport mode and tunneling policy to be used.

To configure an LDP VPLS instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a PW class template and enter its view.	**pw-class** *pw-class-name*	Optional. By default, no PW class template is created.
3. Configure the PW transport mode.	**trans-mode** { **ethernet** | **vlan** }	Optional. VLAN by default.
4. Specify a tunneling policy.	**pw-tunnel-policy** *policy-name*	Optional. By default, the tunneling policy specified through **tnl-policy** in VSI view is used.
5. Return to system view.	**quit**	—
6. Create an LDP VPLS instance and enter VSI view.	**vsi** *vsi-name* **static**	Required.
7. Specify LDP as the PW signaling protocol and enter VSI LDP view.	**pwsignal ldp**	Required.
8. Specify an ID for the VPLS instance.	**vsi-id** *vsi-id*	Required.

Step	Command	Remarks
9. Create a peer PE for the VPLS instance.	**peer** *ip-address* [**pw-id** *pw-id*] [**upe** | **backup-peer** *ip-address* [**backup-pw-id** *pw-id*]] [**pw-class** *class-name*]	Required.
10. Enable the PW switchback function and set the switchback delay time.	**dual-npe revertive** [**wtr-time** *wtr-time*]	Optional. Disabled by default.

Binding an LDP VPLS instance

You can bind a Layer 2 Ethernet interface and a VLAN with the VPLS instance. After you configure such a binding, the VPLS connection services packets that carry the specified VLAN tag and are received on the specified Layer 2 Ethernet interface.

To bind a Layer 2 Ethernet interface and a VLAN with a VPLS instance, you must create a service instance on the Layer 2 Ethernet interface, configure a packet matching rule for the service instance, and bind the service instance with the VPLS instance. After these configurations, packets that arrive at the Layer 2 Ethernet interface and match the packet matching rule are forwarded through the bound VPLS instance.

To bind a service instance with an LDP VPLS instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter the view of the interface connecting a CE.	**interface** *interface-type interface-number*	—
3. Create a service instance and enter its view.	**service-instance** *service-instance-id*	Required. By default, no service instance is created.
4. Configure a packet matching rule for the service instance.	**encapsulation** { **port-based** | **s-vid** *vlan-id* [**only-tagged**] | **tagged** | **untagged** }	Required. By default, no packet matching rule is configured for a service instance.
5. Associate the service instance with a VPLS instance.	**xconnect vsi** *vsi-name* [**access-mode** { **ethernet** | **vlan** }]	Required. By default, a service instance is not associated with any VPLS instance.

Configuring BGP VPLS

Prerequisites

- Configure IGP on the MPLS backbone devices (PEs and P devices) to guarantee the IP connectivity of the MPLS backbone. For configuration details, see *Layer 3—IP Routing Configuration Guide*.
- Configure MPLS basic capability on the MPLS backbone devices (PEs and P devices) to establish LSP tunnels on the backbone network. For configuration information, see "Configuring MPLS basics."

Configuring the BGP extension

Before configuring BGP VPLS, you must configure BGP parameters on the PEs. For configuration details, see *Layer 3—IP Routing Configuration Guide*.

To configure BGP extension:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Enter VPLS address family view.	**vpls-family**	Required.
4. Activate a peer.	**peer** *peer-address* **enable**	Required. No peer is activated by default.

For configurations in VPLS address family view, see "Configuring MPLS L3VPN."

Enabling L2VPN and MPLS L2VPN

You must enable L2VPN and MPLS L2VPN before you can configure VPLS related configurations.

To enable L2VPN and MPLS L2VPN:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enable L2VPN and enter L2VPN view.	**l2vpn**	Required.
3. Enable MPLS L2VPN.	**mpls l2vpn**	Required.

For detailed information about **l2vpn** and **mpls l2vpn**, see *MPLS Command Reference*.

Configuring a BGP VPLS instance

When creating a BGP VPLS instance, you must specify a globally unique name for the VPLS instance and set the peer discovery mechanism to automatic configuration.

When configuring a BGP VPLS instance, you must configure BGP as the signaling protocol to be used.

To configure a BGP VPLS instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a BGP VPLS instance and enter VSI view.	**vsi** *vsi-name* **auto**	Required.
3. Specify BGP as the PW signaling protocol and enter VSI BGP view.	**pwsignal bgp**	Required.
4. Configure an RD for the VPLS instance.	**route-distinguisher** *route-distinguisher*	Required.

Step	Command	Remarks
5. Configure VPN targets for the VPLS instance.	**vpn-target** *vpn-target*&<1-16> [**both** | **import-extcommunity** | **export-extcommunity**]	Required.
6. Create a site for the VPLS instance.	**site** *site-id* [**range** *site-range*] [**default-offset** { 0 | 1 }]	Required.

Binding a BGP VPLS instance

See "Binding an LDP VPLS instance."

Configuring MAC address learning

To configure the MAC address learning function:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter VSI view.	**vsi** *vsi-name*	—
3. Enable/disable MAC address learning for the VPLS instance.	**mac-learning** { **enable** | **disable** }	Optional. Enabled by default.
4. Configure the maximum number of MAC addresses to be learned.	**mac-table limit** *mac-limit-number*	Optional. 524288 by default.

Configuring MAC address transition

When MAC address transition is enabled on a PE, the source MAC address of a packet incoming from a port that is different from the port in the existing MAC entry that contains the source MAC address is added into the MAC address table of the VPLS instance of the incoming port.

When MAC address transition is disabled, the source MAC address of such a packet is not added to the MAC address table of the VPLS instance of the incoming port if the timeout timer of that existing MAC address entry does not expire.

You can disable MAC address transition to block illegal users that use spoofed MAC addresses.

To enable MAC address transition:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter VSI view.	**vsi** *vsi-name*	—
3. Enable MAC address transition.	**mac-move enable**	Optional. Enabled by default.

Configuring VPLS instance attributes

To configure VPLS instance attributes:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter VSI view.	**vsi** *vsi-name*	—
3. Specify the encapsulation type of the VPLS instance.	**encapsulation** { **bgp-vpls** | **ethernet** | **vlan** }	Optional. **vlan** by default, which corresponds to the VSI PW encapsulation type of VLAN.
4. Set the description of the VPLS instance.	**description** *text*	Optional. No description set by default.
5. Shut down the VPLS service of the VPLS instance.	**shutdown**	Optional. Enabled by default.
6. Specify a tunneling policy for the VPLS instance.	**tnl-policy** *tunnel-policy-name*	Optional. By default, no tunneling policy is specified for a VPLS instance and a VPLS instance uses the default tunneling policy. The default tunneling policy selects only one tunnel in this order: LSP tunnel, GRE tunnel, CR-LSP tunnel. For how to configure a tunneling policy, see "MPLS L3VPN configuration."

Inspecting PWs

On a VPLS network, you can use the MPLS LSP ping function to check PW connectivity and get necessary information for troubleshooting PW failures.

On the local PE, the MPLS LSP ping function adds the label of the PW to be inspected into MPLS Echo Request messages so that the messages travel along the PW. The local PE determines whether the PW is valid and reachable to the peer PE according to the replies received from the peer PE.

Use the MPLS LSP ping function to check the connectivity of a PW:

Task	Command
Use MPLS LSP ping to check the connectivity of a PW.	**ping lsp** [**-a** *source-ip* | **-c** *count* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *wait-time* | **-r** *reply-mode* | **-s** *packet-size* | **-t** *time-out* | **-v**] * **pw** *ip-address* **pw-id** *pw-id*

MPLS LSP ping can be used to inspect only an LDP PW.

Displaying and maintaining VPLS

Task	Command	Remarks
Display the VPLS information in the BGP routing table.	**display bgp vpls** { **all** | **group** [*group-name*] | **peer** [[*ip-address*] **verbose**] | **route-distinguisher** *route-distinguisher* [**site-id** *site-id* [**label-offset** *label-offset*]] } [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display the MAC address table information of one or all VPLS instances.	**display mac-address vsi** [*vsi-name*] [**blackhole** | **dynamic** | **static**] [**count**] [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about VPLS connections.	**display vpls connection** [**bgp** | **ldp** | **vsi** *vsi-name*] [**block** | **down** | **up**] [**verbose**] [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display the AC information of one or all VPLS instances.	**display mpls l2vpn fib ac vpls** [**vsi** *vsi-name* | **interface** *interface-type interface-number* [**service-instance** *service-instanceid*]] [**slot** *slot-number*] [**verbose**] [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display the PW information of one or all VPLS instances.	**display mpls l2vpn fib pw vpls** [**vsi** *vsi-name* [**link** *link-id*]] [**slot** *slot-number*] [**verbose**] [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about one or all VPLS instances.	**display vsi** [*vsi-name*] [**verbose**] [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about remote VPLS connections.	**display vsi remote** { **bgp** | **ldp** } [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about one or all PW class templates.	**display pw-class** [*pw-class-name*] [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Clear the MAC address table of one or all VPLS instances.	**reset mac-address vsi** [*vsi-name*]	Available in user view.

Resetting VPLS

Task	Command	Remarks
Reset a specified or all VPLS BGP connections.	**reset bgp vpls** { *as-number* | *ip-address* | **all** | **external** | **internal** }	Available in user view.

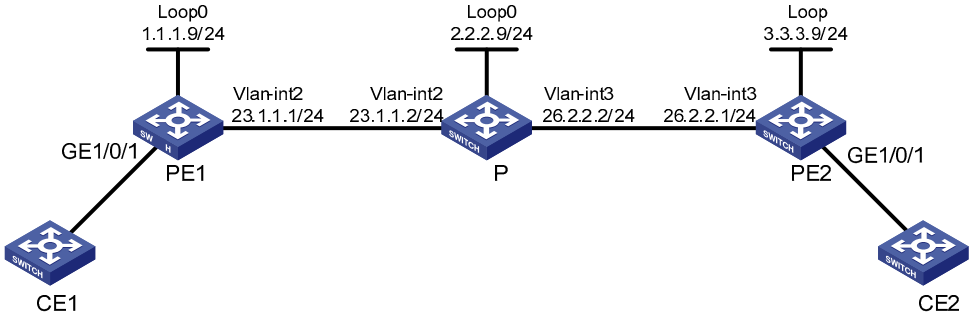
VPLS configuration examples

Binding service instances with VPLS instances

Network requirements

- CE 1 and CE 2 are connected to PE 1 and PE 2 respectively through VLANs.
- Configure VPLS instance **aaa** to use LDP (Martini mode) and VPLS instance **bbb** to use BGP (Kompella mode), and configure the AS number as 100.
- On PE 1 and PE 2: Configure service instance 1000 to match packets that are received on GigabitEthernet 1/0/1 and carry the VLAN tag of 100. Bind service instance 1000 to VPLS instance **aaa**. Configure service instance 2000 to match packets that are received on GigabitEthernet 1/0/1 and carry VLAN tag of 200. Bind service instance 2000 to VPLS instance **bbb**.

Figure 43 Network diagram for binding service instances with VPLS instances



Procedure

1. Configure PE 1

```
<Sysname> system-view
[Sysname] sysname PE1
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 1.1.1.9 32
[PE1-LoopBack0] quit
```

Configure the LSR ID and enable MPLS globally.

```
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls
[PE1-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

Enable LDP globally.

```
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

Configure PE 1 to establish an LDP remote session with PE 2.

```
[PE1] mpls ldp remote-peer 1
[PE1-mpls-ldp-remote-1] remote-ip 3.3.3.9
```

```
[PE1-mpls-ldp-remote-1] quit
```

Configure the interface connected with the P device and enable LDP on the interface.

```
[PE1] interface vlan-interface 2
[PE1-Vlan-interface2] ip address 23.1.1.1 24
[PE1-Vlan-interface2] mpls
[PE1-Vlan-interface2] mpls ldp
[PE1-Vlan-interface2] quit
```

Configure OSPF.

```
[PE1] ospf
[PE1-ospf-1] area 0
[PE1-ospf-1-area-0.0.0.0] network 23.1.1.1 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 BGP extensions.

```
[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] vpls-family
[PE1-bgp-af-vpls] peer 3.3.3.9 enable
[PE1-bgp-af-vpls] quit
[PE1-bgp] quit
```

Configure the basic attributes of VPLS instance **aaa**, which uses LDP.

```
[PE1] vsi aaa static
[PE1-vsi-aaa] pwsignal ldp
[PE1-vsi-aaa-ldp] vsi-id 500
[PE1-vsi-aaa-ldp] peer 3.3.3.9
[PE1-vsi-aaa-ldp] quit
[PE1-vsi-aaa] quit
```

Configure the basic attributes of VPLS instance **bbb**, which uses BGP.

```
[PE1] vsi bbb auto
[PE1-vsi-bbb] pwsignal bgp
[PE1-vsi-bbb-bgp] route-distinguisher 100:1
[PE1-vsi-bbb-bgp] vpn-target 111:1
[PE1-vsi-bbb-bgp] site 100
[PE1-vsi-bbb-bgp] quit
[PE1-vsi-bbb] quit
```

On the interface connecting CE 1, create service instance 1000 and bind it with VPLS instance **aaa**, and create service instance 2000 and bind it with VPLS instance **bbb**.

```
[PE1] interface gigabitethernet 1/0/1
[PE1-GigabitEthernet1/0/1] service-instance 1000
[PE1-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 100
[PE1-GigabitEthernet1/0/1-srv1000] xconnect vsi aaa
[PE1-GigabitEthernet1/0/1-srv1000] quit
[PE1-GigabitEthernet1/0/1] service-instance 2000
[PE1-GigabitEthernet1/0/1-srv2000] encapsulation s-vid 200
```

```
[PE1-GigabitEthernet1/0/1-srv2000] xconnect vsi bbb
[PE1-GigabitEthernet1/0/1-srv2000] quit
[PE1-GigabitEthernet1/0/1] quit
```

2. Configure the P device

```
<Sysname> system-view
[Sysname] sysname P
[P] interface loopback 0
[P-LoopBack0] ip address 2.2.2.9 32
[P-LoopBack0] quit
```

Configure the LSR ID and enable MPLS globally.

```
[P] mpls lsr-id 2.2.2.9
[P] mpls
[P-mpls] quit
```

Enable LDP globally.

```
[P] mpls ldp
[P-mpls-ldp] quit
```

Configure the interface connected with PE 1 and enable LDP on the interface.

```
[P] interface vlan-interface 2
[P-Vlan-interface2] ip address 23.1.1.2 24
[P-Vlan-interface2] mpls
[P-Vlan-interface2] mpls ldp
[P-Vlan-interface2] quit
```

Configure the interface connected with PE 2 and enable LDP on the interface.

```
[P] interface vlan-interface 3
[P-Vlan-interface3] ip address 26.2.2.2 24
[P-Vlan-interface3] mpls
[P-Vlan-interface3] mpls ldp
[P-Vlan-interface3] quit
```

Configure OSPF.

```
[P] ospf
[P-ospf-1] area 0
[P-ospf-1-area-0.0.0.0] network 23.1.1.2 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 26.2.2.2 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
```

3. Configure PE 2

```
<Sysname> system-view
[Sysname] sysname PE2
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 3.3.3.9 32
[PE2-LoopBack0] quit
```

Configure the LSR-ID and enable MPLS globally.

```
[PE2] mpls lsr-id 3.3.3.9
[PE2] mpls
[PE2-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```

Enable LDP globally.

```
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

Configure PE 2 to establish a remote LDP session with PE 1.

```
[PE2] mpls ldp remote-peer 2
[PE2-mpls-ldp-remote-2] remote-ip 1.1.1.9
[PE2-mpls-ldp-remote-2] quit
```

Configure the interface connected with the P device and enable LDP on the interface.

```
[PE2] interface vlan-interface 3
[PE2-Vlan-interface3] ip address 26.2.2.1 24
[PE2-Vlan-interface3] mpls
[PE2-Vlan-interface3] mpls ldp
[PE2-Vlan-interface3] quit
```

Configure OSPF.

```
[PE2] ospf
[PE2-ospf-1] area 0
[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] network 26.2.2.0 0.0.0.255
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit
```

Configure BGP extensions.

```
[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] vpls-family
[PE2-bgp-af-vpls] peer 1.1.1.9 enable
[PE2-bgp-af-vpls] quit
[PE2-bgp] quit
```

Configure the basic attributes of VPLS instance **aaa**, which uses LDP.

```
[PE2] vsi aaa static
[PE2-vsi-aaa] pwsignal ldp
[PE2-vsi-aaa-ldp] vsi-id 500
[PE2-vsi-aaa-ldp] peer 1.1.1.9
[PE2-vsi-aaa-ldp] quit
[PE2-vsi-aaa] quit
```

Configure the basic attributes of VPLS instance **bbb**, which uses BGP.

```
[PE2] vsi bbb auto
[PE2-vsi-bbb] pwsignal bgp
[PE2-vsi-bbb-bgp] route-distinguisher 100:1
[PE2-vsi-bbb-bgp] vpn-target 111:1
[PE2-vsi-bbb-bgp] site 200
```

```
[PE2-vsi-bbb-bgp] quit
[PE2-vsi-bbb] quit
```

On the interface connecting CE 2, create service instance 1000 and bind it with VPLS instance **aaa**, and create service instance 2000 and bind it with VPLS instance **bbb**.

```
[PE2] interface gigabitethernet 1/0/1
[PE2-GigabitEthernet1/0/1] service-instance 1000
[PE2-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 100
[PE2-GigabitEthernet1/0/1-srv1000] xconnect vsi aaa
[PE2-GigabitEthernet1/0/1-srv1000] quit
[PE2-GigabitEthernet1/0/1] service-instance 2000
[PE2-GigabitEthernet1/0/1-srv2000] encapsulation s-vid 200
[PE2-GigabitEthernet1/0/1-srv2000] xconnect vsi bbb
[PE2-GigabitEthernet1/0/1-srv2000] quit
[PE2-GigabitEthernet1/0/1] quit
```

After completing the configurations, issue **display vpls connection** on the PEs. The output shows that PW connections have been established and in the up state. Take PE 2 as an example:

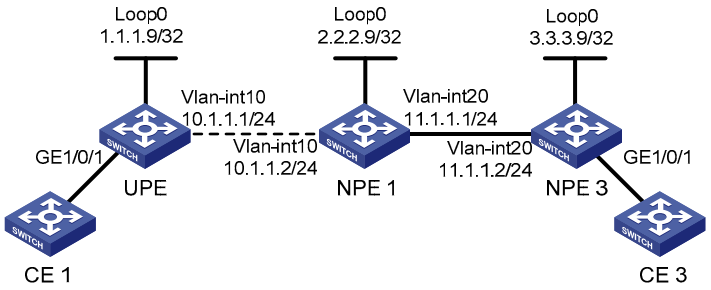
```
[PE2] display vpls connection vsi aaa verbose
VSI Name: aaa                              Signaling: ldp
  **Remote Vsi ID   : 500
    VC State        : up
    Encapsulation   : vlan
    Group ID        : 0
    MTU             : 1500
    Peer Ip Address : 1.1.1.9
    PW Type         : label
    Local VC Label  : 89766
    Remote VC Label : 81922
    Link ID         : 1
    Tunnel Policy   : --
    Tunnel ID       : 0x4600068
```

Configuring H-VPLS by using LSP

Network requirements

- A PW connection U-PW is required between UPE and NPE 1. CE 1 accesses the network through UPE.
- A PW connection N-PW is required between NPE 1 and NPE 3. CE 3 accesses the network through NPE 3.
- CE 1 is connected to port GigabitEthernet 1/0/1 of UPE. CE 3 is connected to port GigabitEthernet 1/0/1 of NPE 3. CE 1 and CE 3 communicate with UPE and NPE 3 respectively through VLAN 100.
- UPE and NPE 1 are connected through their VLAN-interface 10 interfaces.
- NPE 1 and NPE 3 are connected through their VLAN-interface 20 interfaces.
- VPLS instance aaa uses LDP, or, the Martini mode.

Figure 44 Network diagram for configuring H-VPLS by using LSP



Procedure

1. Configure the IGP protocol on the MPLS backbone, which is OSPF in this example. The detailed configuration steps are omitted.
2. Configure UPE

Configure MPLS basic capability.

```
<Sysname> system-view
[Sysname] sysname UPE
[UPE] interface loopback 0
[UPE-LoopBack0] ip address 1.1.1.9 32
[UPE-LoopBack0] quit
[UPE] mpls lsr-id 1.1.1.9
[UPE] mpls
[UPE-mpls] quit
[UPE] mpls ldp
[UPE-mpls-ldp] quit
```

Configure MPLS basic capability on the interface connected with NPE 1.

```
[UPE] interface vlan-interface 10
[UPE-Vlan-interface10] ip address 10.1.1.1 24
[UPE-Vlan-interface10] mpls
[UPE-Vlan-interface10] mpls ldp
[UPE-Vlan-interface10] quit
```

Configure the remote LDP session.

```
[UPE] mpls ldp remote-peer 1
[UPE-mpls-remote-1] remote-ip 2.2.2.9
[UPE-mpls-remote-1] quit
```

Enable L2VPN and MPLS L2VPN.

```
[UPE] l2vpn
[UPE-l2vpn] mpls l2vpn
[UPE-l2vpn] quit
```

Configure the basic attributes of VPLS instance aaa, which uses LDP.

```
[UPE] vsi aaa static
[UPE-vsi-aaa] pwsignal ldp
[UPE-vsi-aaa-ldp] vsi-id 500
[UPE-vsi-aaa-ldp] peer 2.2.2.9
[UPE-vsi-aaa-ldp] quit
```

```
[UPE-vsi-aaa] quit
```

Create a service instance on port GigabitEthernet 1/0/1, and then bind VLAN 100 and VPLS instance aaa.

```
[UPE] interface gigabitethernet 1/0/1
[UPE-GigabitEthernet1/0/1] service-instance 1000
[UPE-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 100
[UPE-GigabitEthernet1/0/1-srv1000] xconnect vsi aaa
[UPE-GigabitEthernet1/0/1-srv1000] quit
```

3. Configure NPE 1

Configure MPLS basic capability.

```
<Sysname> system-view
[Sysname] sysname NPE1
[NPE1] interface loopback 0
[NPE1-LoopBack0] ip address 2.2.2.9 32
[NPE1-LoopBack0] quit
[NPE1] mpls lsr-id 2.2.2.9
[NPE1] mpls
[NPE1-mpls] quit
[NPE1] mpls ldp
[NPE1-mpls-ldp] quit
```

Configure MPLS basic capability on the interface connected with UPE.

```
[NPE1] interface vlan-interface 10
[NPE1-Vlan-interface10] ip address 10.1.1.2 24
[NPE1-Vlan-interface10] mpls
[NPE1-Vlan-interface10] mpls ldp
[NPE1-Vlan-interface10] quit
```

Configure MPLS basic capability on the interface connected with NPE 3.

```
[NPE1] interface vlan-interface 20
[NPE1-Vlan-interface20] ip address 11.1.1.1 24
[NPE1-Vlan-interface20] mpls
[NPE1-Vlan-interface20] mpls ldp
[NPE1-Vlan-interface20] quit
```

Configure the remote LDP session with UPE.

```
[NPE1] mpls ldp remote-peer 2
[NPE1-mpls-remote-2] remote-ip 1.1.1.9
[NPE1-mpls-remote-2] quit
```

Configure the remote LDP session with NPE 3.

```
[NPE1] mpls ldp remote-peer 3
[NPE1-mpls-remote-3] remote-ip 3.3.3.9
[NPE1-mpls-remote-3] quit
```

Enable L2VPN and MPLS L2VPN.

```
[NPE1] l2vpn
[NPE1-l2vpn] mpls l2vpn
[NPE1-l2vpn] quit
```

Configure the basic attributes of VPLS instance aaa, which uses LDP.

```
[NPE1] vsi aaa static
[NPE1-vsi-aaa] pwsignal ldp
[NPE1-vsi-aaa-ldp] vsi-id 500
[NPE1-vsi-aaa-ldp] peer 1.1.1.9 upe
[NPE1-vsi-aaa-ldp] peer 3.3.3.9
[NPE1-vsi-aaa-ldp] quit
[NPE1-vsi-aaa] quit
```

4. Configure NPE 3

Configure MPLS basic capability.

```
<Sysname> system-view
[Sysname] sysname NPE3
[NPE3] interface loopback 0
[NPE3-LoopBack0] ip address 3.3.3.9 32
[NPE3-LoopBack0] quit
[NPE3] mpls lsr-id 3.3.3.9
[NPE3] mpls
[NPE3-mpls] quit
[NPE3] mpls ldp
[NPE3-mpls-ldp] quit
```

Configure MPLS basic capability on the interface connected with NPE 1.

```
[NPE3] interface vlan-interface 20
[NPE3-Vlan-interface20] ip address 11.1.1.2 24
[NPE3-Vlan-interface20] mpls
[NPE3-Vlan-interface20] mpls ldp
[NPE3-Vlan-interface20] quit
```

Configure the remote LDP session.

```
[NPE3] mpls ldp remote-peer 1
[NPE3-mpls-remote-1] remote-ip 2.2.2.9
[NPE3-mpls-remote-1] quit
```

Enable L2VPN and MPLS L2VPN.

```
[NPE3] l2vpn
[NPE3-l2vpn] mpls l2vpn
[NPE3-l2vpn] quit
```

Configure the basic attributes of VPLS instance aaa, which uses LDP.

```
[NPE3] vsi aaa static
[NPE3-vsi-aaa] pwsignal ldp
[NPE3-vsi-aaa-ldp] vsi-id 500
[NPE3-vsi-aaa-ldp] peer 2.2.2.9
[NPE3-vsi-aaa-ldp] quit
[NPE3-vsi-aaa] quit
```

Create a service instance on port GigabitEthernet 1/0/1, and then bind VLAN 100 and VPLS instance aaa.

```
[NPE3] interface gigabitethernet 1/0/1
[NPE3-GigabitEthernet1/0/1] service-instance 1000
[NPE3-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 100
[NPE3-GigabitEthernet1/0/1-srv1000] xconnect vsi aaa
```

```
[NPE3-GigabitEthernet1/0/1-srv1000] quit
```

After completing the configurations, issue **display vpls connection** on the PEs. You can see that a PW connection has been established and in the up state.

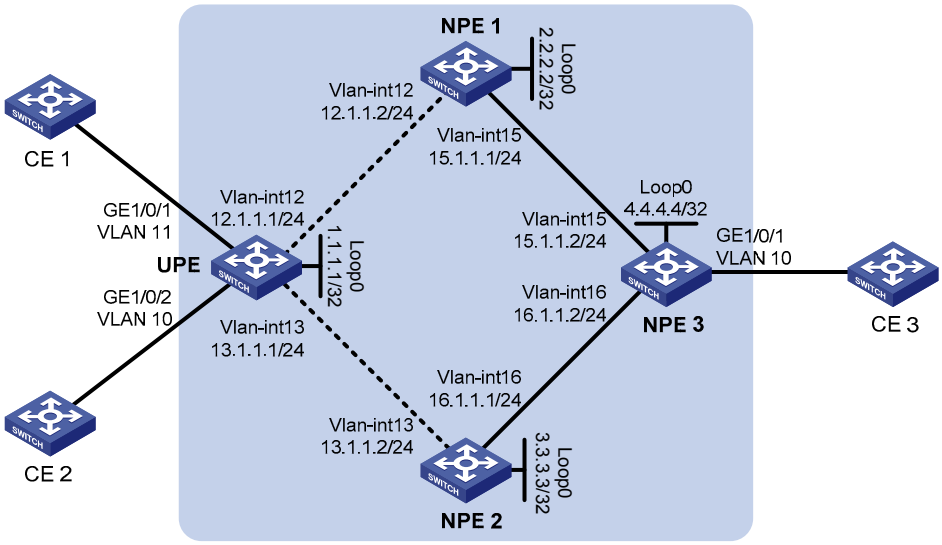
Configuring a backup link for H-VPLS access

Network requirements

As shown in Figure 45:

- CE 1 and CE 2 are connected to UPE through VLANs.
- UPE establishes a PW connection (U-PW) with NPE 1 and NPE 2 respectively, with the NPE 2 link as the backup.
- NPE 1 and NPE 2 each establish a PW connection (N-PW) with NPE 3. CE 3 is connected to the network through NPE 3.
- UPE is connected to NPE 1 through VLAN-interface 13, and is connected to NPE 2 through VLAN-interface 12.
- NPE 1 is connected to NPE 3 through VLAN-interface 15, and NPE 2 is connected to NPE 3 through VLAN-interface 16.
- Configure the VPLS instance to support H-VPLS networking.

Figure 45 Network diagram for configuring H-VPLS using LSP



Procedure

1. Configure the IGP protocol on the MPLS backbone, which is OSPF in this example. The detailed configuration steps are omitted.
2. Configure UPE

Configure MPLS basic capability.

```
<Sysname> system-view
[Sysname] sysname UPE
[UPE] interface loopback 0
[UPE-LoopBack0] ip address 1.1.1.1 32
[UPE-LoopBack0] quit
```

```
[UPE] mpls lsr-id 1.1.1.1
[UPE] mpls
[UPE-mpls] quit
[UPE] mpls ldp
[UPE-mpls-ldp] quit
```

Configure MPLS basic capability on the interface connected with NPE 1.

```
[UPE] interface vlan-interface 12
[UPE-Vlan-interface12] ip address 12.1.1.1 24
[UPE-Vlan-interface12] mpls
[UPE-Vlan-interface12] mpls ldp
[UPE-Vlan-interface12] quit
```

Configure the remote LDP session with NPE 1.

```
[UPE] mpls ldp remote-peer 1
[UPE-mpls-remote-1] remote-ip 2.2.2.2
[UPE-mpls-remote-1] quit
```

Configure the remote LDP session with NPE 2.

```
[UPE] mpls ldp remote-peer 2
[UPE-mpls-remote-1] remote-ip 3.3.3.3
[UPE-mpls-remote-1] quit
```

Enable L2VPN and MPLS L2VPN.

```
[UPE] l2vpn
[UPE-l2vpn] mpls l2vpn
[UPE-l2vpn] quit
```

Configure the basic attributes of VPLS instance aaa, which uses LDP.

```
[UPE] vsi aaa static
[UPE-vsi-aaa] pwsignal ldp
[UPE-vsi-aaa-ldp] vsi-id 500
[UPE-vsi-aaa-ldp] peer 2.2.2.2 backup-peer 3.3.3.3
[UPE-vsi-aaa-ldp] dual-npe revertive wtr-time 1
[UPE-vsi-aaa-ldp] quit
[UPE-vsi-aaa] quit
```

Configure interface VLAN-interface 12 and enable MPLS on the interface.

```
[UPE] interface vlan-interface 12
[UPE-Vlan-interface12] ip address 12.1.1.1 255.255.255.0
[UPE-Vlan-interface12] mpls
[UPE-Vlan-interface12] mpls ldp
[UPE-Vlan-interface12] quit
```

Configure interface VLAN-interface 13 and enable MPLS on the interface.

```
[UPE] interface vlan-interface 13
[UPE-Vlan-interface13] ip address 13.1.1.1 255.255.255.0
[UPE-Vlan-interface13] mpls
[UPE-Vlan-interface13] mpls ldp
[UPE-Vlan-interface13] quit
```

On the interface connected with CE 1, create a service instance and bind the VSI.

```
[UPE] interface gigabitethernet 1/0/1
```

```
[UPE-GigabitEthernet1/0/1] service-instance 1000
[UPE-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 10
[UPE-GigabitEthernet1/0/1-srv1000] xconnect vsi aaa
[UPE-GigabitEthernet1/0/1-srv1000] quit
```

On the interface connected with CE 2, create a service instance and bind the VSI.

```
[UPE] interface gigabitethernet 1/0/2
[UPE-GigabitEthernet1/0/2] service-instance 1000
[UPE-GigabitEthernet1/0/2-srv1000] encapsulation s-vid 11
[UPE-GigabitEthernet1/0/2-srv1000] xconnect vsi aaa
[UPE-GigabitEthernet1/0/2-srv1000] quit
```

3. Configure NPE 1

Configure MPLS basic capability.

```
<Sysname> system-view
[Sysname] sysname NPE1
[NPE1] interface loopback 0
[NPE1-LoopBack0] ip address 2.2.2.2 32
[NPE1-LoopBack0] quit
[NPE1] mpls lsr-id 2.2.2.2
[NPE1] mpls
[NPE1-mpls] quit
[NPE1] mpls ldp
[NPE1-mpls-ldp] quit
```

Configure MPLS basic capability on the interface connected with UPE.

```
[NPE1] interface vlan-interface 13
[NPE1-Vlan-interface13] ip address 13.1.1.2 24
[NPE1-Vlan-interface13] mpls
[NPE1-Vlan-interface13] mpls ldp
[NPE1-Vlan-interface13] quit
```

Configure MPLS basic capability on the interface connected with NPE 3.

```
[NPE1] interface vlan-interface 15
[NPE1-Vlan-interface15] ip address 15.1.1.1 24
[NPE1-Vlan-interface15] mpls
[NPE1-Vlan-interface15] mpls ldp
[NPE1-Vlan-interface15] quit
```

Configure the remote LDP session with UPE.

```
[NPE1] mpls ldp remote-peer 2
[NPE1-mpls-remote-2] remote-ip 1.1.1.1
[NPE1-mpls-remote-2] quit
```

Configure the remote LDP session with NPE 3.

```
[NPE1] mpls ldp remote-peer 3
[NPE1-mpls-remote-3] remote-ip 4.4.4.4
[NPE1-mpls-remote-3] quit
```

Enable L2VPN and MPLS L2VPN.

```
[NPE1] l2vpn
[NPE1-l2vpn] mpls l2vpn
```

```
[NPE1-l2vpn] quit
```

Configure the basic attributes of VPLS instance aaa, which uses LDP.

```
[NPE1] vsi aaa static
[NPE1-vsi-aaa] pwsignal ldp
[NPE1-vsi-aaa-ldp] vsi-id 500
[NPE1-vsi-aaa-ldp] peer 1.1.1.1 upe
[NPE1-vsi-aaa-ldp] peer 4.4.4.4
[NPE1-vsi-aaa-ldp] quit
[NPE1-vsi-aaa] quit
```

The configuration procedure on NPE 2 is similar to that on NPE 1, and therefore is omitted.

4. Configure NPE 3

Configure MPLS basic capability.

```
<Sysname> system-view
[Sysname] sysname NPE3
[NPE3] interface loopback 0
[NPE3-LoopBack0] ip address 4.4.4.4 32
[NPE3-LoopBack0] quit
[NPE3] mpls lsr-id 4.4.4.4
[NPE3] mpls
[NPE3-mpls] quit
[NPE3] mpls ldp
[NPE3-mpls-ldp] quit
```

Configure MPLS basic capability on the interface connected with NPE 1.

```
[NPE3] interface vlan-interface 15
[NPE3-Vlan-interface15] ip address 15.1.1.2 24
[NPE3-Vlan-interface15] mpls
[NPE3-Vlan-interface15] mpls ldp
[NPE3-Vlan-interface15] quit
```

Configure the remote LDP session.

```
[NPE3] mpls ldp remote-peer 1
[NPE3-mpls-remote-1] remote-ip 2.2.2.2
[NPE3-mpls-remote-1] quit
[NPE3] mpls ldp remote-peer 2
[NPE3-mpls-remote-2] remote-ip 3.3.3.3
[NPE3-mpls-remote-2] quit
```

Enable L2VPN and MPLS L2VPN.

```
[NPE3] l2vpn
[NPE3-l2vpn] mpls l2vpn
[NPE3-l2vpn] quit
```

Configure the basic attributes of VPLS instance aaa, which uses LDP.

```
[NPE3] vsi aaa static
[NPE3-vsi-aaa] pwsignal ldp
[NPE3-vsi-aaa-ldp] vsi-id 500
[NPE3-vsi-aaa-ldp] peer 2.2.2.2
[NPE3-vsi-aaa-ldp] peer 3.3.3.3
```

```
[NPE3-vsi-aaa-ldp] quit
[NPE3-vsi-aaa] quit
```

Configure interface VLAN-interface 15 and enable MPLS.

```
[NPE3] interface vlan-interface 15
[NPE3-Vlan-interface15] ip address 15.1.1.1 255.255.255.0
[NPE3-Vlan-interface15] mpls
[NPE3-Vlan-interface15] mpls ldp
[NPE3-Vlan-interface15] quit
```

Configure interface VLAN-interface 16 and enable MPLS.

```
[NPE3] interface vlan-interface 16
[NPE3-Vlan-interface16] ip address 16.1.1.1 255.255.255.0
[NPE3-Vlan-interface16] mpls
[NPE3-Vlan-interface16] mpls ldp
[NPE3-Vlan-interface16] quit
```

Create service instance on GigabitEthernet 1/0/1, the interface connecting CE 3, and bind the VSI.

```
[NPE3] interface gigabitethernet 1/0/1
[NPE3-GigabitEthernet1/0/1] service-instance 1000
[NPE3-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 10
[NPE3-GigabitEthernet1/0/1-srv1000] xconnect vsi aaa
[NPE3-GigabitEthernet1/0/1-srv1000] quit
```

The configuration of VLAN-interface 15 and VLAN-interface 16 is similar to the configuration of VLAN-interface 12 and VLAN-interface 13 on UPE. The configuration procedure is omitted.

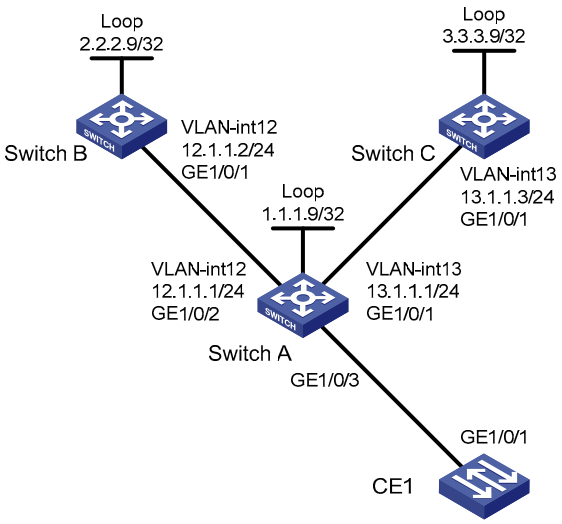
After completing the configurations, execute **display vpls connection** on the PEs. You can see that a PW connection has been established and in the up state.

Configuring BFD in an H-VPLS network to detect errors of the main link

Network requirements

- In the H-VPLS network, Switch A is the UPE, Switch B is the main NPE and Switch C is the backup NPE. Enable MPLS on the interfaces connecting the switches, and enable OSPF on the switches to ensure the reachability at the network layer.
- CE 1 is connected to port GigabitEthernet 1/0/3 of Switch A, and communicates with Switch A through VLAN 100.
- It is required that when the link between Switch A and Switch B is down, the link failure be detected and informed to the MPLS LDP protocol for fast PW switchover.

Figure 46 Network diagram for configuring BFD in an H-VPLS network for main link detection



Procedure

1. Configure MPLS basic capabilities

Configure Switch A.

```
<SwitchA> system-view
[SwitchA] mpls lsr-id 1.1.1.9
[SwitchA] mpls
[SwitchA-mpls] quit
[SwitchA] mpls ldp
[SwitchA-mpls-ldp] quit
[SwitchA] mpls ldp remote-peer switchb
[SwitchA-mpls-ldp-remote-switchb] remote-ip 2.2.2.9
[SwitchA-mpls-ldp-remote-switchb] remote-ip bfd
[SwitchA-mpls-ldp-remote-switchb] quit
[SwitchA] mpls ldp remote-peer switchc
[SwitchA-mpls-ldp-remote-switchc] remote-ip 3.3.3.9
[SwitchA-mpls-ldp-remote-switchc] remote-ip bfd
[SwitchA-mpls-ldp-remote-switchc] quit
[SwitchA] vlan 12
[SwitchA-vlan12] port gigabitethernet 1/0/2
[SwitchA-vlan12] quit
[SwitchA] vlan 13
[SwitchA-vlan13] port gigabitethernet 1/0/1
[SwitchA-vlan13] quit
[SwitchA] interface vlan-interface 12
[SwitchA-Vlan-interface12] mpls
[SwitchA-Vlan-interface12] mpls ldp
[SwitchA-Vlan-interface12] quit
[SwitchA] interface vlan-interface 13
[SwitchA-Vlan-interface13] mpls
[SwitchA-Vlan-interface13] mpls ldp
```

```
[SwitchA-Vlan-interface13] quit
```

Configure Switch B.

```
<SwitchB> system-view
[SwitchB] mpls lsr-id 2.2.2.9
[SwitchB] mpls
[SwitchB-mpls] quit
[SwitchB] mpls ldp
[SwitchB-mpls-ldp] quit
[SwitchB] mpls ldp remote-peer switcha
[SwitchB-mpls-ldp-remote-switcha] remote-ip 1.1.1.9
[SwitchB-mpls-ldp-remote-switcha] remote-ip bfd
[SwitchB-mpls-ldp-remote-switcha] quit
[SwitchB] vlan 12
[SwitchB-vlan12] port gigabitethernet 1/0/1
[SwitchB-vlan12] quit
[SwitchB] interface vlan-interface 12
[SwitchB-Vlan-interface12] mpls
[SwitchB-Vlan-interface12] mpls ldp
[SwitchB-Vlan-interface12] quit
```

Configure Switch C.

```
<SwitchC> system-view
[SwitchC] mpls lsr-id 3.3.3.9
[SwitchC] mpls
[SwitchC-mpls] quit
[SwitchC] mpls ldp
[SwitchC-mpls-ldp] quit
[SwitchC] mpls ldp remote-peer switcha
[SwitchC-mpls-ldp-remote-switcha] remote-ip 1.1.1.9
[SwitchC-mpls-ldp-remote-switcha] remote-ip bfd
[SwitchC-mpls-ldp-remote-switcha] quit
[SwitchC] vlan 13
[SwitchC-vlan13] port gigabitethernet 1/0/1
[SwitchC-vlan13] quit
[SwitchC] interface vlan-interface 13
[SwitchC-Vlan-interface13] mpls
[SwitchC-Vlan-interface13] mpls ldp
[SwitchC-Vlan-interface13] quit
```

2. Configure related interfaces on the switches

Configure Switch A.

```
[SwitchA] interface vlan-interface 12
[SwitchA-Vlan-interface12] ip address 12.1.1.1 24
[SwitchA-Vlan-interface12] quit
[SwitchA] interface vlan-interface 13
[SwitchA-Vlan-interface13] ip address 13.1.1.1 24
[SwitchA-Vlan-interface13] quit
[SwitchA] interface loopback 0
[SwitchA-LoopBack0] ip address 1.1.1.9 32
```

```
[SwitchA-LoopBack0] quit
```

Configure Switch B.

```
[SwitchB] interface vlan-interface 12
[SwitchB-Vlan-interface12] ip address 12.1.1.2 24
[SwitchB-Vlan-interface12] quit
[SwitchB] interface loopback 0
[SwitchB-LoopBack0] ip address 2.2.2.9 32
[SwitchB-LoopBack0] quit
```

Configure Switch C.

```
[SwitchC] interface vlan-interface 13
[SwitchC-Vlan-interface13] ip address 13.1.1.3 24
[SwitchC-Vlan-interface13] quit
[SwitchC] interface loopback 0
[SwitchC-LoopBack0] ip address 3.3.3.9 32
[SwitchC-LoopBack0] quit
```

3. Configure basic OSPF functions

Configure Switch A.

```
[SwitchA] ospf
[SwitchA-ospf-1] area 0
[SwitchA-ospf-1-area-0.0.0.0] network 12.1.1.1 0.0.0.255
[SwitchA-ospf-1-area-0.0.0.0] network 13.1.1.1 0.0.0.255
[SwitchA-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[SwitchA-ospf-1-area-0.0.0.0] quit
[SwitchA-ospf-1] quit
```

Configure Switch B.

```
[SwitchB] ospf
[SwitchB-ospf-1] area 0
[SwitchB-ospf-1-area-0.0.0.0] network 12.1.1.2 0.0.0.255
[SwitchB-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[SwitchB-ospf-1-area-0.0.0.0] quit
[SwitchB-ospf-1] quit
```

Configure Switch C.

```
[SwitchC] ospf
[SwitchC-ospf-1] area 0
[SwitchC-ospf-1-area-0.0.0.0] network 13.1.1.3 0.0.0.255
[SwitchC-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[SwitchC-ospf-1-area-0.0.0.0] quit
[SwitchC-ospf-1] quit
```

4. Configure a VSI for each switch.

Configure Switch A.

```
[SwitchA] l2vpn
[SwitchA-l2vpn] mpls l2vpn
[SwitchA-l2vpn] quit
[SwitchA] vsi vpna static
[SwitchA-vsi-vpna] pwsignal ldp
```

```
[SwitchA-vsi-vpna-ldp] vsi-id 100
[SwitchA-vsi-vpna-ldp] peer 2.2.2.9 backup-peer 3.3.3.9
[SwitchA-vsi-vpna-ldp] quit
[SwitchA-vsi-vpna] quit
[SwitchA] vlan 100
[SwitchA-vlan100] port gigabitethernet 1/0/3
[SwitchA-vlan100] quit
[SwitchA] interface gigabitethernet 1/0/3
[SwitchA-GigabitEthernet1/0/3] service-instance 1000
[SwitchA-GigabitEthernet1/0/3-srv1000] encapsulation s-vid 100
[SwitchA-GigabitEthernet1/0/3-srv1000] xconnect vsi vpna
[SwitchA-GigabitEthernet1/0/3-srv1000] quit
```

Configure Switch B.

```
[SwitchB] l2vpn
[SwitchB-l2vpn] mpls l2vpn
[SwitchB-l2vpn] quit
[SwitchB] vsi vpna static
[SwitchB-vsi-vpna] pwsignal ldp
[SwitchB-vsi-vpna-ldp] vsi-id 100
[SwitchB-vsi-vpna-ldp] peer 1.1.1.9 upe
[SwitchB-vsi-vpna-ldp] quit
[SwitchB-vsi-vpna] quit
```

Configure Switch C.

```
[SwitchC] l2vpn
[SwitchC-l2vpn] mpls l2vpn
[SwitchC-l2vpn] quit
[SwitchC] vsi vpna static
[SwitchC-vsi-vpna] pwsignal ldp
[SwitchC-vsi-vpna-ldp] vsi-id 100
[SwitchC-vsi-vpna-ldp] peer 1.1.1.9 upe
[SwitchC-vsi-vpna-ldp] quit
[SwitchC-vsi-vpna] quit
```

5. Verification

Use **display bfd session verbose** to display information about the BFD sessions from Switch A to its neighbors.

```
<SwitchA> display bfd session verbose

 Total Session Num: 2            Init Mode: Active

 Session Working Under Ctrl Mode:

     Local Discr: 21                 Remote Discr: 20
       Source IP: 1.1.1.9          Destination IP: 2.2.2.9
   Session State: Up                      Interface: LoopBack0
 Min Trans Inter: 400ms            Act Trans Inter: 400ms
  Min Recv Inter: 400ms           Act Detect Inter: 2000ms
  Running Up for: 00:00:01               Auth mode: None
```

```
    Connect Type: Indirect                Board Num: 6
        Protocol: MFW/LDP
       Diag Info: No Diagnostic

     Local Discr: 4                   Remote Discr: 0
       Source IP: 1.1.1.9           Destination IP: 3.3.3.9
   Session State: Up                      Interface: LoopBack0
 Min Trans Inter: 400ms             Act Trans Inter: 1000ms
  Min Recv Inter: 400ms           Act Detect Inter: 3000ms
  Running Up for: 00:00:01               Auth mode: None
    Connect Type: Indirect               Board Num: 6
        Protocol: MFW/LDP
       Diag Info: No Diagnostic
```

Execute **display vpls connection vsi vpna** on Switch A. You can see that the link between Switch A and Switch B is up.

```
<SwitchA> display vpls connection vsi vpna

Total 2 connection(s),
connection(s): 1 up, 1 block, 0 down

VSI Name: vpna                              Signaling: ldp
VsiID      VsiType           PeerAddr      InLabel OutLabel LinkID  VCState
100        vlan              2.2.2.9       134312  138882   1       up
100        vlan              3.3.3.9       134216  140476   2       block
```

Use **display vpls fib vsi vpna verbose** to display the forwarding table information of the VPLS instance on Switch A.

```
[SwitchA] display vpls fib vsi vpna verbose
VSI Name: vpna                             VSI Index: 0
  **Link ID             : 1
    Role                : Primary
    State               : Active
    In Label            : 134312
    Out Label           : 138882
    TnlType             : LDP-LSP
    MTU                 : 1500
    Tunnel ID           : 0x1130214
    Next Hop            : 12.1.1.2
    Out IfIndex         : 61997067
  **Link ID             : 2
    Role                : Backup
    State               : Standy
    In Label            : 134216
    Out Label           : 140476
    TnlType             : LDP-LSP
    MTU                 : 1500
    Tunnel ID           : 0x1130215
    Next Hop            : 13.1.1.3
```

```
    Out IfIndex         : 61997068
```

Disconnect the link between Switch A and Switch B. Then, execute **display vpls connection vsi vpna**. You can see that the link to 2.2.2.9 is down.

```
<SwitchA> display vpls connection vsi vpna

Total 2 connection(s),
connection(s): 1 up, 0 block, 1 down

VSI Name: vpna                              Signaling: ldp
VsiID       VsiType             PeerAddr        InLabel OutLabel LinkID  VCState
100         vlan                2.2.2.9         134312  138882   1       down
100         vlan                3.3.3.9         134216  140476   2       up
<SwitchA> display vpls fib vsi vpna verbose
VSI Name: vpna                             VSI Index: 0
  **Link ID             : 1
    Role                : Primary
    State               : Standy
    In Label            : 134312
    Out Label           : 138882
    TnlType             : LDP-LSP
    MTU                 : 1500
    Tunnel ID           : 0x1130214
    Next Hop            : 12.1.1.2
    Out IfIndex         : 61997067
  **Link ID             : 2
    Role                : Backup
    State               : Active
    In Label            : 134216
    Out Label           : 140476
    TnlType             : LDP-LSP
    MTU                 : 1500
    Tunnel ID           : 0x1130215
    Next Hop            : 13.1.1.3
    Out IfIndex         : 61997068
```

Troubleshooting VPLS

Symptom:

The VPLS link PW is not up.

Analysis:

- The public network LSP tunnel is not established.
- The extended session is not working normally.
- A private VLAN virtual interface is not bound with the corresponding VPLS instance and is not up.

Solution:

- Check the routing tables of the PEs to see whether a route is available between the two PEs. Check whether each device can ping the loopback interface of the peer and whether the LDP session is normal.
- Check whether any extended session configuration command is missing at either side.
- Check whether the private network interfaces are up or whether the PW to the UPE is up.
- Check whether the PW ID and transport mode are the same on the two peers.

Configuring MPLS L2VPN

The term *router* in this document refers to both routers and Layer 3 switches.

The A5820X switch series do not support MPLS L2VPN.

MPLS L2VPN technologies can provide both point-to-point connections and point-to-multipoint connections. This chapter describes only the MPLS L2VPN technologies that provide point-to-point connections. For information about the MPLS L2VPN technologies that provide point-to-multipoint connections, see "VPLS configuration."

Traditional VPN

Traditional VPNs based on ATM or FR are quite popular. They share the network infrastructure of carriers. However, they have some inherent disadvantages:

- Dependence on dedicated media: To provide both ATM-based and FR-based VPN services, carriers must establish two separate infrastructures across the whole service scope, one ATM infrastructure and one FR infrastructure. Apparently, the cost is very high and the infrastructures are not utilized efficiently.
- Complicated deployment: To add a site to an existing VPN, you have to modify the configurations of all edge nodes connected with the VPN site.

MPLS L2VPN is developed as a solution to address these disadvantages.

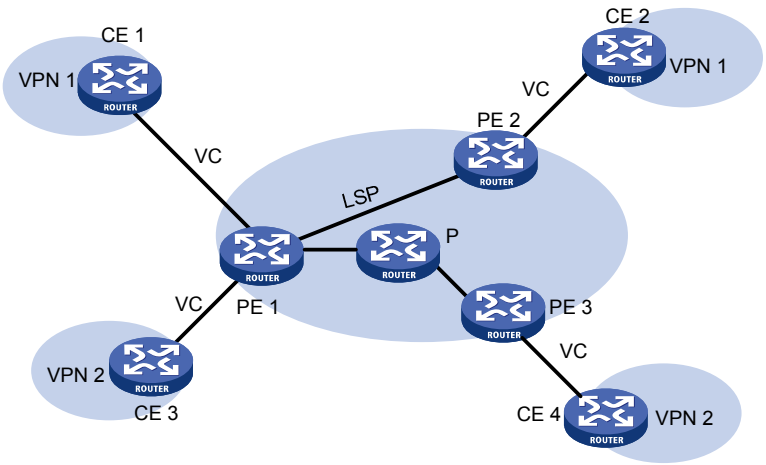
MPLS L2VPN

MPLS L2VPN provides Layer 2 VPN services on the MPLS network. It allows carriers to establish L2VPNs on different data link layer protocols, including ATM, FR, VLAN, Ethernet and PPP.

MPLS L2VPN transfers Layer 2 user data transparently on the MPLS network. For users, the MPLS network is a Layer 2 switched network and they can establish Layer 2 connections over the network.

Consider ATM as an example. Each CE device can connect to the MPLS network through an ATM VC to communicate with another CE. This is similar to that of an ATM network.

Figure 47 Network diagram for MPLS L2VPN



Comparison with MPLS L3VPN

Compared with MPLS L3VPN, MPLS L2VPN has the following advantages:

- High scalability: MPLS L2VPN establishes only Layer 2 connections. It does not involve the routing information of users. This greatly reduces the load of the PEs and even the load of the whole service provider network, enabling carriers to support more VPNs and to service more users.
- Guaranteed reliability and private routing information security: As no routing information of users is involved, MPLS L2VPN neither tries to obtain nor processes the routing information of users, guaranteeing the security of the user VPN routing information.
- Support for multiple network layer protocols, such as IP, IPX, and SNA.

Basic concepts of MPLS L2VPN

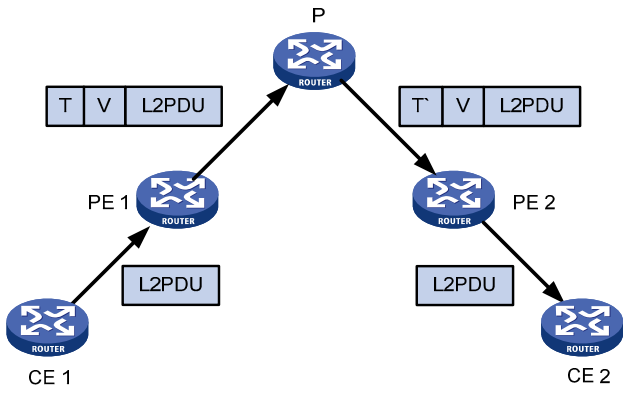
In MPLS L2VPN, the concepts and principles of CE, PE and P are the same as those in MPLS L3VPN:

- **CE device**—A CE resides on a customer network and has one or more interfaces directly connected with service provider networks. It can be a router, a switch, or a host. It cannot "sense" the existence of any VPN, neither does it need to support MPLS.
- **PE device**—A PE resides on a service provider network and connects one or more CEs to the network. On an MPLS network, all VPN processing occurs on the PEs.
- **P device**—A P device is a backbone router on a service provider network. It is not directly connected with any CE. It only needs to be equipped with basic MPLS forwarding capability.

MPLS L2VPN uses label stacks to implement the transparent transmission of user packets in the MPLS network.

- Outer label, also called "tunnel label", transfers packets from one PE to another.
- Inner label, also called "VC label", identifies different connections between VPNs.
- Upon receiving packets, a PE determines to which CE the packets are to be forwarded according to the VC labels.

Figure 48 MPLS L2VPN label stack processing



1) L2 PDU: Layer 2 protocol data unit

2) T represents tunnel label. V represents VC label. T′ represents swapped tunnel label.

Implementation of MPLS L2VPN

MPLS L2VPN can be implemented in one of the following methods:

- **CCC** and **SVC**—Two methods of implementing MPLS L2VPN by configuring VC labels statically.
- **Martini**—A method for establishing PPP links to implement MPLS L2VPN. It uses LDP as a signaling protocol to transfer VC labels.
- **Kompella**—A CE-to-CE mode for implementing MPLS L2VPN on the MPLS network. It uses extended BGP as the signaling protocol to advertise Layer 2 reachability information and VC labels.

The following sections describe these implementation methods for MPLS L2VPN in detail.

CCC MPLS L2VPN

Unlike common MPLS L2VPN, CCC employs just one level of label to transfer user data. Therefore, it uses LSPs exclusively. A CCC LSP can transfer only the data of the CCC connection. It can neither be used for other MPLS L2VPN connections, nor for MPLS L3VPN or common IP packets.

The most significant advantage of this method is that no label signaling is required for transferring Layer 2 VPN information. As long as MPLS forwarding is supported and service provider networks are interconnected, this method works perfectly. In addition, since LSPs are dedicated, this method supports QoS services.

CCC connections falls to two types, local connection and remote connection.

- A local connection is established between two local CEs that are connected to the same PE. The PE functions like a Layer 2 switch and can directly switch packets between the CEs without any static LSP.
- A remote connection is established between a local CE and a remote CE, which are connected to different PEs.

Only remote connection is supported by A5800 Switch Series.

You must configure for each remote CCC connection two LSPs, one for inbound and the other for outbound, on the P device along the remote connection.

SVC MPLS L2VPN

SVC also implements MPLS L2VPN by static configuration. It transfers L2VPN information without using any signaling protocol.

The SVC method resembles the Martini method closely and is in fact a static implementation of the Martini method. The difference is that it does not use LDP to transfer Layer 2 VC and link information. You only need to configure VC label information.

The labels for CCC and SVC range from 16 to 1023, which are reserved for static LSPs.

Martini MPLS L2VPN

The key of the Martini method is to set up VCs between CEs.

Martini MPLS L2VPN employs VC type and VC ID to identify a VC. The VC type indicates the encapsulation type of the VC, which can be ATM, VLAN, or PPP. The VC ID uniquely identifies the VC among the VCs of the same VC type on a PE.

The PEs connecting the two CEs of a VC exchange VC labels through LDP, and bind their respective CE by the VC ID.

Once LDP establishes an LSP between the two PEs and the label exchange and the binding to CE are finished, a VC is set up and ready to transfer Layer 2 data.

To allow the exchange of VC labels between PEs, the Martini method extended LDP by adding the FEC type of VC FEC. Moreover, as the two PEs exchanging VC labels may not be connected directly, a remote LDP session must be set up to transfer the VC FEC and VC labels.

With Martini MPLS L2VPN, only PEs must maintain a small amount of VC labels and LSP mappings and no P device contains Layer 2 VPN information. Therefore, it has high scalability. In addition, to add a new VC, you only need to configure a one-way VC for each of the PEs. Your configuration does not affect the operation of the network.

The Martini method applies to scenarios with sparse Layer 2 connections, such as a scenario with a star topology.

Kompella MPLS L2VPN

Kompella MPLS L2VPN is different from Martini MPLS L2VPN in that it does not operate on the connections between CEs directly. It organizes different VPNs in the whole service provider network and encodes each CE in a VPN. For a connection to be established between two CEs, perform these tasks on the PEs:

- Configure CE IDs of the local and remote CEs respectively
- Specify the circuit ID that the local CE assigns to the connection, such as the VPI/VCI with ATM.

Kompella MPLS L2VPN uses extended BGP as the signaling protocol to distribute VC labels. Its label block mode allows it to assign labels to multiple connections at a time.

With Kompella MPLS L2VPN, you can specify the CE range of a VPN to indicate how many CEs can be connected to the VPN. Then, the system assigns a label block of a size equal to the CE range for the CE. In this way, you can reserve some labels for the VPN for future use. This wastes some label resources in a short term, but can reduce the VPN deployment and configuration workload in the case of expansion.

Assume that an enterprise VPN contains 10 CEs and the number may increase to 20 in future service expansion. In this case, set the CE range of each CE to 20. When you add a CE to the VPN later, you only need to modify the configurations of the PE to which the new CE is connected. No change is required for the other PEs. This makes VPN expansion extremely simple.

Similar to MPLS L3VPN, Kompella MPLS L2VPN also uses VPN targets to identify VPNs. This brings excellent VPN networking flexibility.

A5800 switch series support only the remote Kompella connection mode.

MPLS L2VPN configuration task list

Complete the following tasks to configure MPLS L2VPN:

Task	Remarks
Configuring MPLS L2VPN	Required.
Configuring a PE interface connecting a CE	Required.
Configuring CCC MPLS L2VPN	Use one of the approaches according to the MPLS L2VPN implementation method.
Configuring SVC MPLS L2VPN	
Configuring Martini MPLS L2VPN	
Configuring Kompella MPLS L2VPN	
Inspecting VCs	Optional.

Configuring MPLS L2VPN

Select any of the implementation methods for MPLS L2VPN as needed. However, no matter what method you select, you must complete the following tasks:

- Configure MPLS basic capability
- Enable L2VPN
- Enable MPLS L2VPN

To complete the basic MPLS L2VPN configurations:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Configure the LSR ID.	**mpls lsr-id** *lsr-id*	Required.
3. Configure MPLS basic capability and enter MPLS view.	**mpls**	Required.
4. Return to system view.	**quit**	—
5. Enable L2VPN and enter L2VPN view.	**l2vpn**	Required. Disabled by default.
6. Enable MPLS L2VPN.	**mpls l2vpn**	Required.

Configuring a PE interface connecting a CE

A PE interface connecting a CE can use these encapsulation types:

- Ethernet
- VLAN

Configuring a PE interface connecting a CE to use Ethernet

- An Ethernet interface can use the encapsulation type of Ethernet. For Ethernet interface configuration information, see *Layer 2—LAN Switching Configuration Guide*.
- A VLAN interface using the link type of access can use the encapsulation type of Ethernet. For configuration information about VLAN interface and link type, see *Layer 2—LAN Switching Configuration Guide*.

Configuring a PE interface connecting a CE to use VLAN

A VLAN interface using the link type of trunk or hybrid can use the encapsulation type of VLAN (the VLAN interface and the CE must reside in the same VLAN). For configuration information about VLAN interface and link type, see *Layer 2—LAN Switching Configuration Guide*.

Configuring CCC MPLS L2VPN

Configuration prerequisites

Before you configure CCC L2VPN, complete the following tasks:

- Configure MPLS basic capability for the MPLS backbone on the PEs and P devices.
- Enable MPLS L2VPN on the PEs. You do not need to enable MPLS L2VPN on the P devices.

To configure CCC MPLS L2VPN, you need the following data:

- Name for the CCC connection
- For a remote CCC connection: the type and number of the incoming interface, the address of the next hop or the type and number of the outgoing interface, and the incoming and outgoing labels of the LSRs along the CCC connection

Configuration procedure

Configuring the remote CCC connection

CAUTION:

- You do not need to configure two static LSPs for each remote CCC connection. Instead, you only need to configure the incoming and outgoing labels, where the incoming label must be exclusively for the CCC connection. The labels function as static LSPs.
- When you configure an MPLS L2VPN connection in CCC mode on a VLAN interface, the VLAN interface can provide MPLS L2VPN services for only one of its attached CEs. You must add only the access port of that CE into the corresponding VLAN, so that data received from that CE can be transmitted over the MPLS L2VPN connection.
- With CCC, no static LSPs are required on the PEs but dedicated bidirectional static LSPs are required on all P devices between the PEs for transmitting the data of the CCC connection.
- For static LSP configuration commands, see *MPLS Command Reference*.

To configure a PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a remote CCC connection between CEs connected to different PEs.	**ccc** *ccc-connection-name* **interface** *interface-type interface-number* **in-label** *in-label-value* **out-label** *out-label-value* **nexthop** *ip-address* [**control-word** | **no-control-word**]	Required.

To configure a P device:

Step	Command	Remarks
1. Enter system view.	**system-view**	—

Step	Command	Remarks
2. Configure a transit static LSP.	**static-lsp transit** *lsp-name* **incoming-interface** *interface-type interface-number* **in-label** *in-label* **nexthop** *next-hop-addr* **out-label** *out-label*	Required.

Configuring SVC MPLS L2VPN

SVC MPLS L2VPN does not use any signaling protocol to transfer L2VPN information. Instead, it uses tunnels to transport data between PEs.

SVC supports these tunnel types: LDP LSP and CR-LSP. By default, LDP LSP tunnels are used.

Prerequisites

Before configure SVC MPLS L2VPN, complete the following tasks:

- Configure an IGP on the PEs and P devices to guarantee the IP connectivity of the MPLS backbone
- Configure MPLS basic capability and MPLS LDP on the PEs and P devices to establish LDP LSPs
- Enable MPLS L2VPN on the PEs
- For VLAN access, configure a subinterface; for ATM access, configure a VC
- Establish the tunnels between PEs according to the tunneling policy.

To configure SVC MPLS L2VPN, you need the following data:

- Types and numbers of the interfaces connecting the CEs
- Destination LSR ID of SVC
- Incoming and outgoing labels of the L2VPN connection
- SVC tunneling policy

Procedure

To configure SVC MPLS L2VPN on the PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view for the interface connecting the CE.	**interface** *interface-type interface-number*	—
3. Create an SVC MPLS L2VPN connection.	**mpls static-l2vc destination** *destination-router-id* **transmit-vpn-label** *transmit-label-value* **receive-vpn-label** *receive-label-value* [**tunnel-policy** *tunnel-policy-name*] [**control-word** | **no-control-word**]	Required.

When you configure an MPLS L2VPN connection in SVC mode on a VLAN interface, the VLAN interface can provide MPLS L2VPN services for only one of its attached CEs. You must add only the access port of

that CE into the corresponding VLAN, so that data received from that CE can be transmitted over the MPLS L2VPN connection.

Configuring Martini MPLS L2VPN

Martini MPLS L2VPN uses extended LDP to transfer Layer 2 information and VC labels. To configure Martini MPLS L2VPN, complete the following tasks:

1. Configure the remote peer

In Martini MPLS L2VPN implementation, VC labels must be exchanged between PEs. As two PEs may not be connected to each other directly, you must establish a remote session between the two PEs, so that VC FECs and VC labels can be transferred through the session.

2. Create a Martini MPLS L2VPN connection

Create a Martini MPLS L2VPN connection in either of the following ways:

- Configure it on a VLAN interface: In this way, packets arriving at this interface are forwarded through the created MPLS L2VPN connection. If the VLAN interface is a VLAN interface, all packets carrying the tag of the VLAN is forwarded through the MPLS L2VPN connection, no matter which Ethernet ports that arrive at. The device chooses an MPLS L2VPN connection for a received packet according to only the VLAN tag carried in the packet. In this way, it is not possible to differentiate the users and services of different Ethernet ports. Create the MPLS L2VPN connection in this way when packets of all users connected to a VLAN interface must be forwarded through the same MPLS L2VPN connection.
- Configure it in a service instance: In this way, the device matches packets received on the Ethernet port according to the service instance. Packets matching the service instance are forwarded through the MPLS L2VPN connection. A service instance can match all packets received on the interface, packets carrying the specified VLAN tags, all tagged packets, or packets with no VLAN tags, providing a more flexible access to an MPLS L2VPN connection. Create the MPLS L2VPN connections in this way when packets of the users connected to the same VLAN interface must be forwarded through different MPLS L2VPN connections. For more information about service instance, see "VPLS configuration."

Configuration prerequisites

Before you configure Martini MPLS L2VPN, complete the following tasks:

- Configure an IGP on the PEs and P devices to guarantee the IP connectivity of the MPLS backbone
- Configure MPLS basic capability and MPLS LDP on the PEs and P devices to establish LDP LSPs
- Enable MPLS L2VPN on the PEs

To configure Martini MPLS L2VPN, you need the following data:

- Types and numbers of the interfaces connecting the CEs
- Destination address of the L2VPN connection and the PW ID, or, VC ID
- PW class template

Configuring the remote peer

To configure the remote peer for a PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Configure the remote peer.	**mpls ldp remote-peer** *remote-peer-name*	Required.
	remote-ip *ip-address*	Required.

For remote peer configuration information, see "MPLS basics configuration."

Creating a Martini MPLS L2VPN connection on a VLAN interface

CAUTION:

A Martini connection has two main parameters: IP address of the peer PE, and VC ID. The combination of VC ID and encapsulation type must be unique on a PE. Changing the encapsulation type may result in VC ID conflicts.

On a PE, To create a Martini MPLS L2VPN connection on a VLAN interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter the view of the interface connecting the CE.	**interface** *interface-type interface-number*	—
3. Create a Martini MPLS L2VPN connection.	**mpls l2vc** *destination vcid* [**tunnel-policy** *tunnel-policy-name*] [**control-word** | **no-control-word**]	Required.

When you configure an MPLS L2VPN connection in Martini mode on a VLAN interface, the VLAN interface can provide MPLS L2VPN services for only one of its attached CEs. You must add only the access port of that CE into the corresponding VLAN, so that data received from that CE can be transmitted over the MPLS L2VPN connection.

Creating a Martini MPLS L2VPN connection for a service instance

To complete this task, perform the following configurations:

- Create a service instance on an Ethernet port
- Configure a packet matching rule for the service instance
- Create a Martini MPLS L2VPN connection on the service instance

After these configurations, packets arriving at the Ethernet port and matching the packet matching rule is forwarded by the created MPLS L2VPN connection.

To create a Martini MPLS L2VPN connection for a service instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—

Step	Command	Remarks
2. Create a PW class template and enter PW class template view.	**pw-class** *pw-class-name*	Optional. By default, no PW class template is created.
3. Specify the PW transport mode.	**trans-mode** { **ethernet** | **vlan** }	Optional. VLAN by default.
4. Specify the tunneling policy.	**pw-tunnel-policy** *policy-name*	Optional. By default, the default tunneling policy is used, which selects only one tunnel (with no load balancing) in this order: LSP tunnel, CR-LSP tunnel. For how to configure a tunneling policy, see "MPLS L3VPN configuration."
5. Return to system view.	**quit**	—
6. Enter the view of the interface connecting the CE.	**interface** *interface-type interface-number*	—
7. Create a service instance and enter service instance view.	**service-instance** *instance-id*	Required. By default, no service instance is created.
8. Configure a packet matching rule for the service instance.	**encapsulation** { **port-based** | **s-vid** *vlan-id* [**only-tagged**] | **tagged** | **untagged** }	Required. By default, no packet matching rule is configured for the service instance.
9. Create a Martini MPLS L2VPN connection for the service instance.	**xconnect peer** *peer-ip-address* **pw-id** *pw-id* [**pw-class** *pw-class-name*] [**mtu** *mtu-value*] [**access-mode** { **ethernet** | **vlan** }]	Required. After this command is executed, the VLAN ID, access mode, and MTU configured for the service instance cannot be changed. To modify these parameters, you must use the **undo xconnect peer** command to remove the L2VPN connection first.
10. Display information about one or all service instances configured on the interface.	**display service-instance interface** *interface-type interface-number* [**service-instance** *instance-id*] [| { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.

Xconnect peer is available for service instances with the ID ranging from 1 to 4094.

For detailed information about commands **service-instance**, **encapsulation**, and **display service-instance interface**, see *MPLS Command Reference*.

Configuring Kompella MPLS L2VPN

Kompella MPLS L2VPN uses extended BGP as the signaling protocol to transfer L2VPN information between PEs.

Prerequisites

Before you configure Kompella MPLS L2VPN, complete the following tasks:

- Configure an IGP on the PEs and P devices to guarantee the IP connectivity of the MPLS backbone
- Configure MPLS basic capability and MPLS LDP on the PEs and P devices to establish LDP LSPs
- Enable MPLS L2VPN on the PEs

To configure Kompella MPLS L2VPN, you need the following data:

- AS numbers of the local PE and the peer PE
- Name, RD, and VPN Target attributes of the L2VPN connection
- CE name, CE ID, and CE range
- CE offset

Procedure

Configuring BGP L2VPN capability

To configure BGP L2VPN capability:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Establish the peer relationship with the peer PE.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
4. Specify the interface for the TCP connection.	**peer** { *group-name* | *ip-address* } **connect-interface** *interface-type interface-number*	Required.
5. Enter BGP L2VPN address family view.	**l2vpn-family**	Required.
6. Enable the filtering by the VPN target extended community attributes for the received routing information.	**policy vpn-target**	Optional. Enabled by default.
7. Enable the specified peer or peers to exchange BGP routing information of the BGP-L2VPN address family.	**peer** { *group-name* | *ip-address* } **enable**	Required.

For information about the configuration of BGP-L2VPN address family, see "MPLS L3VPN configuration."

Configuring a VPN

CAUTION:

- HP does not recommend using **mtu**. It affects only parameter negotiation, which may occur; it does not affect data forwarding.
- With Kompella MPLS L2VPN, you must create on the PE an L2VPN instance for each VPN where a directly connected CE resides. When creating an L2VPN, you must specify an encapsulation type matching that of the CE side interface.
- The configuration of the VPN targets and RD are the same as that for MPLS L3VPN. For Kompella MPLS L2VPN, the RD is required. Once configured, an RD cannot be changed, unless you delete the L2VPN and then re-create it.

To configure a VPN:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a VPN and enter MPLS L2VPN view.	**mpls l2vpn** *vpn-name* [**encapsulation** { **ethernet** | **vlan** } [**control-word** | **no-control-word**]]	Required.
3. Configure an RD for the L2VPN.	**route-distinguisher** *route-distinguisher*	Required.
4. Associate a particular VPN with one or more VPN targets.	**vpn-target** *vpn-target*&<1-16> [**both** | **export-extcommunity** | **import-extcommunity**]	Required.
5. Set the Layer 2 MTU for the VPN.	**mtu** *mtu*	Optional.

Creating a CE connection

CE ID is used for uniquely identifying a CE in a VPN. To facilitate the configuration, encode the CE IDs in continuous natural numbers starting from 1.

The CE range of a VPN indicates the maximum number of CEs that can be connected to the VPN. You can configure a CE range greater than what is required based on your estimate of the future VPN expansion if the label resources are abundant (they are usually abundant). This can reduce the configuration modification required when CEs are added into the VPN in future.

When creating a CE connection, if you do not specify the CE offset, the following are true:

- For the first connection of the CE, the CE offset is the value specified by the **default-offset** parameter in **ce**.
- For any other connection of the CE, the CE offset is that of the former connection plus 1.
- When you plan a VPN, HP recommends you to encode CE IDs in incremental sequence starting from 1 and then configure connections in the sequence of the CE IDs, in which case you can omit the **ce-offset** keyword (use the default setting) for most of the connections.

You can only increase the CE range. For example, if the original CE range is 10, you can increase it to 20, but cannot reduce it to 5. The only way to reduce the CE range is to delete the CE and re-create it.

When you increase the CE range, for example, from 10 to 20, the system does not release the original label block and then re-apply for a new label block of the size of 20. Instead, the system applies for a supplementary label block of the size of 10. This ensures that the existing services are not interrupted.

To create a CE connection:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter MPLS L2VPN view.	**mpls l2vpn** *vpn-name*	—
3. Create a CE for a VPN and enter MPLS L2VPN CE view.	**ce** *ce-name* [**id** *ce-id* [**range** *ce-range*] [**default-offset** *ce-offset*]]	Required.
4. Create a Kompella connection.	**connection** [**ce-offset** *id*] **interface** *interface-type interface-number* [**tunnel-policy** *tunnel-policy-name*]	Required.

When you configure an MPLS L2VPN connection in Kompella mode on a VLAN interface, the VLAN interface can provide MPLS L2VPN services for only one of its attached CEs. You must add only the access port of that CE into the corresponding VLAN, so that data received from that CE can be transmitted over the MPLS L2VPN connection.

Inspecting VCs

On a MPLS L2VPN network, you can use the MPLS LSP ping function to check VC connectivity and get necessary information for troubleshooting VC failures

On the local PE, the MPLS LSP ping function adds the label of the VC to be inspected into MPLS Echo Request messages so that the messages travel along the VC. The local PE determines whether the VC is valid and reachable according to the replied received from the peer PE.

Use the MPLS LSP ping function to inspect a VC:

Task	Command
Use MPLS LSP ping to check the connectivity of a VC.	**ping lsp** [**-a** *source-ip* | **-c** *count* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *wait-time* | **-r** *reply-mode* | **-s** *packet-size* | **-t** *time-out* | **-v**] * **pw** *ip-address* **pw-id** *pw-id*

MPLS LSP ping can be used to inspect only a Martini type of VC.

Displaying and maintaining MPLS L2VPN

Use the following commands to display the MPLS L2VPN information of a centralized device:

Task	Command	Remarks
Display information about CCC connections.	**display ccc** [**ccc-name** *ccc-name* | **type** { **local** | **remote** }] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about specified L2VPN VC interfaces.	**display l2vpn ccc-interface vc-type** { **all** | **bgp-vc** | **ccc** | **ldp-vc** | **static-vc** } [**up** | **down**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.

Task	Command	Remarks
Display information about static VCs configured on the router.	**display mpls static-l2vc** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about Martini VCs configured on the router.	**display mpls l2vc**[**interface** *interface-type interface-number* [**service-instance** *instance-id*] | **remote-info**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about Kompella L2VPN connections.	**display mpls l2vpn connection** [**vpn-name** *vpn-name* [**remote-ce** *ce-id* | **down** | **up** | **verbose**] | **summary** | **interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about L2VPN in the BGP routing table.	**display bgp l2vpn** { **all** | **group** [*group-name*] | **peer** [[*ip-address*] **verbose**] | **route-distinguisher** *rd* [**ce-id** *ce-id* [**label-offset** *label-offset*]] } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display L2VPN information on a PE.	**display mpls l2vpn** [**export-route-target-list** | **import-route-target-list** | **vpn-name** *vpn-name* [**local-ce** | **remote-ce**]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display the MPLS L2VPN AC information.	**display mpls l2vpn fib ac vpws** [**interface** *interface-type interface-number* [**service-instance** *service-instanceid*]] [**slot** *slot-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display the MPLS L2VPN PW information.	**display mpls l2vpn fib pw vpws** [**interface** *interface-type interface-number* [**service-instance** *service-instanceid*]] [**slot** *slot-number*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display the MPLS L2VPN packet statistics of an interface.	**display interface** *interface-type interface-number* [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.
Display information about one or all PW class templates.	**display pw-class** [*pw-class-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view.

For description of **display interface**, see *Layer 2—LAN Switching Command Reference*.

Resetting BGP L2VPN connections

Task	Command	Remarks
Reset BGP L2VPN connections.	**reset bgp l2vpn** { *as-number* | *ip-address* | **all** | **external** | **internal** }	Available in user view.

MPLS L2VPN configuration examples

Example for configuring a remote CCC connection

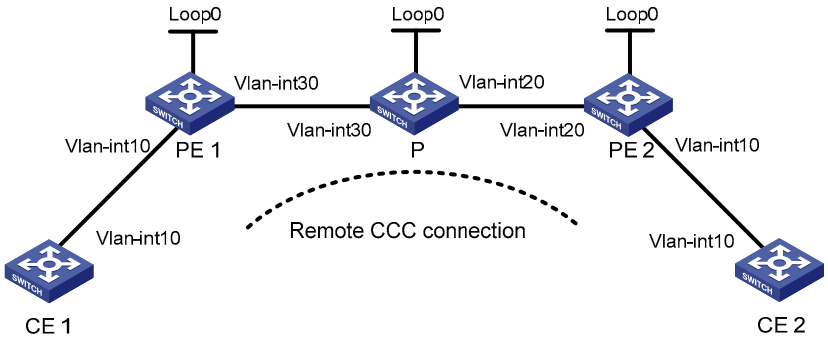
Network requirements

- The CEs are connected to the PEs through VLAN interfaces.
- A remote CCC connection is created between CE 1 and CE 2.

The main steps for configuring a CCC remote connection are:

- Create remote CCC connections on the PEs. No static LSP is required on the PEs.
- Configure two static LSPs on the P device for packets to be transferred in both directions.

Figure 49 Network diagram for configuring a remote CCC connection



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int10	100.1.1.1/24	P	Loop0	10.0.0.2/32
PE 1	Loop0	10.0.0.1/32		Vlan-int20	10.2.2.2/24
	Vlan-int30	10.1.1.1/24		Vlan-int30	10.1.1.2/24
CE 2	Vlan-int10	100.1.1.2/24	PE 2	Loop0	10.0.0.3/32
				Vlan-int20	10.2.2.1/24

Configuration procedure

1. Configure CE 1

```
<Sysname> system-view
[Sysname] sysname CE1
[CE1] interface vlan-interface 10
[CE1-Vlan-interface10] ip address 100.1.1.1 24
```

2. Configure PE 1

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname PE1
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 10.0.0.1 32
[PE1-LoopBack0] quit
[PE1] mpls lsr-id 10.0.0.1
[PE1] mpls
[PE1-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

Configure interface VLAN-interface 30 and enable MPLS.

```
[PE1] interface vlan-interface 30
[PE1-Vlan-interface30] ip address 10.1.1.1 24
[PE1-Vlan-interface30] mpls
[PE1-Vlan-interface30] quit
```

Create a remote connection from CE 1 to CE 2, using the interface connecting CE 1 as the incoming interface and that connecting the P device as the outgoing interface, setting the incoming label to 100 and the outgoing label to 200.

```
[PE1] ccc ce1-ce2 interface vlan-interface 10 in-label 100 out-label 200 nexthop 10.1.1.2
```

3. Configure the P device

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname P
[P] interface loopback 0
[P-LoopBack0] ip address 10.0.0.2 32
[P-LoopBack0] quit
[P] mpls lsr-id 10.0.0.2
[P] mpls
[P-mpls] quit
```

Configure interface VLAN-interface 30 and enable MPLS.

```
[P] interface vlan-interface 30
[P-Vlan-interface30] ip address 10.1.1.2 24
[P-Vlan-interface30] mpls
[P-Vlan-interface30] quit
```

Configure interface VLAN-interface 20 and enable MPLS.

```
[P] interface vlan-interface 20
[P-Vlan-interface20] ip address 10.2.2.2 24
[P-Vlan-interface20] mpls
[P-Vlan-interface20] quit
```

Create a static LSP for forwarding packets from PE 1 to PE 2.

```
[P] static-lsp transit pe1_pe2 incoming-interface vlan-interface 30 in-label 200 nexthop
10.2.2.1 out-label 201
```

Create a static LSP for forwarding packets from PE 2 to PE 1.

```
[P] static-lsp transit pe2_pe1 incoming-interface vlan-interface 20 in-label 101 nexthop
10.1.1.1 out-label 100
```

4. Configure PE 2

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname PE2
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 10.0.0.3 32
[PE2-LoopBack0] quit
[PE2] mpls lsr-id 10.0.0.3
[PE2] mpls
[PE2-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```

Configure interface VLAN-interface 10.

```
[PE2] interface vlan-interface 10
[PE2-Vlan-interface10] quit
```

Configure interface VLAN-interface 20 and enable MPLS.

```
[PE2] interface vlan-interface 20
[PE2-Vlan-interface20] ip address 10.2.2.1 24
[PE2-Vlan-interface20] mpls
[PE2-Vlan-interface20] quit
```

Create a remote connection from CE 2 to CE 1, using the interface connecting CE 2 as the incoming interface and that connecting the P device as the outgoing interface, setting the incoming label to 201 and the outgoing label to 101.

```
[PE2] ccc ce2-ce1 interface vlan-interface 10 in-label 201 out-label 101  nexthop 10.2.2.2
```

5. Configure CE 2

```
<Sysname> system-view
[Sysname] sysname CE2
[CE2] interface vlan-interface 10
[CE2-Vlan-interface10] ip address 100.1.1.2 24
```

6. Verify your configuration

Display CCC connection information on PE 1. A remote CCC connection has been established. CE 1 and CE 2 are able to ping each other.

Display CCC connection information on PE 1.

```
[PE1] display ccc
    Total  ccc vc          : 1
    Local  ccc vc          : 0,  0 up
    Remote ccc vc          : 1,  1 up
 ***Name                   : ce1-ce2
```

```
    Type                : remote
    State               : up
    Intf                : Vlan-interface10 (up)
    In-label            : 100
    Out-label           : 200
    Nexthop             : 10.1.1.2
```

Ping CE 2 from CE 1.

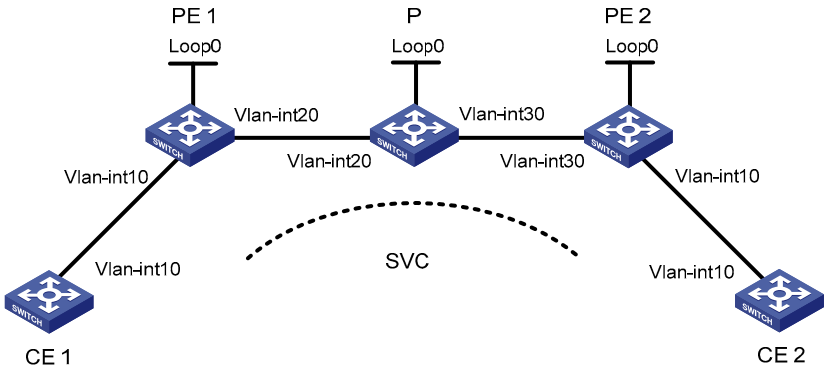
```
[CE1] ping 100.1.1.2
  PING 100.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.2: bytes=56 Sequence=1 ttl=255 time=180 ms
    Reply from 100.1.1.2: bytes=56 Sequence=2 ttl=255 time=60 ms
    Reply from 100.1.1.2: bytes=56 Sequence=3 ttl=255 time=10 ms
    Reply from 100.1.1.2: bytes=56 Sequence=4 ttl=255 time=70 ms
    Reply from 100.1.1.2: bytes=56 Sequence=5 ttl=255 time=60 ms
  --- 100.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 10/76/180 ms
```

Example for configuring SVC MPLS L2VPN

Network requirements

- CEs are connected to PEs through VLAN interfaces.
- An SVC MPLS L2VPN is established between CE 1 and CE 2.

Figure 50 Network diagram for configuring SVC MPLS L2VPN



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int10	100.1.1.1/24	P	Loop0	192.4.4.4/32
PE 1	Loop0	192.2.2.2/32		Vlan-int30	10.2.2.2/24
	Vlan-int20	10.1.1.1/24		Vlan-int20	10.1.1.2/24
CE 2	Vlan-int10	100.1.1.2/24	PE 2	Loop0	192.3.3.3/32
				Vlan-int30	10.2.2.1/24

Procedure

The two main parts of the procedure are:

- Configure MPLS basic forwarding capability on the PEs and P device. This includes configuring the LSR ID, enabling MPLS and LDP, and running IGP (OSPF in this example) between PE 1, the P device, and PE 2 to establish LSPs.
- Establish an SVC MPLS L2VPN connection. This includes enabling MPLS L2VPN on PE 1 and PE 2 and establishing an SVC connection and specifying the VC labels.

Follow these steps.

1. Configure CE 1

```
<Sysname> system-view
[Sysname] sysname CE1
[CE1] interface vlan-interface 10
[CE1-Vlan-interface10] ip address 100.1.1.1 24
```

2. Configure PE 1

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname PE1
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 192.2.2.2 32
[PE1-LoopBack0] quit
[PE1] mpls lsr-id 192.2.2.2
[PE1] mpls
```

Configure the LSP establishment triggering policy.

```
[PE1-mpls] lsp-trigger all
[PE1-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

Enable LDP globally.

```
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

Configure the interface connected with the P device, namely VLAN-interface 20, and enable LDP on the interface.

```
[PE1] interface vlan-interface 20
[PE1-Vlan-interface20] ip address 10.1.1.1 24
[PE1-Vlan-interface20] mpls
[PE1-Vlan-interface20] mpls ldp
[PE1-Vlan-interface20] quit
```

Configure OSPF on PE 1 for establishing LSPs.

```
[PE1] ospf
[PE1-ospf-1] area 0
[PE1-ospf-1-area-0.0.0.0] network 10.1.1.1 0.0.0.255
[PE1-ospf-1-area-0.0.0.0] network 192.2.2.2 0.0.0.0
[PE1-ospf-1-area-0.0.0.0] quit
[PE1-ospf-1] quit
```

On the interface connecting CE 1, namely VLAN-interface 10, create an SVC MPLS L2VPN connection. The interface requires no IP address.

```
[PE1] interface vlan-interface 10
[PE1-Vlan-interface10] mpls static-l2vc destination 192.3.3.3 transmit-vpn-label 100 receive-vpn-label 200
[PE1-Vlan-interface10] quit
```

3. Configure the P device

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname P
[P] interface loopback 0
[P-LoopBack0] ip address 192.4.4.4 32
[P-LoopBack0] quit
[P] mpls lsr-id 192.4.4.4
[P] mpls
```

Configure the LSP establishment triggering policy.

```
[P-mpls] lsp-trigger all
[P-mpls] quit
```

Enable LDP globally.

```
[P] mpls ldp
[P-mpls-ldp] quit
```

Configure the interface connected with PE 1, namely VLAN-interface 20, and enable LDP on the interface.

```
[P] interface vlan-interface 20
[P-Vlan-interface20] ip address 10.1.1.2 24
[P-Vlan-interface20] mpls
[P-Vlan-interface20] mpls ldp
[P-Vlan-interface20] quit
```

Configure the interface connected with PE 2, namely VLAN-interface 30, and enable LDP on the interface.

```
[P] interface vlan-interface 30
[P-Vlan-interface30] ip address 10.2.2.2 24
[P-Vlan-interface30] mpls
[P-Vlan-interface30] mpls ldp
[P-Vlan-interface30] quit
```

Configure OSPF on the P device for establishing LSPs.

```
[P] ospf
[P-ospf-1] area 0
[P-ospf-1-area-0.0.0.0] network 10.1.1.2 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 10.2.2.2 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 192.4.4.4 0.0.0.0
[P-ospf-1-area-0.0.0.0] quit
[P-ospf-1] quit
```

4. Configure PE 2

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname PE2
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 192.3.3.3 32
[PE2-LoopBack0] quit
[PE2] mpls lsr-id 192.3.3.3
[PE2] mpls
```

Configure the LSP establishment triggering policy.

```
[PE2-mpls] lsp-trigger all
[PE2-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```

Enable LDP globally.

```
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

Configure the interface connected with the P device, namely VLAN-interface 30, and enable LDP on the interface.

```
[PE2] interface vlan-interface 30
[PE2-Vlan-interface30] ip address 10.2.2.1 24
[PE2-Vlan-interface30] mpls
[PE2-Vlan-interface30] mpls ldp
[PE2-Vlan-interface30] quit
```

Configure OSPF on PE 2 for establishing LSPs.

```
[PE2] ospf
[PE2-ospf-1] area 0
[PE2-ospf-1-area-0.0.0.0] network 10.2.2.1 0.0.0.255
[PE2-ospf-1-area-0.0.0.0] network 192.3.3.3 0.0.0.0
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit
```

On the interface connecting CE 2, namely VLAN-interface 10, create an SVC MPLS L2VPN connection. The interface requires no IP address.

```
[PE2] interface vlan-interface 10
[PE2-Vlan-interface10] mpls static-l2vc destination 192.2.2.2 transmit-vpn-label 200
receive-vpn-label 100
[PE2-Vlan-interface10] quit
```

5. Configure CE 2

```
<Sysname> system-view
[Sysname] sysname CE2
[CE2] interface vlan-interface 10
[CE2-Vlan-interface10] ip address 100.1.1.2 24
```

6. Verify your configuration

Display SVC L2VPN connection information on PE 1 or PE 2. An L2VPN connection has been established. CE 1 and CE 2 are able to ping each other.

Display SVC L2VPN connection information on PE 1.

```
[PE1] display mpls static-l2vc
Total connections:  1,  1 up,  0 down
ce-intf             state destination     tr-label  rcv-label tnl-policy
Vlan10               up    192.3.3.3       100       200       default
```

Display SVC L2VPN connection information on PE 2.

```
[PE2] display mpls static-l2vc
Total connections:  1,  1 up,  0 down
ce-intf             state destination     tr-label  rcv-label tnl-policy
Vlan20               up    192.2.2.2       200       100       default
```

Ping CE 2 from CE 1.

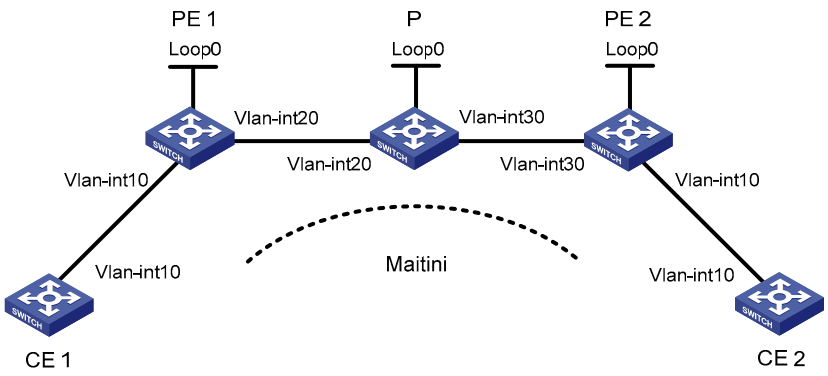
```
[CE1] ping 100.1.1.2
  PING 100.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.2: bytes=56 Sequence=1 ttl=255 time=150 ms
    Reply from 100.1.1.2: bytes=56 Sequence=2 ttl=255 time=130 ms
    Reply from 100.1.1.2: bytes=56 Sequence=3 ttl=255 time=130 ms
    Reply from 100.1.1.2: bytes=56 Sequence=4 ttl=255 time=140 ms
    Reply from 100.1.1.2: bytes=56 Sequence=5 ttl=255 time=80 ms
  --- 100.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 80/126/150 ms
```

Example for configuring Martini MPLS L2VPN on VLAN interfaces

Network requirements

- CEs are connected to PEs through VLAN interfaces.
- A Martini MPLS L2VPN is established between CE 1 and CE 2.

Figure 51 Network diagram for configuring Martini MPLS L2VPN



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int10	100.1.1.1/24	P	Loop0	192.4.4.4/32
PE 1	Loop0	192.2.2.2/32		Vlan-int20	10.1.1.2/24

	Vlan-int20	10.1.1.1/24		Vlan-int30	10.2.2.2/24
CE 2	Vlan-int10	100.1.1.2/24	PE 2	Loop0	192.3.3.3/32
				Vlan-int30	10.2.2.1/24

Procedure

1. Configure CE 1

```
<Sysname> system-view
[Sysname] sysname CE1
[CE1] interface vlan-interface 10
[CE1-Vlan-interface10] ip address 100.1.1.1 24
```

2. Configure PE 1

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname PE1
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 192.2.2.2 32
[PE1-LoopBack0] quit
[PE1] mpls lsr-id 192.2.2.2
[PE1] mpls
```

Configure the LSP establishment triggering policy.

```
[PE1-mpls] lsp-trigger all
[PE1-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

Enable LDP globally.

```
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

Establish a remote session between PE 1 and PE 2.

```
[PE1] mpls ldp remote-peer 1
[PE1-mpls-ldp-remote-1] remote-ip 192.3.3.3
[PE1-mpls-ldp-remote-1] quit
```

Configure the interface connected with the P device, namely VLAN-interface 20, and enable LDP on the interface.

```
[PE1] interface vlan-interface 20
[PE1-Vlan-interface20] ip address 10.1.1.1 24
[PE1-Vlan-interface20] mpls
[PE1-Vlan-interface20] mpls ldp
[PE1-Vlan-interface20] quit
```

Configure OSPF on PE 1 for establishing LSPs.

```
[PE1] ospf
[PE1-ospf-1] area 0
[PE1-ospf-1-area-0.0.0.0] network 10.1.1.1 0.0.0.255
```

```
[PE1-ospf-1-area-0.0.0.0] network 192.2.2.2 0.0.0.0
[PE1-ospf-1-area-0.0.0.0] quit
[PE1-ospf-1] quit
```

On the interface connecting CE 1, namely VLAN-interface 10, create a Martini MPLS L2VPN connection. The interface requires no IP address.

```
[PE1] interface vlan-interface 10
[PE1-Vlan-interface10] mpls l2vc 192.3.3.3 101
[PE1-Vlan-interface10] quit
```

3. Configure the P device

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname P
[P] interface loopback 0
[P-LoopBack0] ip address 192.4.4.4 32
[P-LoopBack0] quit
[P] mpls lsr-id 192.4.4.4
[P] mpls
```

Configure the LSP establishment triggering policy.

```
[P-mpls] lsp-trigger all
[P-mpls] quit
```

Enable LDP globally.

```
[P] mpls ldp
[P-mpls-ldp] quit
```

Configure the interface connected with PE 1, namely VLAN-interface 20, and enable LDP on the interface.

```
[P] interface vlan-interface 20
[P-Vlan-interface20] ip address 10.1.1.2 24
[P-Vlan-interface20] mpls
[P-Vlan-interface20] mpls ldp
[P-Vlan-interface20] quit
```

Configure the interface connected with PE 2, namely VLAN-interface 30, and enable LDP on the interface.

```
[P] interface vlan-interface 30
[P-Vlan-interface30] ip address 10.2.2.2 24
[P-Vlan-interface30] mpls
[P-Vlan-interface30] mpls ldp
[P-Vlan-interface30] quit
```

Configure OSPF on the P device for establishing LSPs.

```
[P] ospf
[P-ospf-1] area 0
[P-ospf-1-area-0.0.0.0] network 10.1.1.2 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 10.2.2.2 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 192.4.4.4 0.0.0.0
[P-ospf-1-area-0.0.0.0] quit
[P-ospf-1] quit
```

4. Configure PE 2

Configure the LSR ID and enable MPLS globally.

```
<Sysname> system-view
[Sysname] sysname PE2
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 192.3.3.3 32
[PE2-LoopBack0] quit
[PE2] mpls lsr-id 192.3.3.3
[PE2] mpls
```

Configure the LSP establishment triggering policy.

```
[PE2-mpls] lsp-trigger all
[PE2-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```

Enable LDP globally.

```
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

Configure an LDP remote session between PE 2 and PE 1.

```
[PE2] mpls ldp remote-peer 2
[PE2-mpls-ldp-remote-2] remote-ip 192.2.2.2
[PE2-mpls-ldp-remote-2] quit
```

Configure the interface connected with the P device, namely VLAN-interface 30, and enable LDP on the interface.

```
[PE2] interface vlan-interface 30
[PE2-Vlan-interface30] ip address 10.2.2.1 24
[PE2-Vlan-interface30] mpls
[PE2-Vlan-interface30] mpls ldp
[PE2-Vlan-interface30] quit
```

Configure OSPF on PE 2 for establishing LSPs.

```
[PE2] ospf
[PE2-ospf-1] area 0
[PE2-ospf-1-area-0.0.0.0] network 192.3.3.3 0.0.0.0
[PE2-ospf-1-area-0.0.0.0] network 10.2.2.0 0.0.0.255
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit
```

On the interface connecting CE 2, namely VLAN-interface 10, create a L2VPN connection. The interface requires no IP address.

```
[PE2] interface vlan-interface 10
[PE2-Vlan-interface10] mpls l2vc 192.2.2.2 101
[PE2-Vlan-interface10] quit
```

5. Configure CE 2

```
<Sysname> system-view
[Sysname] sysname CE2
```

```
[CE2] interface vlan-interface 10
[CE2-Vlan-interface10] ip address 100.1.1.2 24
```

6. Verify your configuration

Display L2VPN connection information on PE 1 or PE 2. An L2VC has been established. CE 1 and CE 2 are able to ping each other.

Display L2VPN connection information on PE 1.

```
[PE1] display mpls l2vc
Total ldp vc : 1     1 up       0 down      0 blocked

Transport   Client               Service  VC       Local     Remote
VC ID       Intf                 ID       State    VC Label  VC Label
101         Vlan10               --       up       8193      8192
```

Display L2VPN connection information on PE 2.

```
[PE2] display mpls l2vc
Total ldp vc : 1     1 up       0 down      0 blocked

Transport   Client               Service  VC       Local     Remote
VC ID       Intf                 ID       State    VC Label  VC Label
101         Vlan10               --       up       8192      8193
```

Ping CE 2 from CE 1.

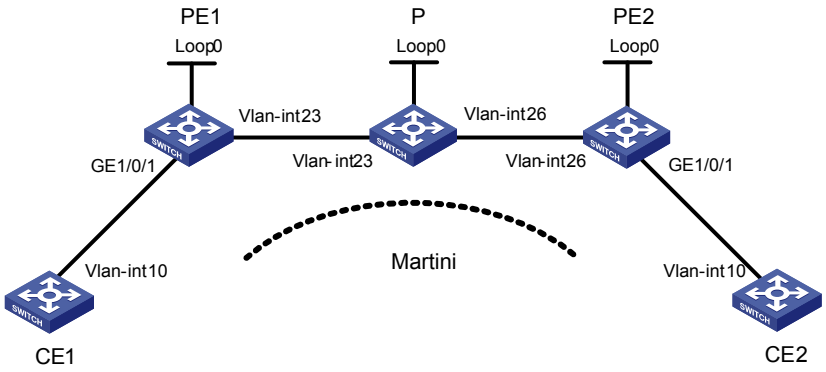
```
[CE1] ping 100.1.1.2
  PING 100.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.2: bytes=56 Sequence=1 ttl=255 time=30 ms
    Reply from 100.1.1.2: bytes=56 Sequence=2 ttl=255 time=60 ms
    Reply from 100.1.1.2: bytes=56 Sequence=3 ttl=255 time=50 ms
    Reply from 100.1.1.2: bytes=56 Sequence=4 ttl=255 time=40 ms
    Reply from 100.1.1.2: bytes=56 Sequence=5 ttl=255 time=70 ms
  --- 100.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 30/50/70 ms
```

Example for configuring Martini MPLS L2VPN for service instances

Netowrk requirments

- CE 1 and CE 2 are connected to PE 1 and PE 2 respectively through VLAN interfaces.
- On PE 1 and PE 2, create MPLS L2VPN connections for CE 1 and CE 2 in service instance view.

Figure 52 Network diagram for configuring MPLS L2VPN connections on service instances



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int10	100.1.1.1/24	P	Loop0	192.4.4.4/32
PE 1	Loop0	192.2.2.2/32		Vlan-int23	23.1.1.2/24
	Vlan-int23	23.1.1.1/24		Vlan-int26	26.2.2.2/24
CE 2	Vlan-int10	100.1.1.2/24	PE 2	Loop0	192.3.3.3/32
				Vlan-int26	26.2.2.1/24

Configuration procedure

1. Configure CE 1

```
<Sysname> system-view
[Sysname] sysname CE1
[CE1] interface vlan-interface 10
[CE1-Vlan-interface10] ip address 100.1.1.1 24
```

2. Configure PE 1

```
<Sysname> system-view
[Sysname] sysname PE1
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 192.2.2.2 32
[PE1-LoopBack0] quit
```

Configure the LSR ID and enable MPLS globally.

```
[PE1] mpls lsr-id 192.2.2.2
[PE1] mpls
[PE1-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

Enable LDP globally.

```
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

Configure PE 1 to establish an LDP remote session with PE 2.

```
[PE1] mpls ldp remote-peer 1
[PE1-mpls-ldp-remote-1] remote-ip 192.3.3.3
```

```
[PE1-mpls-ldp-remote-1] quit
```

Configure the interface connected with the P device and enable LDP on the interface.

```
[PE1] interface vlan-interface 23
[PE1-Vlan-interface23] ip address 23.1.1.1 24
[PE1-Vlan-interface23] mpls
[PE1-Vlan-interface23] mpls ldp
[PE1-Vlan-interface23] quit
```

Configure OSPF.

```
[PE1] ospf
[PE1-ospf-1] area 0
[PE1-ospf-1-area-0.0.0.0] network 23.1.1.1 0.0.0.255
[PE1-ospf-1-area-0.0.0.0] network 192.2.2.2 0.0.0.0
[PE1-ospf-1-area-0.0.0.0] quit
[PE1-ospf-1] quit
```

On the interface connecting CE 1, create a service instance and then establish an MPLS L2VPN connection.

```
[PE1] interface gigabitethernet 1/0/1
[PE1-GigabitEthernet1/0/1] port access vlan 10
[PE1-GigabitEthernet1/0/1] service-instance 1000
[PE1-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 10
[PE1-GigabitEthernet1/0/1-srv1000] xconnect peer 192.3.3.3 pw-id 1000
[PE1-GigabitEthernet1/0/1-srv1000] quit
[PE1-GigabitEthernet1/0/1] quit
```

3. Configure the P device

```
<Sysname> system-view
[Sysname] sysname P
[P] interface loopback 0
[P-LoopBack0] ip address 192.4.4.4 32
[P-LoopBack0] quit
```

Configure the MPLS LSR ID and enable MPLS globally.

```
[P] mpls lsr-id 192.4.4.4
[P] mpls
[P-mpls] quit
```

Enable LDP globally.

```
[P] mpls ldp
[P-mpls-ldp] quit
```

Configure the interface connected with PE 1 and enable LDP on the interface.

```
[P] interface vlan-interface 23
[P-Vlan-interface23] ip address 23.1.1.2 24
[P-Vlan-interface23] mpls
[P-Vlan-interface23] mpls ldp
[P-Vlan-interface23] quit
```

Configure the interface connected with PE 2 and enable LDP on the interface.

```
[P] interface vlan-interface 26
[P-Vlan-interface26] ip address 26.2.2.2 24
```

```
[P-Vlan-interface26] mpls
[P-Vlan-interface26] mpls ldp
[P-Vlan-interface26] quit
```

Configure OSPF.

```
[P] ospf
[P-ospf-1] area 0
[P-ospf-1-area-0.0.0.0] network 23.1.1.2 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 26.2.2.2 0.0.0.255
[P-ospf-1-area-0.0.0.0] network 192.4.4.4 0.0.0.0
[P-ospf-1-area-0.0.0.0] quit
[P-ospf-1] quit
```

4. Configure PE 2

```
<Sysname> system-view
[Sysname] sysname PE2
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 192.3.3.3 32
[PE2-LoopBack0] quit
```

Configure the MPLS LSR ID and enable MPLS globally.

```
[PE2] mpls lsr-id 192.3.3.3
[PE2] mpls
[PE2-mpls] quit
```

Enable L2VPN and MPLS L2VPN.

```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```

Enable LDP globally.

```
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

Configure PE 2 to establish a remote LDP connection with PE 1.

```
[PE2] mpls ldp remote-peer 2
[PE2-mpls-ldp-remote-2] remote-ip 192.2.2.2
[PE2-mpls-ldp-remote-2] quit
```

Configure the interface connected with the P device and enable LDP on the interface.

```
[PE2] interface vlan-interface 26
[PE2-Vlan-interface26] ip address 26.2.2.1 24
[PE2-Vlan-interface26] mpls
[PE2-Vlan-interface26] mpls ldp
[PE2-Vlan-interface26] quit
```

Configure OSPF.

```
[PE2] ospf
[PE2-ospf-1] area 0
[PE2-ospf-1-area-0.0.0.0] network 192.3.3.3 0.0.0.0
[PE2-ospf-1-area-0.0.0.0] network 26.2.2.0 0.0.0.255
[PE2-ospf-1-area-0.0.0.0] quit
[PE2-ospf-1] quit
```

On the interface connecting CE 2, create a service instance and establish an MPLS L2VPN connection.

```
[PE2] interface GigabitEthernet1/0/1
[PE2-GigabitEthernet1/0/1] port access vlan 10
[PE2-GigabitEthernet1/0/1] service-instance 1000
[PE2-GigabitEthernet1/0/1-srv1000] encapsulation s-vid 10
[PE2-GigabitEthernet1/0/1-srv1000] xconnect peer 192.2.2.2 pw-id 1000
[PE2-GigabitEthernet1/0/1-srv1000] quit
[PE2-GigabitEthernet1/0/1] quit
```

5. Configure CE 2

```
<Sysname> system-view
[Sysname] sysname CE2
[CE2] interface vlan-interface 10
[CE2-Vlan-interface10] ip address 100.1.1.2 24
```

6. Verify your configuration

Display the L2VPN connection information on PE 1 and PE 2. An L2VC has been established. CE 1 and CE 2 are able to ping each other.

Display the L2VPN connection information on PE 1.

```
[PE1] display mpls l2vc
Total ldp vc : 1     1 up       0 down      0 blocked

Transport   Client               Service  VC       Local     Remote
VC ID       Intf                 ID       State    VC Label  VC Label
1000        GE1/0/1              1000     up       8193      8192
```

Display the L2VPN connection information on PE 2.

```
[PE2] display mpls l2vc
Total ldp vc : 1     1 up       0 down      0 blocked

Transport   Client               Service  VC       Local     Remote
VC ID       Intf                 ID       State    VC Label  VC Label
1000        GE1/0/1              1000     up       8192      8193
```

Ping CE 2 from CE 1.

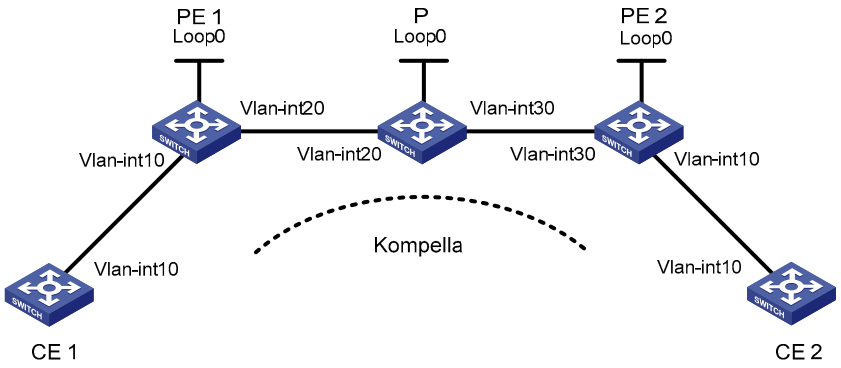
```
[CE1] ping 100.1.1.2
  PING 100.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.2: bytes=56 Sequence=1 ttl=255 time=90 ms
    Reply from 100.1.1.2: bytes=56 Sequence=2 ttl=255 time=77 ms
    Reply from 100.1.1.2: bytes=56 Sequence=3 ttl=255 time=34 ms
    Reply from 100.1.1.2: bytes=56 Sequence=4 ttl=255 time=46 ms
    Reply from 100.1.1.2: bytes=56 Sequence=5 ttl=255 time=94 ms
  --- 100.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 34/68/94 ms
```

Example for configuring Kompella MPLS L2VPN

Network requirements

- CEs are connected to PEs through VLAN interfaces.
- A Kompella MPLS L2VPN is established between CE 1 and CE 2.

Figure 53 Network diagram for configuring Kompella MPLS L2VPN



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int10	100.1.1.1/24	P	Loop0	3.3.3.3/32
PE 1	Loop0	2.2.2.2/32		Vlan-int20	10.1.1.2/24
	Vlan-int20	10.1.1.1/24		Vlan-int30	10.2.2.2/24
CE 2	Vlan-int10	100.1.1.2/24	PE 2	Loop0	4.4.4.4/32
				Vlan-int30	10.2.2.1/24

Procedure

1. Configure IGP on the MPLS backbone

This example uses OSPF. The detailed configuration steps are omitted.

After configuration, issuing **display ip routing-table** on each LSR, you can see that it has learned the routes to the LSR IDs of the other LSRs. Issuing **display ospf peer**, you can see that OSPF adjacencies have been established and reached the Full state.

2. Configure MPLS basic capability and LDP to establish LDP LSPs

The detailed configuration steps are omitted.

After configuration, issue **display mpls ldp session** and **display mpls ldp peer** to view the LDP sessions and peer relationship established, or **display mpls lsp** to view the LSPs established.

3. Configure BGP L2VPN capability

Configure PE 1.

```
<Sysname> system-view
[Sysname] sysname PE1
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
[PE1] bgp 100
[PE1-bgp] peer 4.4.4.4 as-number 100
[PE1-bgp] peer 4.4.4.4 connect-interface loopback 0
```

```
[PE1-bgp] l2vpn-family
[PE1-bgp-af-l2vpn] policy vpn-target
[PE1-bgp-af-l2vpn] peer 4.4.4.4 enable
[PE1-bgp-af-l2vpn] quit
[PE1-bgp] quit
```

Configure PE 2.

```
<Sysname> system-view
[Sysname] sysname PE2
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
[PE2] bgp 100
[PE2-bgp] peer 2.2.2.2 as-number 100
[PE2-bgp] peer 2.2.2.2 connect-interface loopback 0
[PE2-bgp] l2vpn-family
[PE2-bgp-af-l2vpn] policy vpn-target
[PE2-bgp-af-l2vpn] peer 2.2.2.2 enable
[PE2-bgp-af-l2vpn] quit
[PE2-bgp] quit
```

Issue **display bgp l2vpn peer** on PE 1 and PE 2 to view the peer relationship established between the PEs. The status is Established. The following uses PE 1 as an example:

```
[PE1] display bgp l2vpn peer
BGP local router ID : 2.2.2.2
 Local AS number : 100
 Total number of peers : 1                 Peers in established state : 1
  Peer       V    AS   MsgRcvd   MsgSent  OutQ PrefRcv Up/Down  State
  4.4.4.4    4   100        2         5     0       0  00:01:07 Established
```

4. Configure the L2VPN and the CE connection

Configure PE 1. The configurations of the VLAN interfaces are similar to those for Martini MPLS L2VPN and are omitted.

```
[PE1] mpls l2vpn vpn1 encapsulation vlan
[PE1-mpls-l2vpn-vpn1] route-distinguisher 100:1
[PE1-mpls-l2vpn-vpn1] vpn-target 1:1
[PE1-mpls-l2vpn-vpn1] ce ce1 id 1 range 10
[PE1-mpls-l2vpn-ce-vpn1-ce1] connection ce-offset 2 interface vlan-interface 10
[PE1-mpls-l2vpn-ce-vpn1-ce1] quit
[PE1-mpls-l2vpn-vpn1] quit
```

Configure PE 2.

```
[PE2] mpls l2vpn vpn1 encapsulation vlan
[PE2-mpls-l2vpn-vpn1] route-distinguisher 100:1
[PE2-mpls-l2vpn-vpn1] vpn-target 1:1
[PE2-mpls-l2vpn-vpn1] ce ce2 id 2 range 10
[PE2-mpls-l2vpn-ce-vpn1-ce2] connection ce-offset 1 interface vlan-interface 10
[PE2-mpls-l2vpn-ce-vpn1-ce2] quit
[PE2-mpls-l2vpn-vpn1] quit
```

5. Verify your configuration

Issue **display mpls l2vpn connection** on the PEs. An L2VPN connection has been established between the PEs and the connection is up. CE 1 and CE 2 are able to ping each other. The following uses PE 1 as an example:

Display the MPLS L2VPN connection information on PE 1.

```
[PE1] display mpls l2vpn connection
1 total connections,
connections: 1 up, 0 down, 0 local, 1 remote, 0 unknown
VPN name: vpn1,
1 total connections,
connections: 1 up, 0 down, 0 local, 1 remote, 0 unknown
  CE name: ce1, id: 1,
  Rid type status peer-id         route-distinguisher   intf
  2   rmt  up     4.4.4.4         100:1                 Vlan10
```

Ping CE 2 from CE 1.

```
[CE1] ping 100.1.1.2
  PING 100.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 100.1.1.2: bytes=56 Sequence=1 ttl=255 time=90 ms
    Reply from 100.1.1.2: bytes=56 Sequence=2 ttl=255 time=77 ms
    Reply from 100.1.1.2: bytes=56 Sequence=3 ttl=255 time=34 ms
    Reply from 100.1.1.2: bytes=56 Sequence=4 ttl=255 time=46 ms
    Reply from 100.1.1.2: bytes=56 Sequence=5 ttl=255 time=94 ms
  --- 100.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 34/68/94 ms
```

Troubleshooting MPLS L2VPN

Symptom 1:

After the L2VPN configuration, the peer PEs cannot ping each other. The output of **display mpls l2vc** shows that the VC is down and the remote VC label is invalid (displayed as -).

Analysis:

The reason the VC is down may be that the PEs are configured with different encapsulation types.

Solution:

- Check whether the local PE and the peer PE are configured with the same encapsulation type. If not, the connection is destined to fail.
- Check whether the PEs are configured with the Remote argument and whether the peer addresses are correctly configured.

Configuring MPLS L3VPN

This chapter covers only introduction to and configuration of MPLS L3VPN. For information about MPLS basics, see "MPLS basics configuration." For information about BGP, see *Layer 3—IP Routing Configuration Guide*.

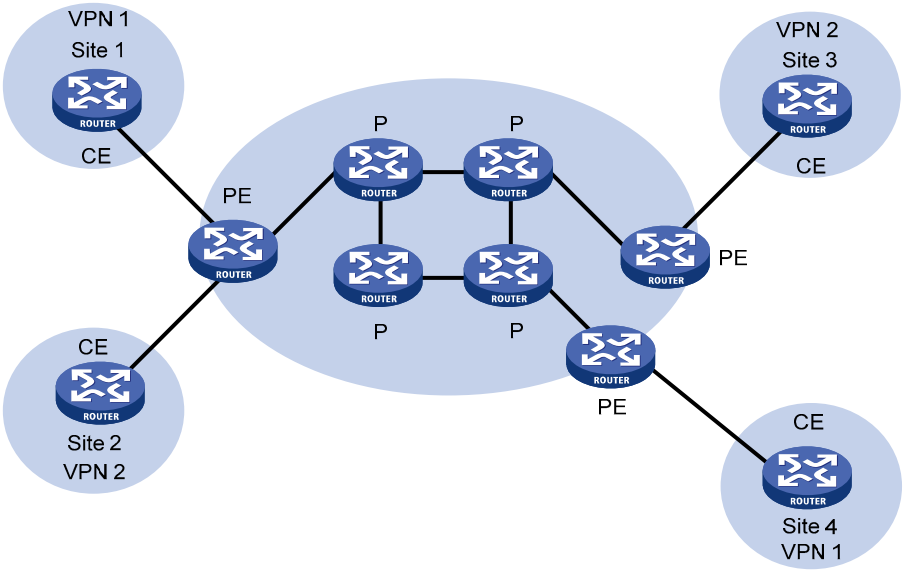
MPLS L3VPN is a kind of PE-based L3VPN technology for service provider VPN solutions. It uses BGP to advertise VPN routes and uses MPLS to forward VPN packets on service provider backbones.

MPLS L3VPN provides flexible networking modes, excellent scalability, and convenient support for MPLS QoS and MPLS TE. Hence, it is widely used.

The MPLS L3VPN model consists of three kinds of devices:

- **CE device**—A CE resides on a customer network and has one or more interfaces directly connected with service provider networks. It can be a router, a switch, or a host. It neither can "sense" the existence of any VPN nor needs to support MPLS.
- **PE device**—A PE resides on a service provider network and connects one or more CEs to the network. On an MPLS network, all VPN processing occurs on the PEs.
- **P device**—A P device is a backbone router on a service provider network. It is not directly connected with any CE. It only needs to be equipped with basic MPLS forwarding capability.

Figure 54 Network diagram for MPLS L3VPN model



CEs and PEs mark the boundary between the service providers and the customers.

After a CE establishes adjacency with a directly connected PE, it advertises its VPN routes to the PE and learns remote VPN routes from the PE. A CE and a PE use BGP/IGP to exchange routing information. You can also configure static routes between them.

After a PE learns the VPN routing information of a CE, it uses BGP to exchange VPN routing information with other PEs. A PE maintains routing information about only VPNs that are directly connected, rather than all VPN routing information on the provider network.

A P router maintains only routes to PEs. It does not need to know anything about VPN routing information.

When VPN traffic travels over the MPLS backbone, the ingress PE functions as the ingress LSR, the egress PE functions as the egress LSR, and P routers function as the transit LSRs.

MPLS L3VPN concepts

Site

Sites are often mentioned in the VPN. 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 to implement.
- 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 adjacent to each other geographically in most cases.
- The devices at a site can belong to multiple VPNs.
- A site is connected to a provider network through one or more CEs. A site can contain many 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.

Address space overlapping

Each VPN independently manages the addresses that it uses. The assembly of such addresses for a VPN is called an address space.

The address spaces of VPNs may overlap. For example, if both VPN 1 and VPN 2 use the addresses on network segment 10.110.10.0/24, address space overlapping occurs.

VPN instance

In MPLS VPN, routes of different VPNs are identified by VPN instance.

A PE creates and maintains a VPN instance for each directly connected site. Each VPN instance contains the VPN membership and routing rules of the corresponding site. If a user at a site belongs to multiple VPNs at the same time, the VPN instance of the site contains information about all VPNs.

For independence and security of VPN data, each VPN instance on a PE maintains a relatively independent routing table and a separate LFIB. VPN instance information contains these items: the LFIB, IP routing table, interfaces bound to the VPN instance, and administration information of the VPN instance. The administration information of the VPN instance includes the RD, route filtering policy, and member interface list.

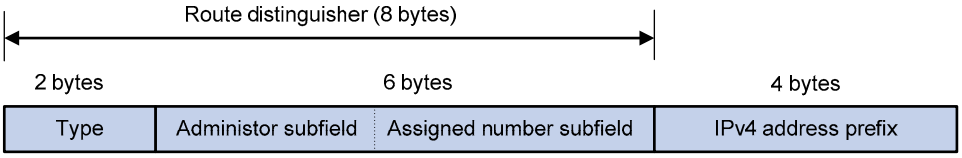
VPN-IPv4 address

Traditional BGP cannot process VPN routes which have overlapping address spaces. If, for example, both VPN 1 and VPN 2 use addresses on the segment 10.110.10.0/24 and each advertise a route to the segment, BGP selects only one of them, which results in loss of the other route.

PEs use MP-BGP to advertise VPN routes, and use VPN-IPv4 address family to solve the problem with traditional BGP.

A VPN-IPv4 address consists of 12 bytes. The first eight bytes represent the RD, followed by a 4-byte IPv4 address prefix, as shown in Figure 55.

Figure 55 VPN-IPv4 address structure



When a PE receives an ordinary IPv4 route from a CE, it must advertise the VPN route to the peer PE. The uniqueness of a VPN route is implemented by adding an RD to the route.

A service provider can independently assign RDs provided the assigned RDs are unique. Thus, a PE can advertise different routes to VPNs even if the VPNs are from different service providers and are using the same IPv4 address space.

Configure a distinct RD for each VPN instance on a PE, so that routes to the same CE use the same RD. The VPN-IPv4 address with an RD of 0 is in fact a globally unique IPv4 address.

By prefixing a distinct RD to a specific IPv4 address prefix, you get a globally unique VPN IPv4 address prefix.

An RD can be related to an AS number, in which case it is the combination of the AS number and a discretionary number; or it can be related to an IP address, in which case it is the combination of the IP address and a discretionary number.

An RD can be in one of the following three formats distinguished by the Type field:

- When the value of 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 value of 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 value of 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 RDs, do not set the Administrator subfield to any private AS number or private IP address.

VPN target attributes

MPLS L3VPN uses the BGP extended community attributes called VPN target attributes, or route target attributes, to control the advertisement of VPN routing information.

A VPN instance on a PE supports two types of VPN target attributes:

- Export target attribute: A local PE sets this type of VPN 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 advertised by other PEs. If the export target attribute matches the import target attribute of the VPN instance, the PE adds the routes to the VPN routing table.

In other words, VPN target attributes define which sites can receive VPN-IPv4 routes, and from which sites that a PE can receive routes.

Like RDs, VPN target attributes can be of three 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 advertises VPN composition information and routes between PEs. It is backward compatible and supports both traditional IPv4 address family and other address families, such as VPN-IPv4 address family.

Using MP-BGP can guarantee that private routes of a VPN are advertised only in the VPN and implement communications between MPLS VPN members.

Routing policy

In addition to the import and export extended communities for controlling VPN route advertisement, you can also configure import and export routing policies to control the injection and advertisement of VPN routes more precisely.

An import routing policy can further filter the routes that can be advertised to a VPN instance by using the VPN target attribute of import target attribute. It can reject the routes selected by the communities in the import target attribute. An export routing policy can reject the routes selected by the communities in the export target attribute.

After a VPN instance is created, you can configure an import routing policy, an export routing policy, or both as needed.

Tunneling policy

A tunneling policy is used to select the tunnel for the packets of a specific VPN instance to use.

After a VPN instance is created, you can optionally configure a tunneling policy for the VPN instance. By default, only one tunnel is selected (no load balancing) in this order: LSP tunnel, CR-LSP tunnel. A tunneling policy only takes effect within the local AS.

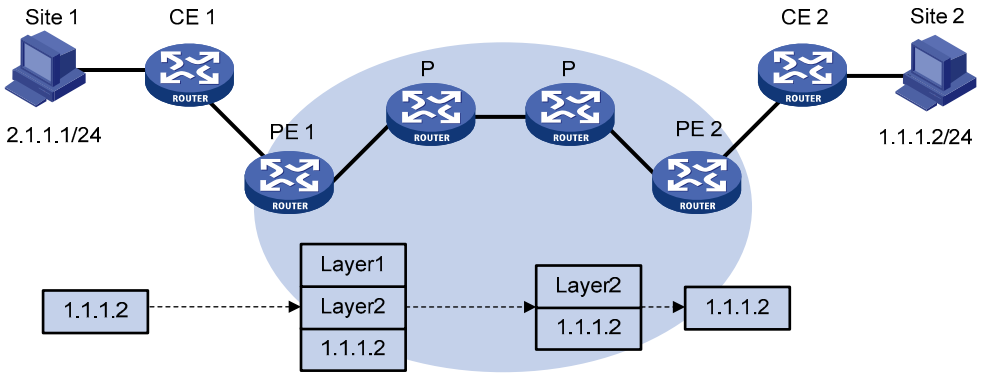
MPLS L3VPN packet forwarding

For basic MPLS L3VPN applications in a single AS, VPN packets are forwarded with two layers of labels:

- Layer 1 labels: Outer labels, used for label switching inside the backbone. They indicate LSPs from the local PEs to the remote PEs. Based on layer 1 labels, VPN packets can be label switched along the LSPs to the remote PEs.
- Layer 2 labels: Inner labels, used for forwarding packets from the remote PEs to the CEs. An inner label indicates to which site, or more precisely, to which CE the packet should be sent. A PE finds the interface for forwarding a packet according to the inner label.

If two sites (CEs) belong to the same VPN and are connected to the same PE, each CE only needs to know how to reach the other CE.

Figure 56 VPN packet forwarding



1. Site 1 sends an IP packet with the destination address of 1.1.1.2. CE 1 transmits the packet to PE 1.
2. PE 1 searches VPN instance entries based on the inbound interface and destination address of the packet. Once finding a matching entry, PE 1 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. PE 2 searches VPN instance entries according to the inner label and destination address of the packet to determine the outbound interface and then forwards the packet out the interface to CE 2.
5. CE 2 transmits the packet to the destination by IP forwarding.

MPLS L3VPN networking schemes

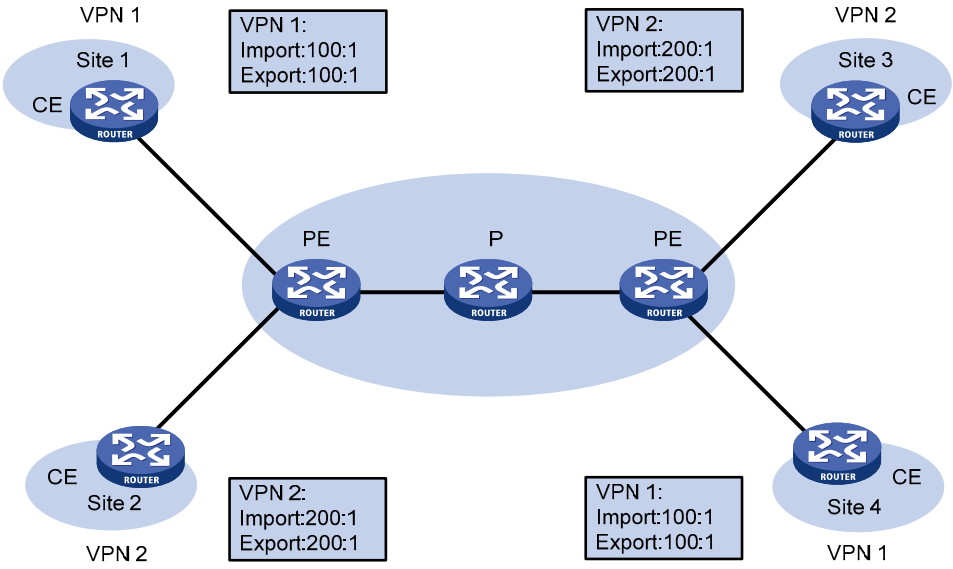
In MPLS L3VPNs, VPN target attributes are used to control the advertisement and reception of VPN routes between sites. They work independently and can be configured with multiple values to support flexible VPN access control and implement multiple types of VPN networking schemes.

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 this networking scheme, the basic VPN networking scheme, you must assign a VPN target to each VPN for identifying the export target attribute and import target attribute of the VPN. Moreover, this VPN target cannot be used by any other VPNs.

Figure 57 Network diagram for basic VPN networking scheme



In Figure 57, for example, the VPN target for VPN 1 is 100:1 on the PEs, and 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

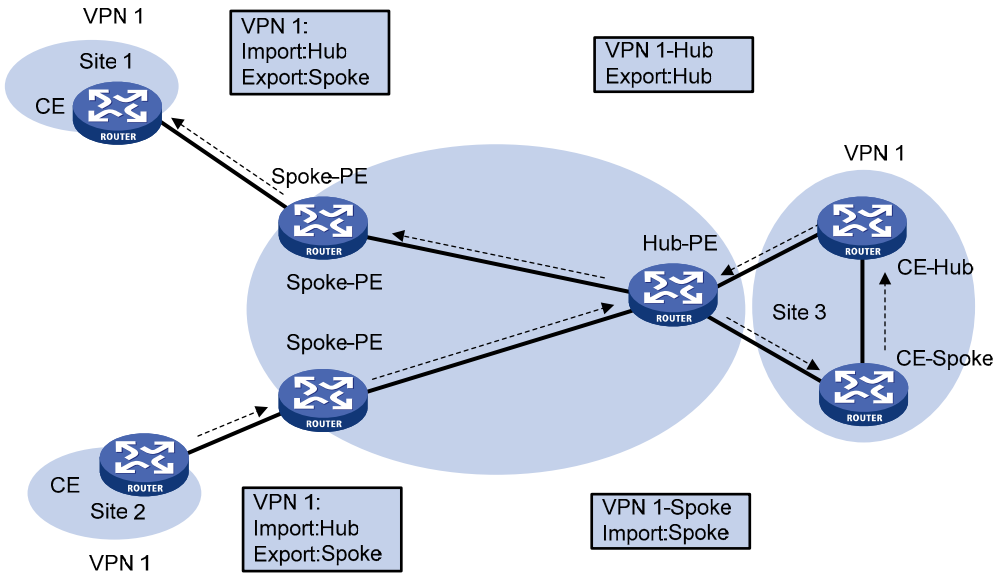
For a VPN where a central access control device is required and all users must communicate with each other through the access control device, the hub and spoke networking scheme can be used to implement the monitoring and filtering of user communications.

This networking scheme requires two VPN targets: one for the “hub” and the other for the “spoke”.

Observe the following VPN target setting rules for VPN instances of all sites on PEs:

- On spoke PEs—the PEs connected with spoke sites—set the export target attribute to Spoke and the import target attribute to Hub.
- On the hub PE—the PE connected to the hub site—specify two interfaces or sub-interfaces, one for receiving routes from spoke PEs, and the other for advertising routes to spoke PEs. Set the import target attribute of the VPN instance for the former to spoke, and the export target attribute of the VPN instance for the latter to hub.

Figure 58 Network diagram for hub and spoke networking scheme



In Figure 58, the spoke sites communicate with each other through the hub site. The arrows in the figure indicate the advertising path of routes from Site 2 to Site 1:

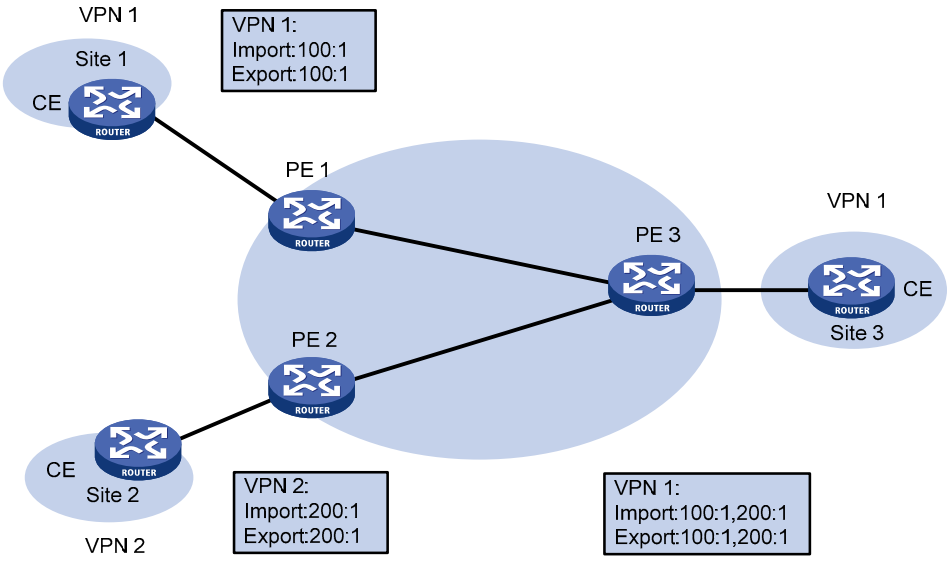
- The hub PE can receive all VPN-IPv4 routes advertised by spoke PEs.
- All spoke PEs can receive the VPN-IPv4 routes advertised by the hub PE.
- The hub PE advertises the routes learned from a spoke PE to the other spoke PEs. Thus, the spoke sites can communicate with each other through the hub site.
- The import target attribute of any spoke PE is distinct from the export VPN targets of the other spoke PEs. Therefore, any two spoke PEs can neither directly advertise VPN-IPv4 routes to each other nor directly access each other.

Extranet networking scheme

The extranet networking scheme can be used when some resources in a VPN are to be accessed by users that are not in the VPN.

In this kind of networking scheme, if a VPN needs to access a shared site, the export target attribute and the import target attribute of the VPN must be contained respectively in the import target attribute and the export target attribute of the VPN instance of the shared site.

Figure 59 Network diagram for extranet networking scheme



In Figure 59, VPN 1 and VPN 2 can access Site 3 of VPN 1.

- PE 3 can receive the VPN-IPv4 routes advertised by PE 1 and PE 2.
- PE 1 and PE 2 can receive the 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. Site 1 of VPN 1 and Site 2 of VPN 2 cannot communicate with each other.

MPLS L3VPN routing information advertisement

In basic MPLS L3VPN networking, the advertisement of VPN routing information involves CEs and PEs. A P router maintains only the routes of the backbone and does not need to know any VPN routing information. A PE maintains only the routing information of the VPNs directly connected to it, rather than that of all VPNs. Therefore, MPLS L3VPN has excellent scalability.

The VPN routing information of a local CE is advertised in three phases:

1. Advertised from the local CE to the ingress PE.
2. Advertised from the ingress PE to the egress PE.
3. Advertised from the egress PE to the remote CE.

Then, a route is available between the local CE and the remote CE, and the VPN routing information can be advertised on the backbone.

The following describes these phases in detail.

Routing information exchange from the local CE to the ingress PE

After establishing an adjacency with the directly connected PE, a CE advertises its VPN routing information to the PE.

The route between the CE and the PE can be a static route, RIP route, OSPF route, IS-IS route, EBGP route, or IBGP route. No matter which routing protocol is used, the CE always advertises standard IPv4 routes to the PE.

Routing information exchange from the ingress PE to the egress PE

After learning the VPN routing information from the CE, the ingress PE adds RDs and VPN targets for these standard IPv4 routes to create VPN-IPv4 routes, save them to the routing table of the VPN instance created for the CE, and then triggers MPLS to assign private labels for them.

Then, the ingress PE advertises the VPN-IPv4 routes to the egress PE through MP-BGP.

Finally, the egress PE compares the export target attribute of the VPN-IPv4 routes with the import target attribute that it maintains for the VPN instance and determines whether to add the routes to the routing table of the VPN instance.

PEs use IGP to ensure the connectivity between them.

Routing information exchange from the egress PE to the remote CE

A remote CE can learn VPN routes from the egress PE in a number of ways. The routes can be static routes, RIP routes, OSPF routes, IS-IS routes, EBGP routes, and IBGP routes. The exchange of routing information between the egress PE and the remote CE is the same as that between the local CE and the ingress PE.

Inter-AS VPN

In some networking scenarios, multiple sites of a VPN may be connected to multiple ISPs in different ASs, or to multiple ASs of an ISP. Such an application is called inter-AS VPN.

RFC 2547bis presents three inter-AS VPN solutions:

- VRF-to-VRF: ASBRs manage VPN routes between them through subinterfaces. This solution is also called inter-AS VPN option A.
- EBGP advertisement of labeled VPN-IPv4 routes: ASBRs advertise labeled VPN-IPv4 routes to each other through MP-EBGP. This solution is also called inter-AS VPN option B.
- Multi-hop EBGP advertisement of labeled VPN-IPv4 routes: PEs advertise labeled VPN-IPv4 routes to each other through MP-EBGP. This solution is also called inter-AS VPN option C.

The following describes these three solutions.

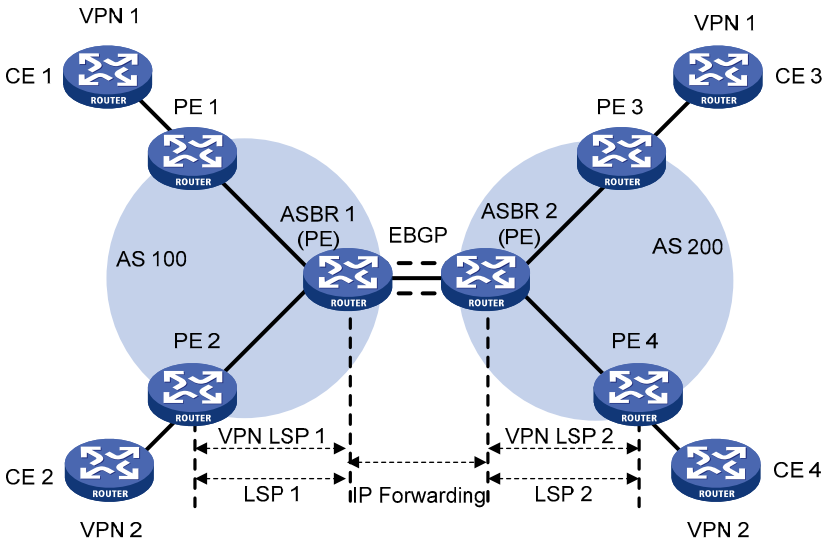
Inter-AS VPN option A

In this kind of 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 of its own and advertises IPv4 routes through conventional EBGP. Within an AS, packets are forwarded using two-level label forwarding as VPN packets. Between ASBRs, conventional IP forwarding is used.

Ideally, each inter-AS VPN has a pair of subinterfaces to exchange VPN routing information.

Figure 60 Network diagram for inter-AS VPN option A



This kind of solution 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 have to manage all VPN routes and create VPN instances on a per-VPN basis. This leads to excessive VPN-IPv4 routes on the PEs. Moreover, the requirement to create a separate subinterface for each VPN also calls for higher performance of the PEs.

Inter-AS VPN option B

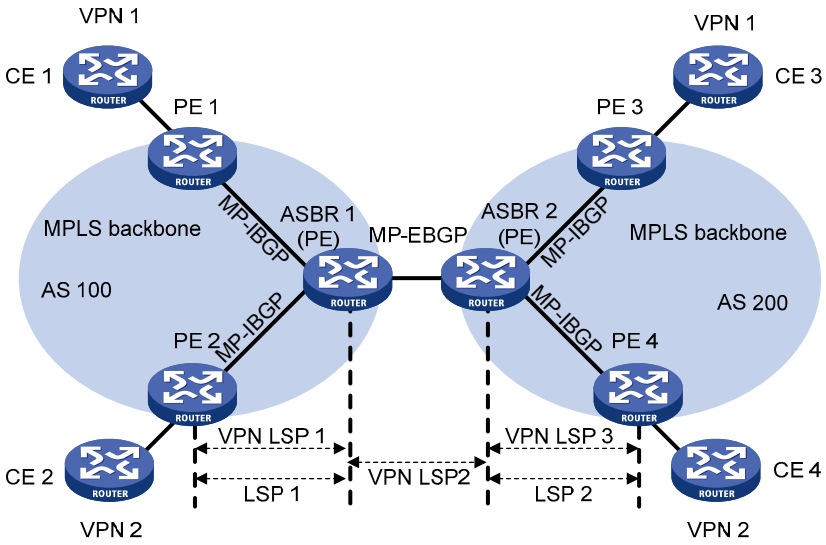
In this kind of solution, two ASBRs use MP-EBGP to exchange labeled VPN-IPv4 routes that they have obtained from the PEs in their respective ASs.

As shown in Figure 61, the routes are advertised through the following steps:

1. PEs in AS 100 advertise labeled VPN-IPv4 routes to the ASBR PE of AS 100 or the RR for 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 for the PEs through MP-IBGP.

The ASBRs must perform the special processing on the labeled VPN-IPv4 routes, which is also called ASBR extension method.

Figure 61 Network diagram for inter-AS VPN option B



In terms of scalability, inter-AS VPN option B is better than option A.

The following issues apply when adopting the MP-EBGP method:

- ASBRs perform no VPN target filtering on VPN-IPv4 routes that they receive from each other. Therefore, the ISPs in different ASs that exchange VPN-IPv4 routes must agree on the route exchange.
- VPN-IPv4 routes are exchanged only between VPN peers. A VPN user 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.

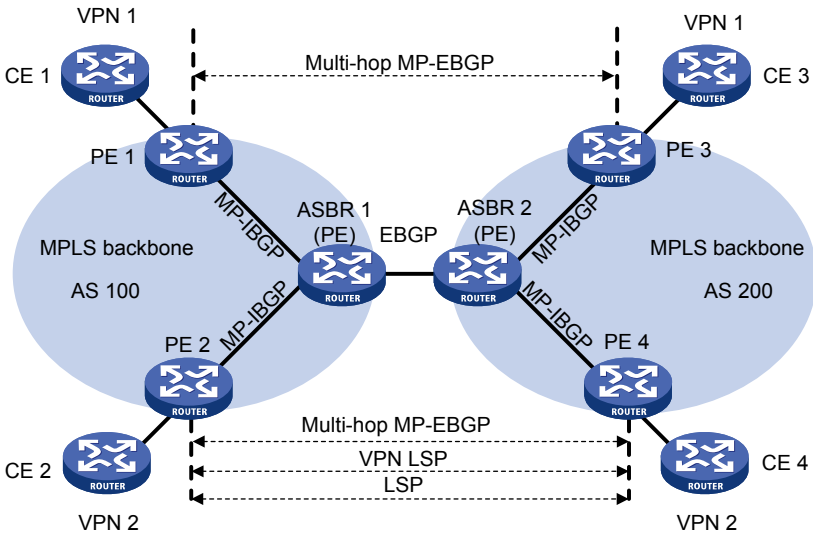
Inter-AS VPN option C

The inter-AS VPN options A and B can satisfy 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 may become bottlenecks hindering network extension.

One way to solve this problem is to make PEs directly exchange VPN-IPv4 routes without the participation of ASBRs:

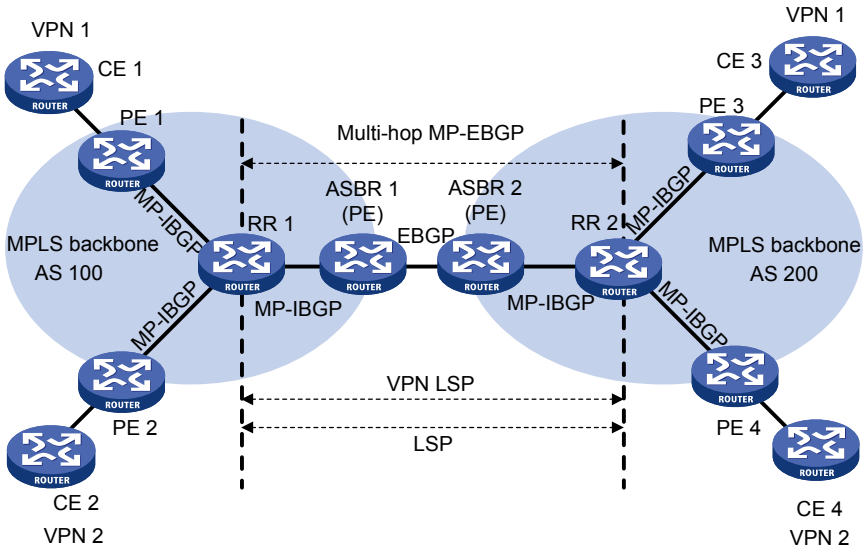
- Two ASBRs advertise labeled IPv4 routes to PEs in their respective ASs through MP-IBGP.
- The ASBRs neither maintain VPN-IPv4 routes nor advertise VPN-IPv4 routes to each other.
- An ASBR maintains labeled IPv4 routes of the PEs in the AS and advertises them to the peers in the other ASs. The ASBR 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 62 Network diagram for inter-AS VPN option C



To improve the scalability, you can specify an RR in each AS, making it maintain all VPN-IPv4 routes and 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 63.

Figure 63 Network diagram for inter-AS VPN option C using RRs



Carrier's carrier

It is possible that a customer of the MPLS L3VPN service provider is also a service provider. In this case, the MPLS L3VPN service provider is called the "provider carrier" or the "Level 1 carrier", and 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 inject the external routes of a Level 2 carrier; it only injects routes for switching packets from different sites of the Level 2 carrier. The external routes

maintained by a Level 2 carrier are exchanged through BGP sessions established between related routes of the Level 2 carrier. This can greatly reduce the number of routes maintained by the Level 1 carrier network.

Implementation of carrier's carrier

CAUTION:

If equal cost routes exist between the Level 1 carrier and the Level 2 carrier, HP recommends you to establish equal cost LSPs between them.

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, or, a Level 2 carrier, accesses a PE of the Level 1 carrier:

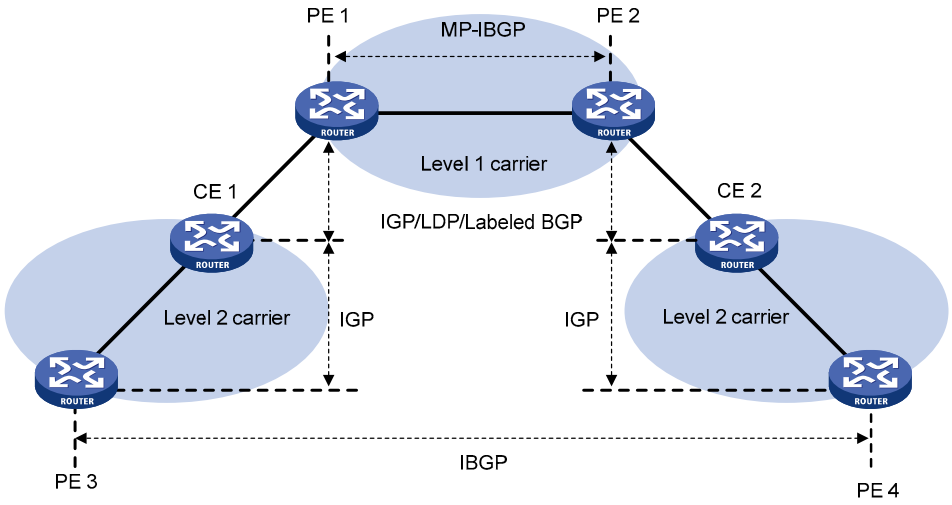
- If the PE and the CE are in a same AS, you must configure IGP and LDP between them.
- If the PE and the CE are not in the same AS, you must configure MP-EBGP to label the routes exchanged between them.

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.

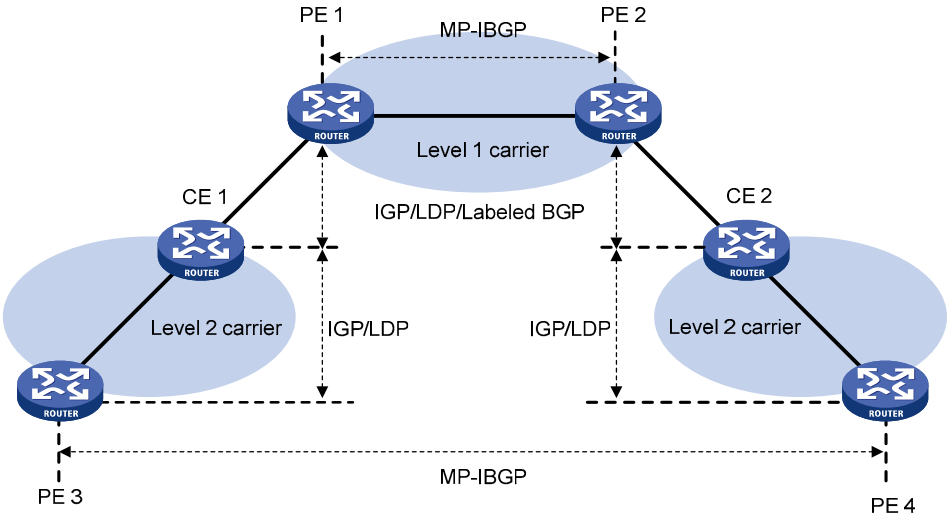
When the Level 2 carrier is an ordinary ISP, its PEs run IGP to communicate with the CEs, rather than MPLS. As shown in Figure 64, PE 3 and PE 4 exchange VPN routes of the Level 2 carrier through IBGP sessions.

Figure 64 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 65, PE 3 and PE 4 exchange VPN routes of the Level 2 carrier through MP-IBGP sessions.

Figure 65 Scenario where the Level 2 carrier is an MPLS L3VPN service provider



Nested VPN

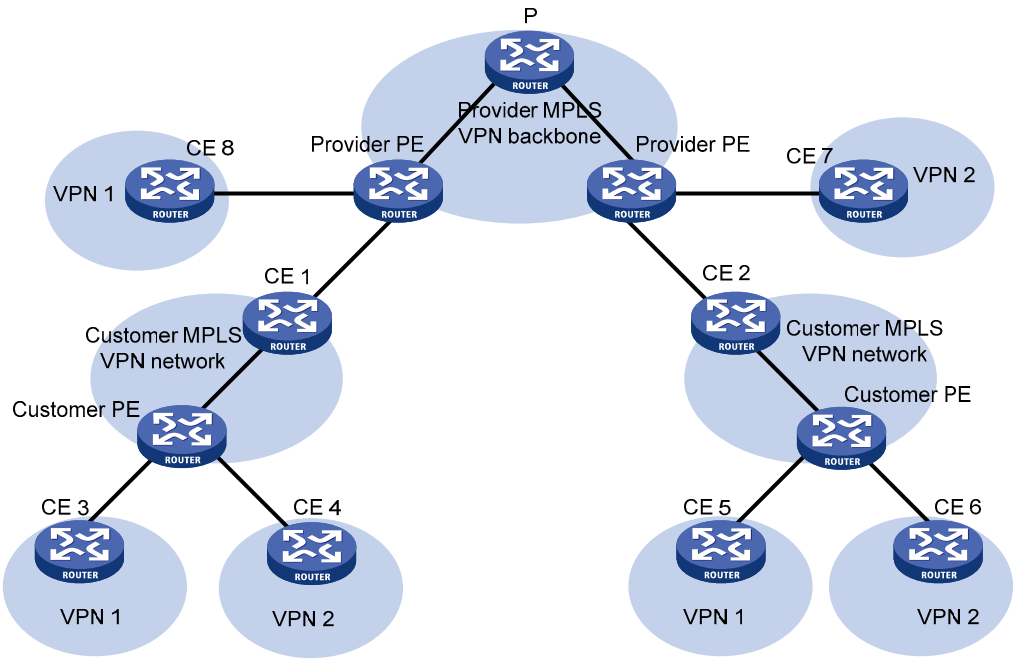
In an MPLS L3VPN network, generally a service provider runs an MPLS L3VPN backbone and provides VPN services through PEs. Different sites of a VPN customer are connected to the PEs through CEs to implement communication. In this scenario, a customer's networks are ordinary IP networks and cannot be further divided into sub-VPNs.

However, in actual applications, customer networks can be dramatically different in form and complexity, and a customer network may need to use VPNs to further group its users. The traditional solution to this request is to implement internal VPN configuration on the service provider's PEs. This solution is easy to deploy, but it increases the network operation cost and brings issues on management and security because:

- The number of VPNs that PEs must support increases sharply.
- Any modification of an internal VPN must be done through the service provider.

The nested VPN technology offers a better solution. It exchanges VPNv4 routes between PEs and CEs of the ISP MPLS L3VPN and allows a customer to manage its own internal VPNs. Figure 66 depicts 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 treat the customer's network as a common VPN user and do not join any sub-VPNs. The customer's CE devices (CE 1, CE 2, CE 7 and CE 8) exchange VPNv4 routes that carry the sub-VPN routing information with the service provider PEs, implementing the propagation of the sub-VPN routing information throughout the customer network.

Figure 66 Network diagram for nested VPN



Propagation of routing information

In a nested VPN network, routing information is propagated in the following process:

1. A provider PE and its CEs exchange VPNv4 routes, which carry information about users' internal VPNs.
2. After receiving a VPNv4 route, a provider PE keeps the user's internal VPN information, and appends the user's MPLS VPN attributes on the service provider network. It replaces the RD of the VPNv4 route with the RD of the user's MPLS VPN on the service provider network and adds the ERT attribute of the user's MPLS VPN on the service provider network to the extended community attribute list of the route. The internal VPN information of the user is maintained on the provider PE.
3. The provider PE advertises VPNv4 routes which carry 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 based on its local VPNs. Each local VPN accepts routes of its own and advertises them to its connected sub-VPN CEs (such as CE 3 and CE 4, or CE 5 and CE 6 in Figure 66). If a CE is connected to a provider PE through an IPv4 connection, the PE advertises IPv4 routes to the CE. If a CE is connected to a provider PE through a VPNv4 connection (a user MPLS VPN network), the PE advertises VPNv4 routes to the CE.

Benefits

The nested VPN technology features the following main 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 levels of nesting of internal VPNs.

Nested VPN is flexible and easy to implement and can reduce the cost because a customer only needs to pay for one MPLS VPN to have multiple internal VPNs connected. Nested VPN provides diversified VPN networking methods for a customer, and allows for multi-level hierarchical access control over the internal VPNs.

HoVPN

1. Hierarchical model and plane model

In MPLS L3VPN solutions, PEs are the key devices. They provide two functions:

- User access. This means that the PEs must have a large amount of interfaces.
- VPN route managing and advertising, and user packet processing. These require that a PE must have a large-capacity memory and high forwarding capability.

Most of the current network schemes use the typical hierarchical architecture. For example, the MAN architecture contains typically three layers, namely, the core layer, convergence layer, and access layer. From the core layer to the access layer, the performance requirements on the devices reduce 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 has limited performance or scalability, the performance or scalability of the whole network is influenced.

Due to the difference, you are faced with the scalability problem when deploying PEs at any of the three layers. Therefore, the plane model is not applicable to the large-scale VPN deployment.

2. HoVPN

To solve the scalability problem of the plane model, MPLS L3VPN must transition to the hierarchical model.

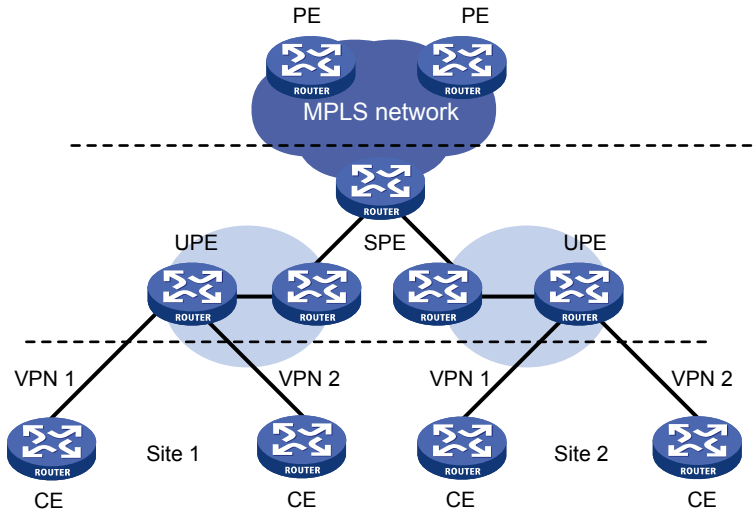
In MPLS L3VPN, HoVPN was proposed to meet that requirement. With HoVPN, the PE functions can be distributed among multiple PEs, which take different roles for the same functions and form a hierarchical architecture.

As in the typical hierarchical network model, HoVPN has different requirements on the devices at different layers of the hierarchy.

Implementation of HoVPN

1. Basic architecture of HoVPN

Figure 67 Basic architecture of HoVPN



As shown in Figure 67, devices directly connected to CEs are called UPEs or user-end PEs, whereas devices that are connected with UPEs and are in the internal network are called SPEs.

The hierarchical PE consists of multiple UPEs and SPEs, which function together as a traditional PE.

With the HoVPN solution, PE functions are implemented hierarchically. Hence, the solution is also called HoPE.

UPEs and SPEs play different roles:

- A UPE allows user access. It maintains the routes of the VPN sites that are directly connected with it, It does not maintain the routes of the remote sites in the VPN, or only maintains their summary routes. A UPE assigns inner labels to the routes of its directly connected sites, and advertises the labels to the SPE along with VPN routes through MP-BGP.
- An SPE manages and advertises VPN routes. It maintains all 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 nodes in a VPN can communicate with each other.

Different roles mean different requirements:

- SPE: An SPE is required to have large-capacity routing table, high forwarding performance, and fewer interface resources.
- UPE: A UPE is required to have small-capacity routing table, low forwarding performance, but higher access capability.

HoVPN takes 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 may 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.

2. SPE-UPE

The MP-BGP running between SPE and UPE can be either MP-IBGP or MP-EBGP. Which one to use depends on whether the UPE and SPE belong to a same AS.

With MP-IBGP, in order 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.

3. 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.

With recursion of HoPEs, a VPN can be extended infinitely in theory.

Figure 68 Recursion of HoPEs

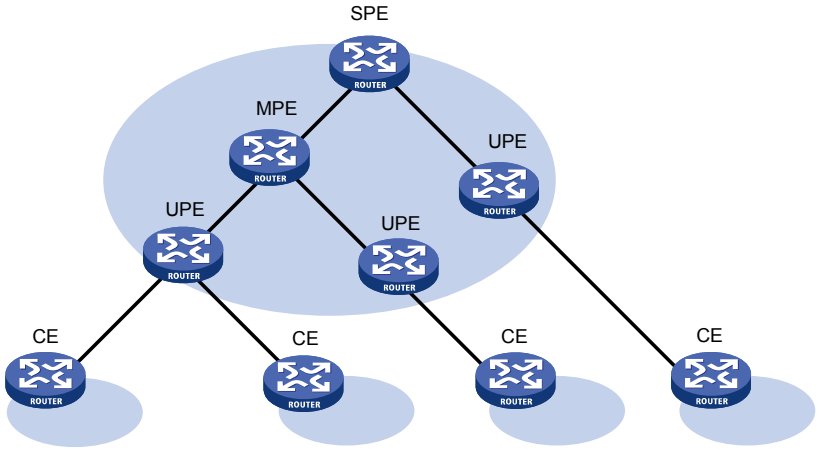


Figure 68 shows a three-level HoPE. The PE in the middle is called the MPE. MP-BGP runs between SPE and MPE, as well as between MPE and UPE.

The term of MPE does not really exist in a HoVPN model. It is used here just for the convenience of description.

MP-BGP advertises all VPN routes of the 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, and each UPE maintains only VPN routes of its directly connected sites. The number of routes maintained by the MPE is in between.

OSPF VPN extension

This section focuses on the OSPF VPN extension. For more information about OSPF, see *Layer 3—IP Routing Configuration Guide*.

OSPF multi-process on a PE

OSPF is a prevalent IGP protocol. In many cases, VPN clients are connected through BGP peers, and the clients often run OSPF. Running OSPF between PEs and CEs can simplify the configuration and management of the CEs, because the CEs only need to support OSPF. In addition, if the customers require MPLS L3VPN services through conventional OSPF backbone, using OSPF between PEs and CEs can simplify the transition.

For OSPF to run between CEs and PEs, the PEs must support multiple OSPF processes. Each OSPF process must correspond to a VPN instance and have its own interface and routing table.

The following describes details of OSPF configuration between PEs and CEs.

1. Configuration of OSPF areas between PEs and CEs

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). Because OSPF requires that the backbone area must be contiguous, the area 0 of each VPN site must be connected to the MPLS VPN backbone.

If a VPN site contains an OSPF area 0, the connected PE must be connected to the backbone area of the VPN site through area 0. You can configure a logical connection by using a virtual link.

2. BGP/OSPF interaction

With OSPF running between PEs and CEs, PEs advertise VPN routes to each other through BGP and to CEs through OSPF.

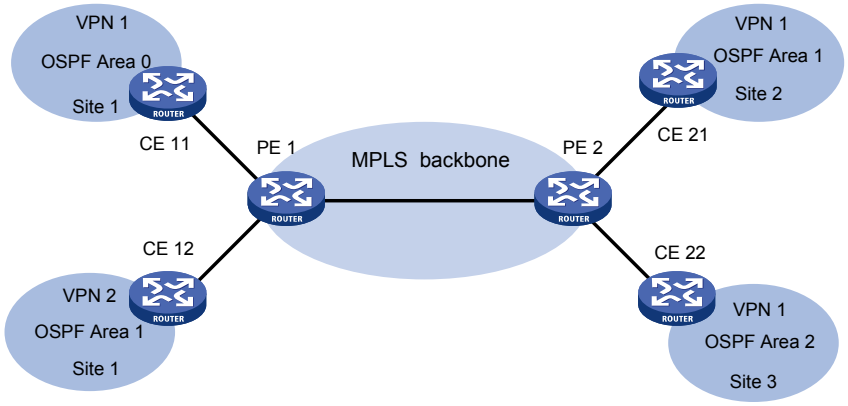
With conventional OSPF, two sites are considered to be in different ASs even if they belong to the same VPN. Therefore, the routes that one site learns are advertised to the other as external routes. This results in higher OSPF traffic and network management problems.

The extended OSPF protocol supports multiple instances and therefore can address the problems. Properly configured, OSPF sites are considered directly connected, and PEs can exchange OSPF routing information as they are using dedicated lines. This improves the network management and makes OSPF applications more effective.

As shown in Figure 69, PE 1 and PE 2 are connected through the MPLS backbone, and CE 11, CE 21, and CE 22 belong to VPN 1. Assumes that all routers in the figure belong to the same AS—CE 11, CE 21, and CE 22 belong to the same OSPF domain. PEs advertise VPN 1 routes in the following procedure:

- At first, PE 1 redistributes OSPF routes from CE 11 into BGP.
- Then, PE 1 advertises the VPN routes to PE 2 through BGP.
- Finally, PE 2 redistributes the BGP VPN routes into OSPF and advertises them to CE 21 and CE 22.

Figure 69 Application of OSPF in VPN



With the standard BGP/OSPF interaction, PE 2 advertises the BGP VPN routes to CE 21 and CE 22 through Type 5 LSAs (ASE LSAs). However, CE 11, CE 21, and CE 22 belong to the same OSPF domain, and the route advertisement between them should use Type 3 LSAs (inter-AS routes).

To solve the problems, PE uses an extended BGP/OSPF interaction process called BGP/OSPF interoperability to advertise routes from one site to another, differentiating the routes from real AS-External routes. The process requires that extended BGP community attributes carry the information for identifying the OSPF attributes.

Each OSPF domain must have a configurable domain ID. HP recommends you to configure the same domain ID for all OSPF instances in the network related to each VPN instance, or adopt the default ID. Thus, the system can know that all VPN routes with the same domain ID are from the same VPN instance.

3. Routing loop detection

If OSPF runs between CEs and PEs and a VPN site is connected to multiple PEs, when a PE advertises the BGP VPN routes learned from MPLS/BGP to the VPN site through LSAs, the LSAs may be received by another PE, resulting in a routing loop.

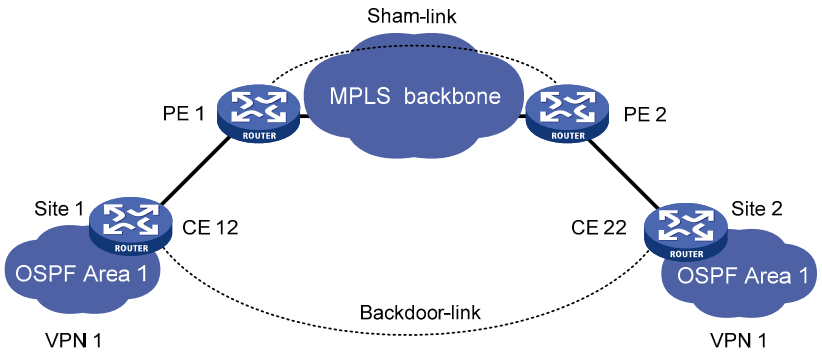
To avoid routing loops, when creating Type 3 LSAs, the PE always sets the flag bit DN for BGP VPN routes learned from MPLS/BGP, regardless of whether the PE and the CEs are connected through the OSPF backbone. When performing route calculation, the OSPF process of the PE ignores the Type 3 LSAs whose DN bit is set.

If the PE needs to advertise to a CE the routes from other OSPF domains, it must indicate that it is the ASBR, and advertise the routes using Type 5 LSAs.

Sham link

Generally, BGP peers carry routing information on the MPLS VPN backbone through the BGP extended community attributes. The OSPF that runs on the remote PE can use the information to create Type 3 summary LSAs to be transmitted to the CEs. As shown in Figure 70, both site 1 and site 2 belong to VPN 1 and OSPF area1. They are connected to different PEs, PE 1 and PE 2. There is an intra-area OSPF link called backdoor link between them. In this case, the route connecting the two sites through PEs is an inter-area route. It is not preferred by OSPF because its preference is lower than that of the intra-area route across the backdoor link.

Figure 70 Network diagram for sham link



To solve the problem, you can establish a sham link between the two PEs so that the routes between them over the MPLS VPN backbone become an intra-area route.

The sham link acts as an intra-area point-to-point link and is advertised through the Type 1 LSA. You can select a route between the sham link and backdoor link by adjusting the metric.

The sham link is considered the link between the two VPN instances with one endpoint address in each VPN instance. The endpoint address is a loopback interface address with a 32-bit mask in the VPN address space on the PE. Different sham links of the same OSPF process can share an endpoint address, but that of different OSPF processes cannot.

BGP advertises the endpoint addresses of sham links as VPN-IPv4 addresses. A route across the sham link cannot be redistributed into BGP as a VPN-IPv4 route.

A sham link can be configured in any area. You must configure it manually. In addition, the local VPN instance must have a route to the destination of the sham link.

When configuring an OSPF sham link between two PEs, redistribute OSPF VPN routes to BGP, but do not redistribute BGP routes to OSPF to avoid route loops.

BGP AS number substitution

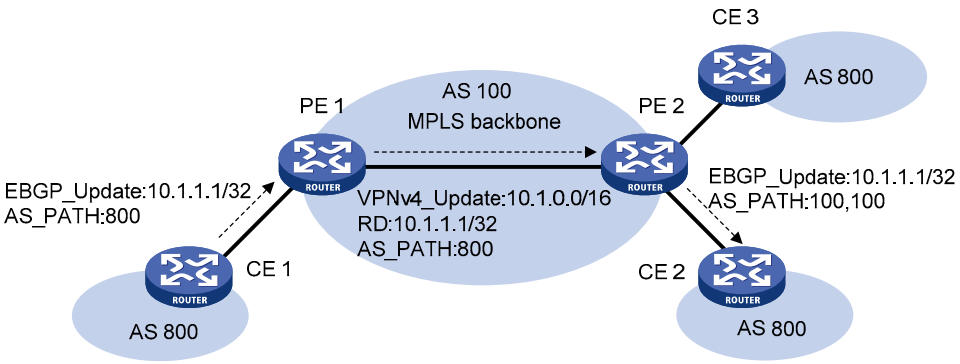
Since BGP detects routing loops by AS number, if EBGP runs between PEs and CEs, you must assign different AS numbers to geographically different sites to ensure correct transmission of the routing information.

The BGP AS number substitution function allows physically dispersed CEs to use the same AS number. The function is a BGP outbound policy and functions on routes to be advertised.

With the BGP AS number substitution function, when a PE advertises a route to a CE of the specified peer, if an AS number identical to that of the CE exist in the AS_PATH of the route, it is replaced with that of the PE.

After you enable the BGP AS number substitution function, the PE re-advertises all routing information to the connected CEs in the peer group, performing BGP AS number substitution based on the above principle.

Figure 71 Application of BGP AS number substitution



In Figure 71, both CE 1 and CE 2 use the AS number of 800. AS number substitution is enabled on PE 2 for CE 2. Before advertising updates received from CE 1 to CE 2, PE 2 finds that an AS number in the AS_PATH is the same as that of CE 2 and hence substitutes its own AS number 100 for the AS number. In this way, CE 2 can normally receive the routing information from CE 1.

AS number substitution also applies to a PE connecting multiple CEs through different interfaces, such as PE 2 in Figure 71, which connects CE 2 and CE 3.

For a multi-homed CE—a CE connected with multiple PEs, the BGP AS number substitution function must be used in combination with the SOO function. Otherwise, routing loops may appear.

MPLS L3VPN configuration task list

Complete the following tasks to configure MPLS L3VPN:

Task	Remarks
Configuring VPN instances	Required.
Configuring basic MPLS L3VPN	Required.

Task	Remarks
Configuring inter-AS VPN	Optional. Configure it as needed
Configuring nested VPN	Optional. Configure it as needed
Configuring HoVPN	Optional. Configure it as needed
Configuring an OSPF sham link	Optional. Configure it as needed
Configuring BGP AS number substitution	Optional. Configure it as needed

Configuring VPN instances

By configuring VPN instances on a PE, you can isolate not only VPN routes from public network routes, but also routes of a VPN from those of another VPN. This feature allows VPN instances to be used in networking scenarios besides MPLS L3VPNs.

All VPN instance configurations are performed on PEs.

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.

A VPN instance only takes effect after you configure an RD for it. Before configuring an RD for a VPN instance, you can configure no parameters for the instance other than a description.

You can configure a description for a VPN instance to record its related information, such as its relationship with a certain VPN.

To create and configure a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a VPN instance and enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required.
3. Configure an RD for the VPN instance.	**route-distinguisher** *route-distinguisher*	Required.
4. Configure a description for the VPN instance.	**description** *text*	Optional.

Associating a VPN instance with an interface

After creating and configuring a VPN instance, you associate the VPN instance with the interface for connecting CEs. Any interface supporting MPLS LDP capability can be associated with a VPN instance. For interfaces supporting MPLS LDP capability, see "MPLS basics configuration."

To associate a VPN instance with an interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view.	**interface** *interface-type interface-number*	—
3. Associate the current interface with the VPN instance.	**ip binding vpn-instance** *vpn-instance-name*	Required. No VPN instance is associated with an interface by default.

Ip binding vpn-instance clears the IP address of the interface on which it is configured. Be sure to re-configure an IP address for the interface after configuring the command.

Configuring route related attributes of a VPN instance

VPN route advertisement follows the following control process:

- When a VPN route learned from a CE gets redistributed into BGP, BGP associates it with a VPN 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 VPN target.
- The VPN instance determines how to change the VPN targets attributes for routes to be advertised according to the **export-extcommunity** in the VPN target.

To configure route related attributes of a VPN instance

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required.
3. Enter IPv4 VPN view.	**ipv4-family**	Optional.
4. Associate the current VPN instance with one or more VPN targets.	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	Required.
5. Configure the maximum number of routes for the VPN instance.	**routing-table limit** *number* { *warn-threshold* | **simply-alert** }	Optional.
6. Apply an import routing policy to the current VPN instance.	**import route-policy** *route-policy*	Optional. By default, all routes permitted by the import target attribute can be redistributed into the VPN instance.
7. Apply an export routing policy to the current VPN instance.	**export route-policy** *route-policy*	Optional. By default, all VPN instance routes permitted by the export target attribute can be redistributed.

A single **vpn-target** command can configure up to eight VPN targets. You can configure up to 64 VPN targets for a VPN instance.

You can configure route related attributes for IPv4 VPNs in both VPN instance view and IPv4 VPN view. Those configured in IPv4 VPN view take precedence.

You can define the maximum number of routes for a VPN instance to support, preventing too many routes from being redistributed into the PE. The maximum number of routes supported by a PE varies by device.

Create the routing policy you want to associate with a VPN instance. If you do not create a routing policy first, the default routing policy is used.

Configuring a tunneling policy of a VPN instance

Configuring a tunneling policy

When multiple tunnels exist in a MPLS L3VPN network, you can configure a tunneling policy to specify the type and number of tunnels to be used by using **tunnel select-seq** or **preferred-path**.

With **tunnel select-seq**, you can specify the tunnel selection preference order and the number of tunnels for load balancing.

With **preferred-path**, you can configure preferred tunnels that each corresponds to a tunnel interface.

After a tunneling policy is applied on a PE, the PE selects tunnels in this order:

- The PE matches the peer PE address against the destination addresses of preferred tunnels, starting from the tunnel with the smallest number. If no match is found, the local PE selects tunnels as configured by **tunnel select-seq** or the default tunneling policy.
- If a matching tunnel is found and the tunnel is available, the local PE stops matching other tunnels and forwards the traffic to the specified tunnel interface.

If the matching tunnel is unavailable (for example, the tunnel is down or the tunnel's ACL does not permit the traffic) and is not specified with the **disable-fallback** keyword, the local PE continues to match other preferred tunnels; if the tunnel is specified with the **disable-fallback** keyword, the local PE stops matching and tunnel selection fails.

To configure a tunneling policy:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a tunneling policy and enter tunneling policy view.	**tunnel-policy** *tunnel-policy-name*	Required.
3. Configure a preferred tunnel and specify a tunnel interface for it.	**preferred-path** *number* **interface tunnel** *tunnel-number* [**disable-fallback**]	Optional. Not configured by default.
4. Specify the priorities of tunnels and the number of tunnels for load balancing.	**tunnel select-seq** { **cr-lsp** | **lsp** } * **load-balance-number** *number*	Optional. By default, only one tunnel is selected (no load balancing) in this order: LSP tunnel, CR-LSP tunnel.

In a tunneling policy, you can configure up to 64 preferred tunnels.

The tunnel interfaces specified for the preferred tunnels can have the same destination address and the tunnel encapsulation type must be MPLS TE.

When you configure tunnel priorities using **tunnel select-seq**, a tunnel type closer to the **select-seq** keyword has a higher priority. For example, with **tunnel select-seq lsp cr-lsp load-balance-number 1**

configured, VPN uses a CR-LSP tunnel if no LSP exists. Once an LSP is created, the LSP tunnel is used instead.

If you specify more than one tunnel type and the number of tunnels of a type is less than the specified number of tunnels for load balance, tunnels of different types are used for load balance.

Associating a tunneling policy with the VPN instance

To associate a tunneling policy with the VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required.
3. Associate a tunneling policy with the VPN instance.	**tnl-policy** *tunnel-policy-name*	Required. By default, only one tunnel is selected (no load balancing) in this order: LSP tunnel, CR-LSP tunnel.

Create the tunneling policy before associating it with the VPN instance. Otherwise, the default policy is used.

Configuring an LDP instance

LDP instances are for carrier's carrier networking applications of MPLS L3VPN.

This task is to configure the LDP capability for an existing VPN instance, create the LDP instance, and configure LDP parameters for the VPN instance in MPLS LDP VPN instance view.

Prerequisites

Before you configure an LDP instance, complete the following tasks:

- Configure a VPN instance
- Configure MPLS basic capability
- Configure MPLS LDP capability

Procedure

CAUTION:

- Except the command for LDP GR, all commands available in MPLS LDP view can be configured in MPLS LDP VPN instance view. For more information about MPLS LDP, see "MPLS basics configuration."
- Configurations in MPLS LDP VPN instance view affect only the LDP-enabled interface bound to the VPN instance, but configurations in MPLS LDP view do not affect interfaces bound to VPN instances. When configuring the transport address of an LDP instance, you must use the IP address of the interface bound to the VPN instance.
- By default, LDP adjacencies on a private network are established by using addresses of the LDP-enabled interfaces, and those on the public network are established by using the LDP LSR ID.

To configure an LDP instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—

Step	Command	Remarks
2. Enable LDP capability for a VPN instance and enter MPLS LDP VPN instance view.	**mpls ldp vpn-instance** *vpn-instance-name*	Required. Not enabled by default.
3. Configure the LDP parameters except LDP GR for the VPN instance.	For more information, see "MPLS basics configuration."	Optional.

Configuring basic MPLS L3VPN

This section describes how to configure a simple MPLS L3VPN, where only one carrier is involved, the MPLS backbone is not inter-AS, and none of the PEs or CEs functions as a PE and a CE at the same time.

Some special MPLS L3VPN networking scenarios such as HoVPN, multi-role host, and inter-AS VPN require additional configurations.

In configuring MPLS L3VPN, the key task is to manage the advertisement of VPN routes on the MPLS backbone and includes the management of route advertisement between PEs and CEs and that between PEs.

As for the route exchange between a PE and a CE, you can configure static routes, RIP instances, OSPF instances, IS-IS instances, EBGP instances, or IBGP instances, depending on the networking situations. MP-IBGP is adopted between PEs.

Prerequisites

Before you configure basic MPLS L3VPN, complete the following tasks:

- Configure an IGP for the MPLS backbone (PEs and Ps) to achieve IP connectivity
- Configure MPLS basic capability for the MPLS backbone (PEs and Ps)
- Configure MPLS LDP for the MPLS backbone (PEs and Ps) so that LDP LSPs can be established
- Configure the IP addresses for the CE interfaces connected to the PEs

Configuring a VPN instance

To configure a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a VPN instance and enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required. No VPN instance exists by default.
3. Specify a reserved VLAN for the VPN instance.	**reserve-vlan** *vlan-id*	Required.
4. Configure an RD for the VPN instance.	**route-distinguisher** *route-distinguisher*	Required.
5. Associate the current VPN instance with one or more VPN targets.	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	Required.
6. Return to system view.	**quit**	—

Step	Command	Remarks
7. Enter interface view.	**interface** *interface-type interface-number*	—
8. Associate the current interface with the VPN instance.	**ip binding vpn-instance** *vpn-instance-name*	Required. By default, an interface is associated with no VPN instance.

Configuring PE-CE route exchange

PE-CE route exchange can be implemented through static routes, RIP, OSPF, IS-IS, EBGP, and IBGP. You may choose one as needed.

Configuring PE-CE route exchange through static routes

To configure PE-CE route exchange through static routes:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Configure static routes for a specified VPN instance.	**ip route-static** *dest-address* { *mask* | *mask-length* } { *gateway-address* | *interface-type interface-number* [*gateway-address*] | **vpn-instance** *d-vpn-instance-name gateway-address* } [**preference** *preference-value*] [**tag** *tag-value*] [**description** *description-text*] **ip route-static vpn-instance** *s-vpn-instance-name&<1-5> dest-address* { *mask* | *mask-length* } { *gateway-address* [**public**] | *interface-type interface-number* [*gateway-address*] | **vpn-instance** *d-vpn-instance-name gateway-address* } [**preference** *preference-value*] [**tag** *tag-value*] [**description** *description-text*]	Required. Perform this configuration on PEs. On CEs, configure normal static routes.

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

Configuring PE-CE route exchange through RIP

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 PE-CE route exchange through RIP:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a RIP process for a VPN instance and enter RIP view.	**rip** [*process-id*] **vpn-instance** *vpn-instance-name*	Required. Perform this configuration on PEs. On CEs, create a normal RIP process.
3. Enable RIP on the interface attached to the specified network.	**network** *network-address*	Required. By default, RIP is disabled on an interface.

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

Configuring PE-CE route exchange through OSPF

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 the router ID when starting a process or to configure the 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 PE-CE route exchange through OSPF:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create an OSPF process for a VPN instance and enter the OSPF view.	**ospf** [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*]	Required. Perform the configurations on PEs. On CEs, create a normal OSPF process.
3. Configure the OSPF domain ID.	**domain-id** *domain-id* [**secondary**]	Optional. 0 by default.
4. Configure the type codes of OSPF extended community attributes.	**ext-community-type** { **domain-id** *type-code1* | **router-id** *type-code2* | **route-type** *type-code3* }	Optional. The defaults are as follows: 0x0005 for Domain ID, 0x0107 for Router ID, and 0x0306 for Route Type. Configure this command on a PE.

Deleting a VPN instance deletes all related OSPF processes at the same time.

An OSPF process can be configured with only one domain ID. Domain IDs of different OSPF processes are independent of each other.

All OSPF processes of a VPN must be configured with the same domain ID for routes to be correctly advertised. OSPF processes on PEs in different VPNs can be configured with domain IDs as desired.

The domain ID of an OSPF process is included in the routes generated by the process. When an OSPF route is injected into BGP, the OSPF domain ID is included in the BGP VPN route and delivered as a BGP extended community attribute.

After configuring an OSPF instance, you must start OSPF by using the same method for starting a common OSPF process.

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

Configuring PE-CE route exchange through IS-IS

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 PE-CE route exchange through IS-IS:

Step	Command	Remarks
1. Enter system view.	**system-view**	—

Step	Command	Remarks
2. Create an IS-IS process for a VPN instance and enter IS-IS view.	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	Required.

After configuring an IS-IS process for a VPN instance, you must start IS-IS by using the same method for starting a common IS-IS process.

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

Configuring PE-CE route exchange through EBGP

1. On a PE

To configure PE-CE route exchange through EBGP on a PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Enter BGP VPN instance view.	**ipv4-family vpn-instance** *vpn-instance-name*	Required.
4. Configure the CE as the VPN EBGP peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
5. Inject the routes of the local CEs.	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	Required. A PE needs to inject the routes of the local CEs into its VPN routing table so that it can advertise them to the peer PE.
6. Configure BGP to filter routes to be advertised.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct** | **isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	Optional. By default, BGP does not filter routes to be advertised.
7. Configure BGP to filter received routes.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	Optional. By default, BGP does not filter received routes.
8. Allow the local AS number to appear in the AS_PATH attribute of a received route and set the maximum number of repetitions.	**peer** { *group-name* | *ip-address* } **allow-as-loop** [*number*]	Optional. For the hub and spoke networking scheme

Normally, BGP detects routing loops by AS number. In the hub and spoke networking scheme, however, with EBGP running between PE and CE, the routing information the PE advertises to a CE carries the number of the AS where the PE resides. Therefore, the route updates that the PE receives from the CE also include the number of the AS where the PE resides. This causes the PE unable to receive the route updates. In this case, routing loops must be allowed.

2. On a CE

To configure PE-CE route exchange through EBGP on a CE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Configure the PE as the peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
4. Configure the route redistribution and advertisement behavior.	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	Optional. A CE needs to advertise its routes to the connected PE so that the PE can advertise them to the peer CE.

Exchange of BGP routes for a VPN instance is the same as that of ordinary BGP routes.

The BGP configuration task in BGP-VPN instance view is the same as that in BGP view. For more information, see *Layer 3—IP Routing Configuration Guide*.

For information about BGP peer and peer group configuration, see *Layer 3—IP Routing Configuration Guide*. This chapter does not differentiate between peer and peer group.

Configuring PE-CE route exchange through IBGP

IBGP can be used between PE and CE devices in only common MPLS L3VPN networking. In Extranet, inter-AS VPN, carrier's carrier, nested VPN, and HoVPN networking, you cannot use IBGP between PE and CE devices.

1. On a PE

To configure PE-CE route exchange through IBGP on a PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Enter BGP VPN instance view.	**ipv4-family vpn-instance** *vpn-instance-name*	Required.
4. Configure the CE as the VPN IBGP peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
5. Configure the system to be the RR and specify the CE as the client of the RR.	**peer** { *group-name* | *ip-address* } **reflect-client**	Optional. By default, no RR or RR client is configured.
6. Enable route reflection between clients.	**reflect between-clients**	Optional. Enabled by default.
7. Configure the cluster ID for the RR.	**reflector cluster-id** { *cluster-id* | *ip-address* }	Optional. Router ID of an RR in the cluster by default.
8. Configure BGP to filter routes to be advertised.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct** | **isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	Optional. By default, BGP does not filter routes to be advertised.

Step	Command	Remarks
9. Configure BGP to filter received routes.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	Optional. By default, BGP does not filter received routes.

By default, a PE does not advertise routes learned from IBGP peer CEs to IBGP peers, including VPNv4 IBGP peers. Only when you configure an IBGP peer CE as a client of the RR, does the PE advertise routes learned from it to other IBGP peers.

You can execute the reflect between-clients command and the reflector cluster-id command in multiple views, such as BGP-VPN instance view and BGP-VPNv4 subaddress family view. The two commands take effect for only the RR in the view where they are executed. For RRs in other views, they do not take effect.

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 of the route.

2. On a CE

To configure PE-CE route exchange through IBGP on a CE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Configure the PE as the IBGP peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
4. Configure the route redistribution and advertisement behavior.	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	Optional. A CE needs to advertise its routes to the connected PE so that the PE can advertise them to the peer CE.

Exchange of BGP routes of a VPN instance is the same as that of ordinary BGP routes.

The BGP configuration task in BGP VPN instance view is the same as that in BGP view. For more information, see *Layer 3—IP Routing Configuration Guide*.

For information about BGP peer and BGP peer group configuration, see *Layer 3—IP Routing Configuration Guide*. This chapter does not differentiate between peer and peer group.

Configuring PE-PE route exchange

To configure PE-PE route exchange:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	Required.
3. Configure the remote PE as the peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
4. Specify the source interface for route updates.	**peer** { *group-name* | *ip-address* } **connect-interface** *interface-type interface-number*	Required. By default, BGP uses the source interface of the optimal route update packet.

Step	Command	Remarks
5. Enter BGP-VPNv4 subaddress family view.	**ipv4-family vpnv4** [**unicast**]	Required.
6. Enable the exchange of BGP-VPNv4 routing information with the specified peer.	**peer** { *group-name* | *ip-address* } **enable**	Required. By default, BGP peers only exchange IPv4 routing information.

Configuring routing features for BGP VPNv4 subaddress family

With BGP VPNv4 subaddress family, there are a variety of routing features that are the same as those for BGP IPv4 unicast routing. You can select any of the features as required.

Configuring common routing features for all types of subaddress families

For VPN applications, BGP address families include BGP VPN-IPv4 address family, BGP-L2VPN address family, and VPLS address family. Every command in the following table has the same function on BGP routes for each type of the address families and only takes effect for the BGP routes in the address family view where the command is executed.

To configure common routing features for all types of subaddress families:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	Required.
3. Configure the remote PE as the peer.	**peer** *ip-address* **as-number** *as-number*	Required.
4. Specify the interface for TCP connection.	**peer** *ip-address* **connect-interface** *interface-type interface-number*	Required.
5. Enter address family view.	**ipv4-family vpnv4** **l2vpn-family** **vpls-family**	Required. Use one of the commands as needed.
6. Allow the local AS number to appear in the AS_PATH attribute of a received route and set the maximum number of repetitions.	**peer** { *group-name* | *ip-address* } **allow-as-loop** [*number*]	Optional.
7. Enable a peer or peer group for an address family and enable the exchange of BGP routing information of the address family.	**peer** { *group-name* | *ip-address* } **enable**	Required. By default, only IPv4 routing information is exchanged between BGP peers.
8. Add a peer into an existing peer group.	**peer** *ip-address* **group** *group-name*	Optional.

Step	Command	Remarks
9. Configure the system to use the local address as the next hop of a route to be advertised to a specified peer or peer group.	**peer** { *group-name* | *ip-address* } **next-hop-local**	Optional. By default, the system uses the local address as the next hop of a route to be advertised to an EBGP peer. In the inter-AS VPN option C solution, you must configure **peer** { *group-name* | *ip-address* } **next-hop-invariable** on the RR for multi-hop EBGP neighbors and reflector clients to make sure that the next hop of a VPN route is not changed.
10. Configure the system to be the RR and set a peer or peer group as the client of the RR.	**peer** { *group-name* | *ip-address* } **reflect-client**	Optional. By default, no RR or RR client is configured.
11. Enable the ORF capability for a BGP peer/peer group.	**peer** { *group-name* | *ip-address* } **capability-advertise orf ip-prefix** { **both** | **receive** | **send** }	Optional. By default, the ORF capability is disabled on a BGP peer or peer group.
12. Enable VPN target filtering for received VPNv4 routes.	**policy vpn-target**	Optional. Enabled by default
13. Enable route reflection between clients.	**reflect between-clients**	Optional. Enabled by default
14. Specify the cluster ID of the RR.	**reflector cluster-id** { *cluster-id* | *ip-address* }	Optional. Router ID of an RR in the cluster by default.
15. Create an RR reflection policy.	**rr-filter** *extended-community-list-number*	Optional.

For information about BGP-L2VPN address family and VPLS address family, see *MPLS Command Reference*.

Configuring specific routing features for BGP-VPNv4 subaddress family

To configure specific routing features for BGP-VPNv4 subaddress family:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Configure the remote PE as the peer.	**peer** *ip-address* **as-number** *as-number*	Required.
4. Specify the interface for TCP connection.	**peer** *ip-address* **connect-interface** *interface-type interface-number*	Required.
5. Enter BGP-VPNv4 subaddress family view.	**ipv4-family vpnv4**	—
6. Set the default value of the local preference.	**default local-preference** *value*	Optional. 100 by default.

Step	Command	Remarks
7. Set the default system metric.	**default med** *med-value*	Optional. 0 by default.
8. Specify to filter all or certain types of routes to be advertised.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct** | **isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	Optional. By default, BGP does not filter routes to be advertised.
9. Specify to filter received routes.	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	Optional. By default, BGP does not filter received routes.
10. Specify to advertise community attributes to a peer or peer group.	**peer** { *group-name* | *ip-address* } **advertise-community**	Optional. By default, no community attributes are advertised to any peer or peer group.
11. Specify to filter routes received from or to be advertised to a peer or peer group based on an AS_PATH list.	**peer** { *group-name* | *ip-address* } **as-path-acl** *aspath-filter-number* { **import** | **export** }	Optional. By default, no AS filtering list is applied to a peer or peer group.
12. Specify to advertise all default routes of a VPN instance to a peer or peer group.	**peer** { *group-name* | *ip-address* } **default-route-advertise vpn-instance** *vpn-instance-name*	Optional. By default, no default route is advertised to a peer or peer group.
13. Apply a filtering policy to a peer or peer group.	**peer** { *group-name* | *ip-address* } **filter-policy** *acl-number* { **export** | **import** }	Optional. By default, no filtering policy is applied to a peer or peer group.
14. Apply a route filtering policy based on IP prefix list to a peer or peer group.	**peer** { *group-name* | *ip-address* } **ip-prefix** *prefix-name* { **export** | **import** }	Optional. By default, no route filtering policy based on IP prefix list is applied to a peer or peer group.
15. Specify not to change the next hop of a route when advertising it to an EBGP peer.	**peer** { *group-name* | *ip-address* } **next-hop-invariable**	Optional. By default, a device uses its address as the next hop when advertising a route to its EBGP peer.
16. Specify the preference value for the routes received from the peer/peer group.	**peer** { *group-name* | *ip-address* } **preferred-value** *value*	Optional. 0 by default.
17. Make BGP updates to be sent carry no private AS numbers.	**peer** { *group-name* | *ip-address* } **public-as-only**	Optional. By default, a BGP update carries private AS numbers.
18. Apply a routing policy to a peer or peer group.	**peer** { *group-name* | *ip-address* } **route-policy** *route-policy-name* { **export** | **import** }	Optional. By default, no routing policy is applied to a peer or peer group.

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

Configuring inter-AS VPN

If the MPLS backbone on which the VPN routes rely spans multiple ASs, you must configure inter-AS VPN.

There are three inter-AS VPN solutions. You can choose them as required.

Prerequisites

Before you configure inter-AS VPN, complete the following tasks:

- Configure an IGP for the MPLS backbones in each AS to implement IP connectivity of the backbones in the AS
- Configure basic MPLS capabilities for the MPLS backbones of each AS
- Configure MPLS LDP for the MPLS backbones so that LDP LSPs can be established
- Configure basic MPLS L3VPN for each AS

When configuring basic MPLS L3VPN for each AS, specific configurations may be required on PEs or ASBR-PEs. This depends on the inter-AS VPN solution selected.

Configuring inter-AS VPN option A

Inter-AS VPN option A applies to scenarios where the number of VPNs and that of VPN routes on the PEs are relatively small. It is simple to implement.

To configure inter-AS VPN option A:

- Configure basic MPLS L3VPN on each AS.
- Configure each ASBR, taking the peer ASBR PE as its CE.

In other words, configure VPN instances on PEs and ASBR PEs respectively. The VPN instances on PEs are used to allow CEs to access the network, and those on ASBR-PEs are used to access the peer ASBR-PEs.

See "Configuring basic MPLS L3VPN."

In the inter-AS VPN option A solution, for the same VPN, the VPN 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

Configuring inter-AS VPN option B

To configure inter-AS VPN option B on ASBR PEs:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view for the interface connecting to the remote ASBR-PE.	**interface** *interface-type interface-number*	—
3. Configure the IP address of the interface.	**ip address** *ip-address* { *mask* | *mask-length* }	Required.
4. Return to system view.	**quit**	—
5. Enter BGP view.	**bgp** *as-number*	—

Step	Command	Remarks
6. Enter BGP-VPNv4 subaddress family view.	**ipv4-family vpnv4** [**unicast**]	—
7. Disable VPN target filtering for VPNv4 routes.	**undo policy vpn-target**	Required. By default, PE performs VPN target filtering of the received VPNv4 routes. The routes surviving the filtering is added to the routing table, and the others are discarded.

In the inter-AS VPN option B solution, the ASBR PEs must maintain all VPNv4 routing information and advertise the information to peer ASBR PEs. In this case, the ASBR PEs must receive all VPNv4 routing information without performing VPN target filtering.

In the inter-AS VPN option B solution, for the same VPN, the VPN targets for the VPN instances on the PEs in different ASs must match.

Configuring inter-AS VPN option C

Configuring the PEs

CAUTION:

For inter-AS VPN option B, two configuration methods are available:

- Do not change the next hop on an ASBR. With this method, you still must configure MPLS LDP between ASBRs.
- Change the next hop on an ASBR. With this method, MPLS LDP is not required between ASBRs.

Only the second method is supported. Therefore, MP-EBGP routes get their next hops changed by default before being redistributed to MP-IBGP. On conventional BGP, however, EBGP routes to be advertised to IBGP do not have their next hops changed by default. If the next hops must be changed to the local addresses, you can configure **peer** { *ip-address* | *group-name* } **next-hop-local**. For information about the command, see *Layer 3—IP Routing Command Reference*.

You must establish ordinary IBGP peer relationship between PEs and ASBR PEs in an AS and 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 VPN option C:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view	**bgp** *as-number*	—
3. Configure the ASBR PE in the same AS as the IBGP peer	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
4. Enable the PE to exchange labeled IPv4 routes with the ASBR PE in the same AS	**peer** { *group-name* | *ip-address* } **label-route-capability**	Required. By default, the device does not advertise labeled routes to the IPv4 peer/peer group.

Step	Command	Remarks
5. Configure the PE of another AS as the EBGP peer	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
6. Enter BGP-VPNv4 subaddress family view	**ipv4-family vpnv4** [**unicast**]	—
7. Enable the PE to exchange BGP VPNv4 routing information with the peer	**peer** { *group-name* | *ip-address* } **enable**	Required.
8. Configure the PE not to change the next hop of a route when advertising it to the EBGP peer	**peer** { *group-name* | *ip-address* } **next-hop-invariable**	Optional. Required. only when RRs are used to advertise VPNv4 routes, where the next hop of a route advertised between RRs cannot be changed.

Configuring the ASBR PEs

In the inter-AS VPN option C solution, an inter-AS VPN LSP is required, and the routes advertised between the relevant PEs and ASBRs must carry MPLS label information.

An ASBR-PE establishes common IBGP peer relationship with PEs in the same AS, and common EBGP peer relationship with the peer ASBR PE. All of them exchange labeled IPv4 routes.

The public 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 itself. This capability is implemented through BGP extended attributes and requires that the BGP peers can handle labeled IPv4 routes.

To configure an ASBR PE for inter-AS VPN option C:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Configure each PE in the same AS as the IBGP peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
4. Enable the ASBR PE to exchange labeled IPv4 routes with the PEs in the same AS.	**peer** { *group-name* | *ip-address* } **label-route-capability**	Required. By default, the device does not advertise labeled routes to the IPv4 peer/peer group.
5. Configure the ASBR PE to change the next hop to itself when advertising routes to PEs in the same AS.	**peer** { *group-name* | *ip-address* } **next-hop-local**	Required. By default, a BGP speaker does not use its address as the next hop when advertising a route to its IBGP peer/peer group.
6. Configure the remote ASBR PE as the EBGP peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
7. Enable the ASBR PE to exchange labeled IPv4 routes with the peer ASBR PE.	**peer** { *group-name* | *ip-address* } **label-route-capability**	Required. By default, the device does not advertise labeled routes to the IPv4 peer.

Step	Command	Remarks
8. Apply a routing policy to the routes advertised by peer ASBR PE.	**peer** { *group-name* | *ip-address* } **route-policy** *route-policy-name* **export**	Required. By default, no routing policy is applied to a peer or peer group.

Configuring the routing policy

After you configure and apply a routing policy on an ASBR PE, it:

- Assigns MPLS labels to the routes received from the PEs in the same AS before advertising them to the peer ASBR PE.
- Assigns new MPLS labels to the labeled IPv4 routes to be advertised to the PEs in the same AS.

Which IPv4 routes are to be assigned with MPLS labels depends on the routing policy. Only routes that satisfy the criteria are assigned with labels. All other routes are still common IPv4 routes.

To configure a routing policy for inter-AS VPN option C on an ASBR PE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter routing policy view.	**route-policy** *policy-name* **permit node** *seq-number*	Required.
3. Configure the device to match IPv4 routes with labels.	**if-match mpls-label**	Required.
4. Configure the device to assign labels to IPv4 routes.	**apply mpls-label**	Required. By default, an IPv4 route does not carry any label.

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

Configuring nested VPN

For a network with many VPNs, if you want to implement layered management of VPNs and to conceal the deployment of internal VPNs, nested VPN is a good solution. By using nested VPN, you can implement layered management of internal VPNs easily with a low cost and simple management operation.

Prerequisites

Before configuring nested VPN, configure the basic MPLS L3VPN capability. See "Configuring basic MPLS L3VPN."

Configuring nested VPN

To configure nested VPN:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—

Step	Command	Remarks
3. Enter BGP VPN instance view.	**ipv4-family vpn-instance** *vpn-instance-name*	—
4. Configure a CE peer or peer group.	**peer** { *group-name* | *peer-address* } **as-number** *number*	Required.
5. Return to BGP view.	**quit**	—
6. Enter BGP-VPNv4 subaddress family view.	**ipv4-family vpnv4**	—
7. Enable nested VPN.	**nesting-vpn**	Required. Disabled by default.
8. Activate a nested VPN peer or peer group, and enable the BGP-VPNv4 route exchange capability.	**peer** { *group-name* | *peer-address* } **vpn-instance** *vpn-instance-name* **enable**	Required. By default, only IPv4 routes and no BGP-VPNv4 routes can be exchanged between nested VPN peers/peer groups.
9. Add a peer to the nested VPN peer group.	**peer** *peer-address* **vpn-instance** *vpn-instance-name* **group** *group-name*	Optional. By default, a peer is not in any nested VPN peer group.
10. Specify to apply a routing policy to routes received from a nested VPN peer or peer group.	**peer** { *group-name* | *peer-address* } **vpn-instance** *vpn-instance-name* **route-policy** *route-policy-name* **import**	Optional. By default, no routing policy is applied to routes received from a nested VPN peer or peer group.

The address ranges for sub-VPNs of a user VPN cannot overlap.

Do not give nested VPN peers addresses that public network peers use.

Before specifying a nested VPN peer or peer group, be sure to configure the corresponding CE peer or peer group in BGP VPN instance view.

Nested VPN does not support multi-hop EBGP networking. A service provider PE and its peer must use the addresses of the directly connected interfaces to establish neighbor relationship.

On some devices, if a CE of a sub-VPN is directly connected to a service provider's PE, policy routing must be configured on the PE to allow mutual access between the sub-VPN and the VPN on the backbone.

Configuring HoVPN

For hierarchical VPNs, you can adopt HoVPN to reduce the performance requirements for PEs.

Prerequisites

Before you configure HoVPN, complete all basic MPLS L3VPN configurations.

Configuring HoVPNs

To configure HoVPN:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Enter BGP-VPNv4 subaddress family view.	**ipv4-family vpnv4**	Required.
4. Enable the exchange of BGP-VPNv4 routing information with a peer.	**peer** { *group-name* | *ip-address* } **enable**	Required.
5. Specify a BGP peer or peer group as the UPE.	**peer** { *group-name* | *ip-address* } **upe**	Required.
6. Specify to advertise default routes of a VPN instance to a UPE.	**peer** { *group-name* | *ip-address* } **default-route-advertise vpn-instance** *vpn-instance-name*	Required. Configure either command.
7. Specify to advertise routes permitted by a specified routing policy to a UPE.	**peer** { *group-name* | *ip-address* } **upe route-policy** *route-policy-name* **export**	By default, BGP does not advertise default routes to a VPNv4 peer.

With **peer default-route-advertise vpn-instance** configured, the SPE always advertises a default route using the local address as the next hop address to the UPE, regardless of whether the default route is present in the local routing table or not.

The default routes of a VPN instance can be advertised to only a BGP peer or peer group that is UPE.

Do not configure both **peer default-route-advertise vpn-instance** and **peer upe route-policy** 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.

Configuring an OSPF sham link

The sham link is considered an OSPF intra-area route. It is used to ensure that the VPN traffic is transmitted over the backbone instead of the backdoor link between two CEs.

The source and destination addresses of the sham link must be loopback interface addresses with 32-bit masks. Besides, the loopback interfaces must be bound to the VPN instances and be advertised through BGP.

Prerequisites

Before you configure an OSPF sham link, complete the following tasks:

- Configure basic MPLS L3VPN (OSPF is used between PE and CE devices)
- Configure OSPF in the LAN where CEs reside

Configuring a loopback interface

To configure a loopback interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a loopback interface and enter loopback interface view.	**interface loopback** *interface-number*	Required.
3. Bind the loopback interface to VPN instance.	**ip binding vpn-instance** *vpn-instance-name*	Required. By default, an interface is associated with no VPN instance.
4. Configure the address of the loopback interface.	**ip address** *ip-address* { *mask* | *mask-length* }	Required.

Redistributing the loopback interface route and OSPF routes into BGP

To redistribute the loopback interface route and OSPF routes into BGP:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	Required.
3. Enter BGP VPN instance view.	**ipv4-family vpn-instance** *vpn-instance-name*	Required.
4. Redistribute direct routes into BGP (to redistribute the loopback interface route into BGP).	**import-route direct** [**med** *med-value* | **route-policy** *route-policy-name*] *	Required.
5. Redistribute OSPF VPN routes.	**import-route ospf** [{ *process-id* | **all-processes** } [**allow-direct** | **med** *med-value* | **route-policy** *route-policy-name*] *]	Required.

Creating a sham link

To create a sham link:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter OSPF view.	**ospf** [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*] *	—
3. Configure the route tag.	**route-tag** *tag-value*	Required.
4. Enter OSPF area view.	**area** *area-id*	Required.
5. Configure a sham link.	**sham-link** *source-ip-address destination-ip-address* [**cost** *cost* | **dead** *dead-interval* | **hello** *hello-interval* | **retransmit** *retrans-interval* | **trans-delay** *delay* | **simple** [**cipher** | **plain**] *password* | { **md5** | **hmac-md5** } *key-id* [**cipher** | **plain**] *password*]*	Required. By default, no sham link is configured.

If you start OSPF but do not configure the router ID, the system automatically elects one. However, the same election rules produce the same router ID. Therefore, you should configure the router ID when starting an OSPF process. For the election rules, see *Layer 3—IP Routing Configuration Guide*.

If you configure multiple OSPF VPN instances but do not configure the route tag, the system automatically creates one based on the AS number configured. If you do not configure BGP, the tag is 0. However, the same calculation rule produces the same tag, and hence the same tag is created for multiple OSPF VPN instances on the same PE or PEs with the same AS number. Therefore, HP recommends configuring different tags for different OSPF VPN instance.

Configuring BGP AS number substitution

Prerequisites

Before you configure BGP AS number substitution, complete the following tasks:

- Configure basic MPLS L3VPN
- Configure CEs at different sites to have the same AS number

Procedure

When CEs at different sites have the same AS number, configure the BGP AS number substitution function to avoid route loss.

With the BGP AS number substitution function, when a PE advertises a route to a CE of the specified peer, if an AS number identical to that of the CE exist in the AS_PATH of the route, it is replaced with that of the PE before the route is advertised.

To configure the BGP AS number substitution function:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	Required.
3. Enter BGP VPN instance view.	**ipv4-family vpn-instance** *vpn-instance-name*	Required.
4. Enable the BGP AS number substitution function.	**peer** { *ip-address* | *group-name* } **substitute-as**	Required. Disabled by default.

For information about **peer** { *ip-address* | *group-name* } **substitute-as**, see *Layer 3—IP Routing Command Reference*.

Displaying and maintaining MPLS L3VPN

Resetting BGP connections

When BGP configuration changes, you can use the soft reset function or reset BGP connections to make new configurations take effect. Soft reset requires that BGP peers have route refreshment capability (supporting Route-Refresh messages).

Task	Command	Remarks
Soft reset the BGP connections in a specified VPN instance.	**refresh bgp vpn-instance** *vpn-instance-name* { *ip-address* | **all** | **external** | **group** *group-name* } { **export** | **import** }	Available in user view.
Soft reset the BGP VPNv4 connections.	**refresh bgp vpnv4** { *ip-address* | **all** | **external** | **group** *group-name* | **internal** } { **export** | **import** }	Available in user view.
Reset BGP connections of a VPN instance.	**reset bgp vpn-instance** *vpn-instance-name* { *as-number* | *ip-address* | **all** | **external** | **group** *group-name* }	Available in user view.
Reset BGP VPNv4 connections.	**reset bgp vpnv4** { *as-number* | *ip-address* | **all** | **external** | **internal** | **group** *group-name* }	Available in user view.

Displaying and maintaining MPLS L3VPN

Task	Command	Remarks
Display information about the routing table associated with a VPN instance.	**display ip routing-table vpn-instance** *vpn-instance-name* [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about a specified or all VPN instances.	**display ip vpn-instance** [**instance-name** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about the FIB of a VPN instance.	**display fib vpn-instance** *vpn-instance-name* [**acl** *acl-number* | **ip-prefix** *ip-prefix-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about the FIB of a VPN instance that matches the specified destination IP address.	**display fib vpn-instance** *vpn-instance-name ip-address* [*mask* | *mask-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about labeled routes in the BGP routing table.	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **routing-table label** [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about a specified or all BGP VPNv4 peer group.	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **group** [*group-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about BGP VPNv4 routes injected into a specified or all VPN instances.	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **network** [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display BGP VPNv4 AS path information.	**Display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **paths** [*as-regular-expression* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]	Available in any view

Task	Command	Remarks
.Display information about BGP VPNv4 peers.	**display bgp vpnv4 all peer** [*ip-address* **verbose** | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*] **display bgp vpnv4 vpn-instance** *vpn-instance-name* **peer** [*group-name* **log-info** | *ip-address* { **log-info** | **verbose** } | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display the IP prefix information of the ORF packets received from the specified BGP peer.	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **peer** *ip-address* **received ip-prefix** [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display all BGP VPNv4 routing information.	**display bgp vpnv4 all routing-table** [[*network-address* [{ *mask* | *mask-length* } [**longer-prefixes**]] | **as-path-acl** *as-path-acl-number* | **cidr** | **community** [*aa:nn*]&<1-13> [**no-advertise** | **no-export** | **no-export-subconfed**] * [**whole-match**] | **community-list** { *basic-community-list-number* [**whole-match**] | *adv-community-list-number* }&<1-16> | **different-origin-as** | **peer** *ip-address* { **advertised-routes** | **received-routes** } [**statistic**] | **statistic**] [**|** { **begin** | **exclude** | **include** } *regular-expression*] | **regular-expression** *as-regular-expression*]	Available in any view

Task	Command	Remarks
Display the BGP VPNv4 routing information of a specified RD.	**display bgp vpnv4 route-distinguisher** *route-distinguisher* **routing-table** [[*network-address* [*mask* | *mask-length*] | **as-path-acl** *as-path-acl-number* | **cidr** | **community** [*aa:nn*]&<1-13> [**no-advertise** | **no-export** | **no-export-subconfed**] * [**whole-match**] | **community-list** { *basic-community-list-number* [**whole-match**] | *adv-community-list-number* }&<1-16> | **different-origin-as**] [**|** { **begin** | **exclude** | **include** } *regular-expression*] | **regular-expression** *as-regular-expression*]	Available in any view
Display the BGP VPNv4 routing information of a specified VPN instance.	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **routing-table** [[*network-address* [{ *mask* | *mask-length* } [**longer-prefixes**]] | **as-path-acl** *as-path-acl-number* | **cidr** | **community** [*aa:nn*]&<1-13> [**no-advertise** | **no-export** | **no-export-subconfed**] * [**whole-match**] | **community-list** { *basic-community-list-number* [**whole-match**] | *adv-community-list-number* }&<1-16> | **dampened** | **dampening parameter** | **different-origin-as** | **flap-info** [*network-address* [{ *mask* | *mask-length* } [**longer-match**]] | **as-path-acl** *as-path-acl-number*] | **peer** *ip-address* { **advertised-routes** | **received-routes** } | **statistic**] [**|** { **begin** | **exclude** | **include** } *regular-expression*] | [**flap-info**] **regular-expression** *as-regular-expression*]	Available in any view
Display information about OSPF sham links.	**display ospf** [*process-id*] **sham-link** [**area** *area-id*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about a specified or all tunnel policies.	**display tunnel-policy** { **all** | **policy-name** *tunnel-policy-name* } [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about the specified LDP instance.	**display mpls ldp vpn-instance** *vpn-instance-name* [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Clear the route flap dampening information of a VPN instance.	**reset bgp vpn-instance** *vpn-instance-name* **dampening** [*network-address* [*mask* | *mask-length*]	Available in user view
Clear route flap history information about a BGP peer of a VPN instance.	**reset bgp vpn-instance** *vpn-instance-name* *ip-address* **flap-info** **reset bgp vpn-instance** *vpn-instance-name* **flap-info** [*ip-address* [*mask* | *mask-length*] | **as-path-acl** *as-path-acl-number* | **regexp** *as-path-regexp*]	Available in user view

For commands to display information about a routing table, see *Layer 3—IP Routing Command Reference*.

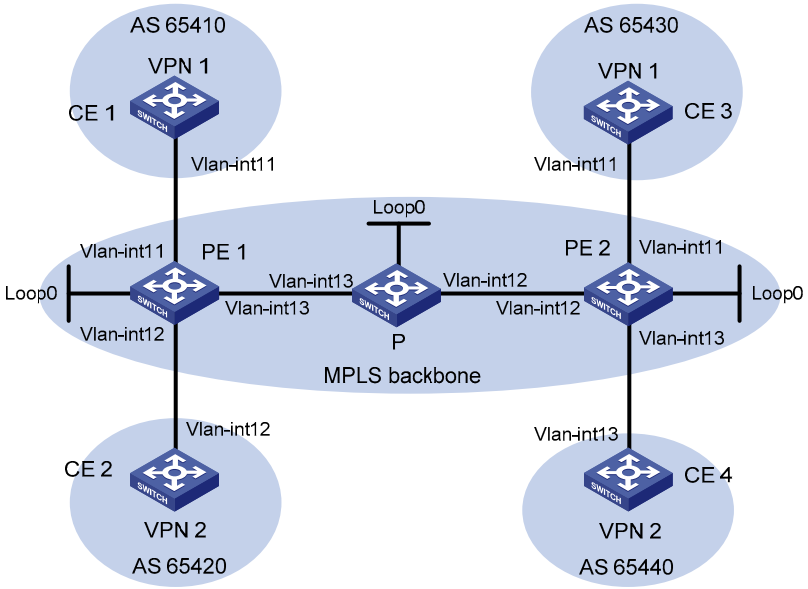
MPLS L3VPN configuration examples

Example for configuring 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 VPN target attributes 111:1. VPN 2 uses VPN target attributes 222:2. Users of different VPNs cannot access each other.
- PEs and the P device support MPLS.

Figure 72 Configure MPLS L3VPNs



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int11	10.1.1.1/24	P	Loop0	2.2.2.9/32
PE 1	Loop0	1.1.1.9/32		Vlan-int12	172.2.1.1/24
	Vlan-int11	10.1.1.2/24		Vlan-int13	172.1.1.2/24
	Vlan-int13	172.1.1.1/24	PE 2	Loop0	3.3.3.9/32
	Vlan-int12	10.2.1.2/24		Vlan-int12	172.2.1.2/24
CE 2	Vlan-int12	10.2.1.1/24		Vlan-int11	10.3.1.2/24
CE 3	Vlan-int11	10.3.1.1/24		Vlan-int13	10.4.1.2/24
CE 4	Vlan-int13	10.4.1.1/24			

Procedure

1. Configure an IGP on the MPLS backbone to implement 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 vlan-interface 13
[PE1-Vlan-interface13] ip address 172.1.1.1 24
[PE1- Vlan-interface13] 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 vlan-interface 13
[P-Vlan-interface13] ip address 172.1.1.2 24
[P- Vlan-interface13] quit
[P] interface vlan-interface 12
[P-Vlan-interface12] ip address 172.2.1.1 24
[P-Vlan-interface12] 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 vlan-interface 12
[PE2-Vlan-interface12] ip address 172.2.1.2 24
[PE2-Vlan-interface12] 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 you complete the configurations, OSPF adjacencies are established between PE 1, P, and PE 2. Issuing **display ospf peer**, you can see that the adjacency status is Full. Issuing **display ip routing-table**, you can see that the PEs have learned the routes to the loopback interfaces of each other. The following uses PE 1 as an example:

```
[PE1] display ip routing-table
Routing Tables: Public
         Destinations : 9        Routes : 9
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         172.1.1.2       Vlan13
3.3.3.9/32        OSPF   10   2         172.1.1.2       Vlan13
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
172.1.1.0/24      Direct 0    0         172.1.1.1       Vlan13
172.1.1.1/32      Direct 0    0         127.0.0.1       InLoop0
172.1.1.2/32      Direct 0    0         172.1.1.2       Vlan13
172.2.1.0/24      OSPF   10   1         172.1.1.2       Vlan13
[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(Vlan-interface13)'s neighbors
 Router ID: 172.1.1.2        Address: 172.1.1.2        GR State: Normal
   State: Full  Mode:Nbr is  Master  Priority: 1
   DR: None   BDR: None   MTU: 1500
   Dead timer due in 38  sec
   Neighbor is up for 00:02:44
   Authentication Sequence: [ 0 ]
   Neighbor state change count: 5
```

2. Configure MPLS basic capability and MPLS LDP on the MPLS backbone to establish LDP LSPs.

Configure PE 1.

```
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
[PE1] interface vlan-interface 13
[PE1-Vlan-interface13] mpls
[PE1-Vlan-interface13] mpls ldp
[PE1-Vlan-interface13] quit
```

Configure the P device.

```
[P] mpls lsr-id 2.2.2.9
[P] mpls
[P-mpls] quit
[P] mpls ldp
[P-mpls-ldp] quit
[P] interface vlan-interface 13
[P-Vlan-interface13] mpls
[P-Vlan-interface13] mpls ldp
[P-Vlan-interface13] quit
[P] interface vlan-interface 12
[P-Vlan-interface12] mpls
```

```
[P-Vlan0interface12] mpls ldp
[P-Vlan-interface12] quit
```

Configure PE 2.

```
[PE2] mpls lsr-id 3.3.3.9
[PE2] mpls
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface vlan-interface 12
[PE2-Vlan-interface12] mpls
[PE2-Vlan-interface12] mpls ldp
[PE2-Vlan-interface12] quit
```

After you complete the configurations, LDP sessions are established between PE 1, P, and PE 2. Issuing **display mpls ldp session**, you can see that the Status field has a value of Operational. Issuing **display mpls ldp lsp**, you can see the LSPs established by LDP. The following uses PE 1 as an example:

```
[PE1] display mpls ldp session
               LDP Session(s) in Public Network
Total number of sessions: 1
------------------------------------------------------------------
 Peer-ID         Status        LAM  SsnRole  FT   MD5  KA-Sent/Rcv
 ------------------------------------------------------------------
 2.2.2.9:0       Operational   DU   Passive  Off  Off  5/5
 ------------------------------------------------------------------
 LAM : Label Advertisement Mode         FT  : Fault Tolerance
[PE1] display mpls ldp lsp
                            LDP LSP Information
 ------------------------------------------------------------------------
 SN  DestAddress/Mask   In/OutLabel  Next-Hop     In/Out-Interface
 ------------------------------------------------------------------------
 1   1.1.1.9/32         3/NULL       127.0.0.1     Vlan-interface13/InLoop0
 2   2.2.2.9/32         NULL/3       172.1.1.2     -------/Vlan-interface13
 3   3.3.3.9/32         NULL/1024    172.1.1.2     -------/Vlan-interface13
------------------------------------------------------------------------
 A '*' before an LSP means the LSP is not established
 A '*' before a Label means the USCB or DSCB is stale
```

3. Configure VPN instances on PEs to allow CEs to 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 vlan-interface 11
[PE1-Vlan-interface11] ip binding vpn-instance vpn1
```

```
[PE1-Vlan-interface11] ip address 10.1.1.2 24
[PE1-Vlan-interface11] quit
[PE1] interface vlan-interface 12
[PE1-Vlan-interface12] ip binding vpn-instance vpn2
[PE1-Vlan-interface12] ip address 10.2.1.2 24
[PE1-Vlan-interface12] 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 vlan-interface 11
[PE2-Vlan-interface11] ip binding vpn-instance vpn1
[PE2-Vlan-interface11] ip address 10.3.1.2 24
[PE2-Vlan-interface11] quit
[PE2] interface vlan-interface 13
[PE2-Vlan-interface13] ip binding vpn-instance vpn2
[PE2-Vlan-interface13] ip address 10.4.1.2 24
[PE2-Vlan-interface13] quit
```

Configure IP addresses for the CEs as required in Figure 72. The detailed configuration steps are omitted.

After completing the configurations, you can issue **display ip vpn-instance** on the PEs to view the configuration of the VPN instance. Use **ping** to test connectivity between the PEs and their attached CEs. The PEs can ping their attached CEs. The following uses 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        2006/08/13 09:32:45
  vpn2                  100:2        2006/08/13 09:42:59
[PE1] ping -vpn-instance vpn1 10.1.1.1
  PING 10.1.1.1: 56  data bytes, press CTRL_C to break
    Reply from 10.1.1.1: bytes=56 Sequence=1 ttl=255 time=56 ms
    Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=255 time=4 ms
    Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=255 time=4 ms
    Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=255 time=52 ms
    Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=255 time=3 ms
  --- 10.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 3/23/56 ms
```

4. Establish EBGP peer relationships between PEs and CEs to allow VPN routes to be redistributed.

Configure CE 1.

```
<CE1> system-view
[CE1] bgp 65410
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] import-route direct
[CE1-bgp] quit
```

The configurations for the other three CEs are similar. The detailed configuration steps are omitted.

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] peer 10.1.1.1 as-number 65410
[PE1-bgp-vpn1] import-route direct
[PE1-bgp-vpn1] quit
[PE1-bgp] ipv4-family vpn-instance vpn2
[PE1-bgp-vpn2] peer 10.2.1.1 as-number 65420
[PE1-bgp-vpn2] import-route direct
[PE1-bgp-vpn2] quit
[PE1-bgp] quit
```

The configurations for PE 2 are similar to those for PE 1. The detailed configuration steps are omitted.

After completing the configurations, if you issue **display bgp vpnv4 vpn-instance peer** on the PEs, you can see that BGP peer relationships have been established between PEs and CEs, and have reached the Established state. The following uses PE 1 and CE 1 as an example:

```
[PE1] display bgp vpnv4 vpn-instance vpn1 peer
 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     11        9     0        1   00:06:37   Established
```

5. Configure MP-IBGP peers 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] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 3.3.3.9 enable
[PE1-bgp-af-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] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 1.1.1.9 enable
[PE2-bgp-af-vpnv4] quit
[PE2-bgp] quit
```

After completing the configurations, if you issue **display bgp peer** or **display bgp vpnv4 all peer** on the PEs, you can see that a BGP peer relationship has been established between the PEs, and has reached the Established state.

```
[PE1] display bgp peer
 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        2        6     0          0  00:00:12 Established
```

6. Verify your configurations

Issuing **display ip routing-table vpn-instance** on the PEs, you can see the routes to the CEs. The following uses PE 1 as an example:

```
[PE1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 3        Routes : 3
Destination/Mask  Proto  Pre  Cost      NextHop         Interface
10.1.1.0/24       Direct 0    0         10.1.1.2        Vlan11
10.1.1.2/32       Direct 0    0         127.0.0.1       InLoop0
10.3.1.0/24       BGP    255  0         3.3.3.9         NULL0
[PE1] display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 3        Routes : 3
Destination/Mask  Proto  Pre  Cost      NextHop         Interface
10.2.1.0/24       Direct 0    0         10.2.1.2        Vlan12
10.2.1.2/32       Direct 0    0         127.0.0.1       InLoop0
10.4.1.0/24       BGP    255  0         3.3.3.9         NULL0
```

CEs of the same VPN can ping each other, whereas those of different VPNs can not. For example, CE 1 can ping CE 3 (10.3.1.1), but cannot ping CE 4 (10.4.1.1):

```
[CE1] ping 10.3.1.1
  PING 10.3.1.1: 56  data bytes, press CTRL_C to break
    Reply from 10.3.1.1: bytes=56 Sequence=1 ttl=253 time=72 ms
    Reply from 10.3.1.1: bytes=56 Sequence=2 ttl=253 time=34 ms
    Reply from 10.3.1.1: bytes=56 Sequence=3 ttl=253 time=50 ms
    Reply from 10.3.1.1: bytes=56 Sequence=4 ttl=253 time=50 ms
    Reply from 10.3.1.1: bytes=56 Sequence=5 ttl=253 time=34 ms
  --- 10.3.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 34/48/72 ms
[CE1] ping 10.4.1.1
  PING 10.4.1.1: 56  data bytes, press CTRL_C to break
    Request time out
    Request time out
    Request time out
    Request time out
    Request time out
```

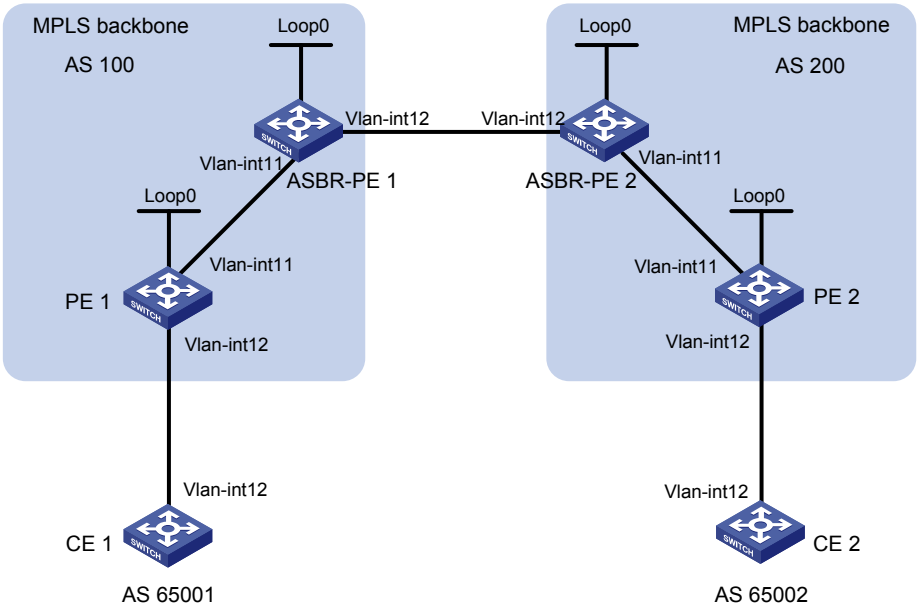
```
--- 10.4.1.1 ping statistics ---
  5 packet(s) transmitted
  0 packet(s) received
  100.00% packet loss
```

Example for configuring inter-AS VPN 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.
- Inter-AS MPLS L3VPN is implemented using option A. The VRF-to-VRF method is used to manage VPN routes.
- The MPLS backbone in each AS runs OSPF.

Figure 73 Configure inter-AS VPN option A



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int12	10.1.1.1/24	CE 2	Vlan-int12	10.2.1.1/24
PE 1	Loop0	1.1.1.9/32	PE 2	Loop0	4.4.4.9/32
	Vlan-int12	10.1.1.2/24		Vlan-int12	10.2.1.2/24
	Vlan-int11	172.1.1.2/24		Vlan-int11	162.1.1.2/24
ASBR-PE 1	Loop0	2.2.2.9/32	ASBR-PE 2	Loop0	3.3.3.9/32
	Vlan-int11	172.1.1.1/24		Vlan-int11	162.1.1.1/24
	Vlan-int12	192.1.1.1/24		Vlan-int12	192.1.1.2/24

Procedure

1. Configure an IGP on the MPLS backbone to implement IP connectivity within the backbone.

This example uses OSPF. The detailed configuration steps are omitted.

The 32-bit loopback interface address used as the LSR ID needs to be advertised by OSPF.

After you complete the configurations, each ASBR PE and the PE in the same AS are able to establish OSPF adjacencies. Issuing **display ospf peer**, you can see that the adjacencies reach the Full state, and that PEs can learn the loopback addresses of each other.

Each ASBR PE and the PE in the same AS are able to ping each other.

2. Configure MPLS basic capability and MPLS LDP on the MPLS backbone to establish LDP LSPs.

Configure MPLS basic capability 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
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
[PE1] interface vlan-interface 11
[PE1-Vlan-interface11] mpls
[PE1-Vlan-interface11] mpls ldp
[PE1-Vlan-interface11] quit
```

Configure MPLS basic capability 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
[ASBR-PE1-mpls] quit
[ASBR-PE1] mpls ldp
[ASBR-PE1-mpls-ldp] quit
[ASBR-PE1] interface vlan-interface 11
[ASBR-PE1-Vlan-interface11] mpls
[ASBR-PE1-Vlan-interface11] mpls ldp
[ASBR-PE1-Vlan-interface11] quit
```

Configure MPLS basic capability 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
[ASBR-PE2-mpls] quit
[ASBR-PE2] mpls ldp
[ASBR-PE2-mpls-ldp] quit
[ASBR-PE2] interface vlan-interface 11
[ASBR-PE2-Vlan-interface11] mpls
[ASBR-PE2-Vlan-interface11] mpls ldp
[ASBR-PE2-Vlan-interface11] quit
```

Configure MPLS basic capability 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
```

```
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface vlan-interface 11
[PE2-Vlan-interface11] mpls
[PE2-Vlan-interface11] mpls ldp
[PE2-Vlan-interface11] quit
```

After you complete the configurations, each PE and the ASBR PE in the same AS are able to establish neighbor relationship. Issuing **display mpls ldp session** on the devices, you can see that the Status field has a value of Operational in the output information.

3. Configure VPN instances on PEs to allow CEs to access.

For the same VPN, the VPN 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 vlan-interface 12
[CE1-Vlan-interface12] ip address 10.1.1.1 24
[CE1-Vlan-interface12] 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 vlan-interface 12
[PE1-Vlan-interface12] ip binding vpn-instance vpn1
[PE1-Vlan-interface12] ip address 10.1.1.2 24
[PE1-Vlan-interface12] quit
```

Configure CE 2.

```
<CE2> system-view
[CE2] interface vlan-interface 12
[CE2-Vlan-interface12] ip address 10.2.1.1 24
[CE2-Vlan-interface12] quit
```

Configure PE 2.

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance] route-distinguisher 200:2
[PE2-vpn-instance] vpn-target 100:1 both
[PE2-vpn-instance] quit
[PE2] interface vlan-interface 12
[PE2-Vlan-interface12] ip binding vpn-instance vpn1
[PE2-Vlan-interface12] ip address 10.2.1.2 24
[PE2-Vlan-interface12] 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-instance-vpn1] route-distinguisher 100:1
```

```
[ASBR-PE1-vpn-instance-vpn1] vpn-target 100:1 both
[ASBR-PE1-vpn-instance-vpn1] quit
[ASBR-PE1] interface vlan-interface 12
[ASBR-PE1-Vlan-interface12] ip binding vpn-instance vpn1
[ASBR-PE1-Vlan-interface12] ip address 192.1.1.1 24
[ASBR-PE1-Vlan-interface12] 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-vpn-vpn1] route-distinguisher 200:1
[ASBR-PE2-vpn-vpn-vpn1] vpn-target 100:1 both
[ASBR-PE2-vpn-vpn-vpn1] quit
[ASBR-PE2] interface vlan-interface 12
[ASBR-PE2-Vlan-interface12] ip binding vpn-instance vpn1
[ASBR-PE2-Vlan-interface12] ip address 192.1.1.2 24
[ASBR-PE2-Vlan-interface12] quit
```

After completing the configurations, you can use **display ip vpn-instance** to view the VPN instance configurations.

The PEs are able to ping the CEs and the ASBR PEs are able to ping each other.

4. Establish EBGP peer relationship between PEs and CEs to allow VPN routes to be redistributed.

Configure CE 1.

```
[CE1] bgp 65001
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] import-route direct
[CE1-bgp] quit
```

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] peer 10.1.1.1 as-number 65001
[PE1-bgp-vpn1] import-route direct
[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] import-route direct
[CE2-bgp] quit
```

Configure PE 2.

```
[PE2] bgp 200
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] peer 10.2.1.1 as-number 65002
[PE2-bgp-vpn1] import-route direct
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

5. Establish MP-IBGP peer relationship between each PE and the ASBR PE in the same AS and 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] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 2.2.2.9 enable
[PE1-bgp-af-vpnv4] peer 2.2.2.9 next-hop-local
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

Configure ASBR PE 1.

```
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] ipv4-family vpn-instance vpn1
[ASBR-PE1-bgp-vpn1] peer 192.1.1.2 as-number 200
[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] ipv4-family vpnv4
[ASBR-PE1-bgp-af-vpnv4] peer 1.1.1.9 enable
[ASBR-PE1-bgp-af-vpnv4] peer 1.1.1.9 next-hop-local
[ASBR-PE1-bgp-af-vpnv4] quit
[ASBR-PE1-bgp] quit
```

Configure ASBR PE 2.

```
[ASBR-PE2] bgp 200
[ASBR-PE2-bgp] ipv4-family vpn-instance vpn1
[ASBR-PE2-bgp-vpn1] peer 192.1.1.1 as-number 100
[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] ipv4-family vpnv4
[ASBR-PE2-bgp-af-vpnv4] peer 4.4.4.9 enable
[ASBR-PE2-bgp-af-vpnv4] peer 4.4.4.9 next-hop-local
[ASBR-PE2-bgp-af-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] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 3.3.3.9 enable
[PE2-bgp-af-vpnv4] peer 3.3.3.9 next-hop-local
[PE2-bgp-af-vpnv4] quit
[PE2-bgp] quit
```

6. Verify your configurations.

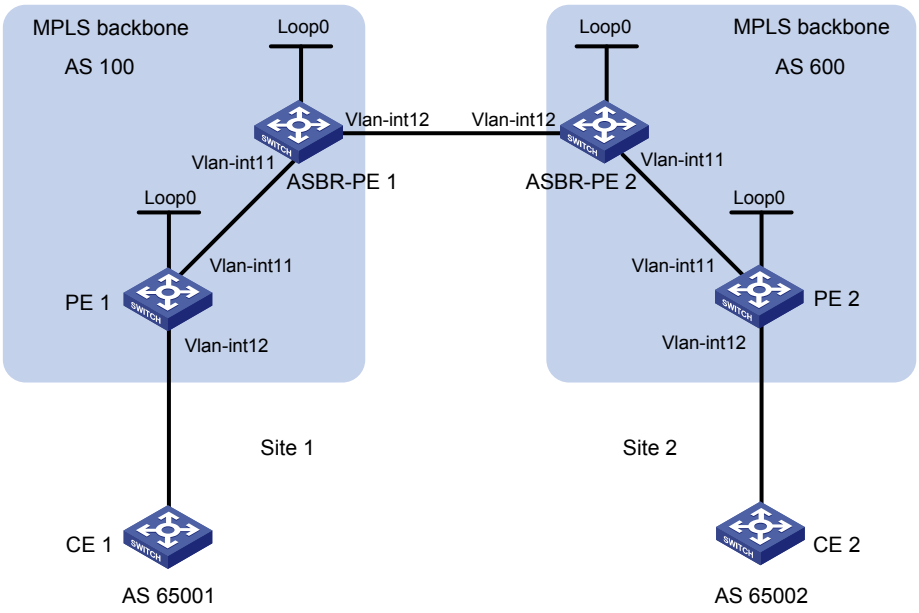
After you complete the configurations, the CEs are able to learn the interface routes from each other and can ping each other.

Example for configuring inter-AS VPN 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 runs IS-IS between them.
- PE 1 and ASBR-PE 1 exchange labeled IPv4 routes by MP-IBGP.
- PE 2 and ASBR-PE 2 exchange labeled IPv4 routes by MP-IBGP.
- ASBR-PE 1 and ASBR-PE 2 exchange labeled IPv4 routes by MP-EBGP.
- ASBRs do not perform VPN target filtering of received VPN-IPv4 routes.

Figure 74 Configure inter-AS VPN option B



Device	Interface	IP address	Device	Interface	IP address
PE 1	Loop0	2.2.2.9/32	PE 2	Loop0	5.5.5.9/32
	Vlan-int12	30.0.0.1/8		Vlan-int12	20.0.0.1/8
	Vlan-int11	1.1.1.2/8		Vlan-int11	9.1.1.2/8
ASBR-PE 1	Loop0	3.3.3.9/32	ASBR-PE 2	Loop0	4.4.4.9/32
	Vlan-int11	1.1.1.1/8		Vlan-int11	9.1.1.1/8
	Vlan-int12	11.0.0.2/8		Vlan-int12	11.0.0.1/8

Procedure

1. Configure PE 1

Run 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, enable MPLS and LDP.

```
[PE1] mpls lsr-id 2.2.2.9
[PE1] mpls
[PE1-mpls] label advertise non-null
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[PE1] interface vlan-interface 11
[PE1-Vlan-interface11] ip address 1.1.1.2 255.0.0.0
[PE1-Vlan-interface11] isis enable 1
[PE1-Vlan-interface11] mpls
[PE1-Vlan-interface11] mpls ldp
[PE1-Vlan-interface11] 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 VPN 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 with CE 1 to the created VPN instance.

```
[PE1] interface vlan-interface 12
[PE1-Vlan-interface12] ip binding vpn-instance vpn1
[PE1-Vlan-interface12] ip address 30.0.0.1 8
[PE1-Vlan-interface12] quit
```

Start 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] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 3.3.3.9 enable
[PE1-bgp-af-vpnv4] quit
```

Redistribute direct routes to the VPN routing table of **vpn1**.

```
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import-route direct
[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 10.222.222.222.222.00
[ASBR-PE1-isis-1] quit
```

Configure LSR ID, enable MPLS and LDP.

```
[ASBR-PE1] mpls lsr-id 3.3.3.9
[ASBR-PE1] mpls
[ASBR-PE1-mpls] label advertise non-null
[ASBR-PE1-mpls] quit
[ASBR-PE1] mpls ldp
[ASBR-PE1-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[ASBR-PE1] interface vlan-interface 11
[ASBR-PE1-Vlan-interface11] ip address 1.1.1.1 255.0.0.0
[ASBR-PE1-Vlan-interface11] isis enable 1
[ASBR-PE1-Vlan-interface11] mpls
[ASBR-PE1-Vlan-interface11] mpls ldp
[ASBR-PE1-Vlan-interface11] quit
```

Configure interface VLAN-interface 12 and enable MPLS on it.

```
[ASBR-PE1] interface vlan-interface 12
[ASBR-PE1-Vlan-interface12] ip address 11.0.0.2 255.0.0.0
[ASBR-PE1-Vlan-interface12] mpls
[ASBR-PE1-Vlan-interface12] 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
```

Start 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 vlan-interface 12
```

Disable VPN target filtering for received VPNv4 routes.

```
[ASBR-PE1-bgp] ipv4-family vpnv4
[ASBR-PE1-bgp-af-vpnv4] undo policy vpn-target
```

Configure both IBGP peer 2.2.2.0 and EBGP peer 11.0.0.1 as VPNv4 peers.

```
[ASBR-PE1-bgp-af-vpnv4] peer 11.0.0.1 enable
[ASBR-PE1-bgp-af-vpnv4] peer 2.2.2.9 enable
[ASBR-PE1-bgp-af-vpnv4] 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 10.222.222.222.222.00
[ASBR-PE2-isis-1] quit
```

Configure LSR ID, enable MPLS and LDP.

```
[ASBR-PE2] mpls lsr-id 4.4.4.9
[ASBR-PE2] mpls
[ASBR-PE2-mpls] label advertise non-null
[ASBR-PE2-mpls] quit
[ASBR-PE2] mpls ldp
[ASBR-PE2-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[ASBR-PE2] interface vlan-interface 11
[ASBR-PE2-Vlan-interface11] ip address 9.1.1.1 255.0.0.0
[ASBR-PE2-Vlan-interface11] isis enable 1
[ASBR-PE2-Vlan-interface11] mpls
[ASBR-PE2-Vlan-interface11] mpls ldp
[ASBR-PE2-Vlan-interface1] quit
```

Configure interface VLAN-interface 12 and enable MPLS on it.

```
[ASBR-PE2] interface vlan-interface 12
[ASBR-PE2-Vlan-interface12] ip address 11.0.0.1 255.0.0.0
[ASBR-PE2-Vlan-interface12] mpls
[ASBR-PE2-Vlan-interface12] 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
```

Start 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 vlan-interface 12
[ASBR-PE2-bgp] peer 5.5.5.9 as-number 600
[ASBR-PE2-bgp] peer 5.5.5.9 connect-interface loopback 0
```

Disable VPN target filtering for received VPNv4 routes.

```
[ASBR-PE2-bgp] ipv4-family vpnv4
[ASBR-PE2-bgp-af-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-af-vpnv4] peer 11.0.0.2 enable
[ASBR-PE2-bgp-af-vpnv4] peer 5.5.5.9 enable
[ASBR-PE2-bgp-af-vpnv4] 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 10.111.111.111.111.00
[PE2-isis-1] quit
```

Configure LSR ID, enable MPLS and LDP.

```
[PE2] mpls lsr-id 5.5.5.9
[PE2] mpls
[PE2-mpls] label advertise non-null
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[PE2] interface vlan-interface 11
[PE2-Vlan-interface11] ip address 9.1.1.2 255.0.0.0
[PE2-Vlan-interface11] isis enable 1
[PE2-Vlan-interface11] mpls
[PE2-Vlan-interface11] mpls ldp
[PE2-Vlan-interface11] 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 VPN 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 2 to the created VPN instance.

```
[PE2] interface vlan-interface 12
[PE2-Vlan-interface12] ip binding vpn-instance vpn1
[PE2-Vlan-interface12] ip address 20.0.0.1 8
[PE2-Vlan-interface12] quit
```

Start 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] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 4.4.4.9 enable
[PE2-bgp-af-vpnv4] quit
```

Redistribute direct routes to the VPN routing table of vpn1.

```
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] import-route direct
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

5. Verify your configurations

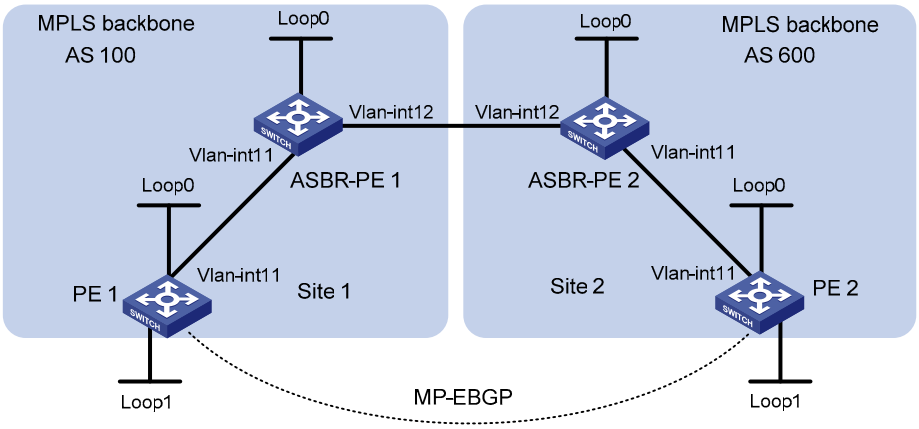
After you complete the configurations, PE 1 and PE 2 are able to ping each other.

Example for configuring inter-AS VPN 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 runs IS-IS between them.
- PE 1 and ASBR-PE 1 exchange labeled IPv4 routes by MP-IBGP.
- PE 2 and ASBR-PE 2 exchange labeled IPv4 routes by MP-IBGP.
- PE 1 and PE 2 are MP-EBGP peers.
- 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 MP-EBGP to exchange labeled IPv4 routes.

Figure 75 Configure inter-AS VPN option C



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
	Vlan-int11	1.1.1.2/8		Vlan-int11	9.1.1.2/8
ASBR-PE 1	Loop0	3.3.3.9/32	ASBR-PE 2	Loop0	4.4.4.9/32
	Vlan-int11	1.1.1.1/8		Vlan-int11	9.1.1.1/8
	Vlan-int12	11.0.0.2/8		Vlan-int12	11.0.0.1/8

Procedure

1. Configure PE 1

Run 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, enable MPLS and LDP.

```
[PE1] mpls lsr-id 2.2.2.9
[PE1] mpls
[PE1-mpls] label advertise non-null
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[PE1] interface vlan-interface 11
[PE1-Vlan-interface11] ip address 1.1.1.2 255.0.0.0
[PE1-Vlan-interface11] isis enable 1
[PE1-Vlan-interface11] mpls
[PE1-Vlan-interface11] mpls ldp
[PE1-Vlan-interface11] 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 VPN 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
```

Configure 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] peer 3.3.3.9 label-route-capability
```

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] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 5.5.5.9 enable
[PE1-bgp-af-vpnv4] quit
```

Redistribute direct routes to the routing table of **vpn1**.

```
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import-route direct
[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 10.222.222.222.222.00
[ASBR-PE1-isis-1] quit
```

Configure LSR ID, enable MPLS and LDP.

```
[ASBR-PE1] mpls lsr-id 3.3.3.9
[ASBR-PE1] mpls
[ASBR-PE1-mpls] label advertise non-null
[ASBR-PE1-mpls] quit
[ASBR-PE1] mpls ldp
[ASBR-PE1-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[ASBR-PE1] interface vlan-interface 11
[ASBR-PE1-Vlan-interface11] ip address 1.1.1.1 255.0.0.0
[ASBR-PE1-Vlan-interface11] isis enable 1
[ASBR-PE1-Vlan-interface11] mpls
[ASBR-PE1-Vlan-interface11] mpls ldp
[ASBR-PE1-Vlan-interface11] quit
```

Configure interface VLAN-interface 12 and enable MPLS on it.

```
[ASBR-PE1] interface vlan-interface 12
[ASBR-PE1-Vlan-interface12] ip address 11.0.0.2 255.0.0.0
[ASBR-PE1-Vlan-interface12] mpls
[ASBR-PE1-Vlan-interface12] 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-policy1] apply mpls-label
[ASBR-PE1-route-policy1] quit
[ASBR-PE1] route-policy policy2 permit node 1
[ASBR-PE1-route-policy2] if-match mpls-label
[ASBR-PE1-route-policy2] apply mpls-label
[ASBR-PE1-route-policy2] quit
```

Start BGP on ASBR-PE 1 and redistribute routes from IS-IS process 1.

```
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] import-route isis 1
```

Use routing policy policy2 to filter routes advertised to IBGP peer 2.2.2.9.

```
[ASBR-PE1-bgp] peer 2.2.2.9 as-number 100
[ASBR-PE1-bgp] peer 2.2.2.9 route-policy policy2 export
```

Configure the capability to advertise labeled routes to IBGP peer 2.2.2.9 and to receive labeled routes from the peer.

```
[ASBR-PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[ASBR-PE1-bgp] peer 2.2.2.9 label-route-capability
```

Use routing policy policy1 to filter routes advertised to EBGP peer 11.0.0.1.

```
[ASBR-PE1-bgp] peer 11.0.0.1 as-number 600
[ASBR-PE1-bgp] peer 11.0.0.1 route-policy policy1 export
```

Configure the capability to advertise labeled routes to EBGP peer 11.0.0.1 and to receive labeled routes from the peer.

```
[ASBR-PE1-bgp] peer 11.0.0.1 label-route-capability
[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 10.222.222.222.222.00
[ASBR-PE2-isis-1] quit
```

Configure LSR ID, enable MPLS and LDP.

```
[ASBR-PE2] mpls lsr-id 4.4.4.9
[ASBR-PE2] mpls
[ASBR-PE2-mpls] label advertise non-null
[ASBR-PE2-mpls] quit
[ASBR-PE2] mpls ldp
[ASBR-PE2-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[ASBR-PE2] interface vlan-interface 11
[ASBR-PE2-Vlan-interface11] ip address 9.1.1.1 255.0.0.0
[ASBR-PE2-Vlan-interface11] isis enable 1
[ASBR-PE2-Vlan-interface11] mpls
[ASBR-PE2-Vlan-interface11] mpls ldp
[ASBR-PE2-Vlan-interface11] 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 VLAN-interface 12 and enable MPLS on it.

```
[ASBR-PE2] interface vlan-interface 12
[ASBR-PE2-Vlan-interface12] ip address 11.0.0.1 255.0.0.0
[ASBR-PE2-Vlan-interface12] mpls
[ASBR-PE2-Vlan-interface12] quit
```

Create routing policies.

```
[ASBR-PE2] route-policy policy1 permit node 1
New Sequence of this List
[ASBR-PE2-route-policy1] apply mpls-label
[ASBR-PE2-route-policy1] quit
[ASBR-PE2] route-policy policy2 permit node 1
[ASBR-PE2-route-policy2] if-match mpls-label
[ASBR-PE2-route-policy2] apply mpls-label
[ASBR-PE2-route-policy2] quit
```

Start BGP on ASBR-PE 2 and redistribute routes from IS-IS process 1.

```
[ASBR-PE2] bgp 600
[ASBR-PE2-bgp] import-route isis 1
```

Configure the capability to advertise labeled routes to IBGP peer 5.5.5.9 and to receive labeled routes from the peer.

```
[ASBR-PE2-bgp] peer 5.5.5.9 as-number 600
[ASBR-PE2-bgp] peer 5.5.5.9 connect-interface loopback 1
[ASBR-PE2-bgp] peer 5.5.5.9 label-route-capability
```

Use routing policy policy2 to filter routes advertised to IBGP peer 5.5.5.9.

```
[ASBR-PE2-bgp] peer 5.5.5.9 route-policy policy2 export
```

Use routing policy policy1 to filter routes advertised to EBGP peer 11.0.0.2.

```
[ASBR-PE2-bgp] peer 11.0.0.2 as-number 100
[ASBR-PE2-bgp] peer 11.0.0.2 route-policy policy1 export
```

Configure the capability to advertise labeled routes to EBGP peer 11.0.0.2 and to receive labeled routes from the peer.

```
[ASBR-PE2-bgp] peer 11.0.0.2 label-route-capability
[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 10.111.111.111.111.00
[PE2-isis-1] quit
```

Configure LSR ID, enable MPLS and LDP.

```
[PE2] mpls lsr-id 5.5.5.9
[PE2] mpls
[PE2-mpls] label advertise non-null
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[PE2] interface vlan-interface 11
[PE2-Vlan-interface11] ip address 9.1.1.2 255.0.0.0
[PE2-Vlan-interface11] isis enable 1
[PE2-Vlan-interface11] mpls
[PE2-Vlan-interface11] mpls ldp
[PE2-Vlan-interface11] 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 VPN 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
```

Start BGP on PE 2.

```
[PE2] bgp 600
```

Configure 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 1
[PE2-bgp] peer 4.4.4.9 label-route-capability
```

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 1
[PE2-bgp] peer 2.2.2.9 ebgp-max-hop 10
```

Configure peer 2.2.2.9 as a VPNv4 peer.

```
[PE2-bgp] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 2.2.2.9 enable
[PE2-bgp-af-vpnv4] quit
```

Redistribute direct routes to the routing table of **vpn1**.

```
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] import-route direct
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

After you complete the configurations, PE 1 and PE 2 are able to ping each other:

```
[PE2] ping -vpn-instance vpn1 30.0.0.1
[PE1] ping -vpn-instance vpn1 20.0.0.1
```

Example for configuring carrier's carrier

Network requirements

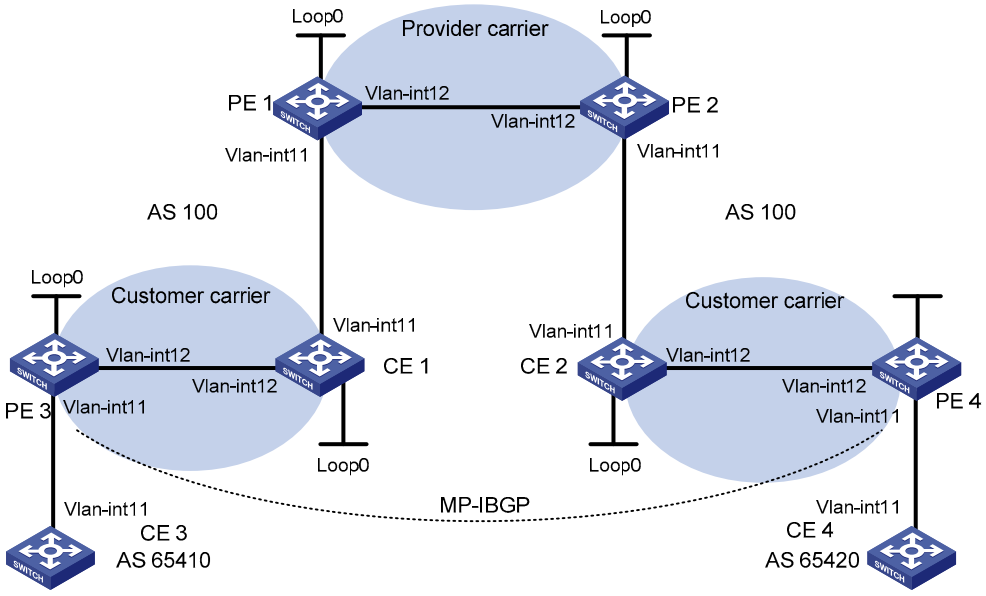
Configure carrier's carrier for the scenario shown in Figure 76. In this scenario:

- PE 1 and PE 2 are the provider carrier's PE switches. They provide VPN services for the customer carrier.
- CE 1 and CE 2 are the customer carrier's switches. They are connected to the provider carrier's backbone as CE switches.
- PE 3 and PE 4 are the customer carrier's PE switches. 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 76 Configure carrier's carrier



Device	Interface	IP address	Device	Interface	IP address
CE 3	Vlan-int11	100.1.1.1/24	CE 4	Vlan-int11	120.1.1.1/24
PE 3	Loop0	1.1.1.9/32	PE 4	Loop0	6.6.6.9/32
	Vlan-int11	100.1.1.2/24		Vlan-int11	120.1.1.2/24
	Vlan-int12	10.1.1.1/24		Vlan-int12	20.1.1.2/24
CE 1	Loop0	2.2.2.9/32	CE 2	Loop0	5.5.5.9/32
	Vlan-int12	10.1.1.2/24		Vlan-int11	21.1.1.2/24
	Vlan-int11	11.1.1.1/24		Vlan-int12	20.1.1.1/24
PE 1	Loop0	3.3.3.9/32	PE 2	Loop0	4.4.4.9/32
	Vlan-int11	11.1.1.2/24		Vlan-int12	30.1.1.2/24
	Vlan-int12	30.1.1.1/24		Vlan-int11	21.1.1.1/24

Procedure

1. Configure MPLS L3VPN on the provider carrier backbone: start IS-IS as the IGP, enable LDP between PE 1 and PE 2, and establish 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
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-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 vlan-interface 12
[PE1-Vlan-interface12] ip address 30.1.1.1 24
[PE1-Vlan-interface12] isis enable 1
[PE1-Vlan-interface12] mpls
[PE1-Vlan-interface12] mpls ldp
[PE1-Vlan-interface2] mpls ldp transport-address interface
[PE1-Vlan-interface2] 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] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 4.4.4.9 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

The configurations for PE 2 are similar to those for PE 1. The detailed configuration steps are omitted.

After completing the configurations, you can see that the LDP session has been established successfully by issuing **display mpls ldp session** on PE 1 or PE 2. Issuing **display bgp peer**, you can see that the BGP peer relationship has been established and has reached the state of Established. Issuing **display isis peer**, you can see that the IS-IS neighbor relationship has been set up. Take PE 1 as an example:

```
[PE1] display mpls ldp session
               LDP Session(s) in Public Network
Total number of sessions: 1
 ------------------------------------------------------------------
 Peer-ID        Status        LAM  SsnRole  FT   MD5  KA-Sent/Rcv
 ------------------------------------------------------------------
 4.4.4.9:0      Operational   DU   Active   Off  Off  378/378
 ------------------------------------------------------------------
 LAM : Label Advertisement Mode         FT  : Fault Tolerance
[PE1] display bgp peer
 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      162      145     0        0  02:12:47 Established
[PE1] display isis peer
                        Peer information for ISIS(1)
                        ----------------------------
  System Id    Interface        Circuit Id  State HoldTime  Type   PRI
0000.0000.0005 Vlan-interface2  001         Up     29s      L1L2   --
```

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 respectively

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
[PE3-mpls] quit
[PE3] mpls ldp
[PE3-mpls-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 vlan-interface 12
[PE3-Vlan-interface12] ip address 10.1.1.1 24
[PE3-Vlan-interface12] isis enable 2
[PE3-Vlan-interface12] mpls
[PE3-Vlan-interface12] mpls ldp
[PE3-Vlan-interface12] mpls ldp transport-address interface
[PE3-Vlan-interface12] 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
[CE1-mpls] quit
[CE1] mpls ldp
[CE1-mpls-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 vlan-interface 12
[CE1-Vlan-interface12] ip address 10.1.1.2 24
[CE1-Vlan-interface12] isis enable 2
[CE1-Vlan-interface12] mpls
[CE1-Vlan-interface12] mpls ldp
[CE1-Vlan-interface12] mpls ldp transport-address interface
[CE1-Vlan-interface12] quit
```

After you complete the configurations, PE 3 and CE 1 can establish the LDP session and IS-IS neighbor relationship between them.

The configurations for PE 4 and CE 2 are similar to those for PE 3 and CE 1. The detailed configuration steps are omitted.

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 and inject IS-IS routes.

```
[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 vpn-instance vpn1
[PE1-mpls-ldp-vpn-instance-vpn1] 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 vlan-interface 11
[PE1-Vlan-interface11] ip binding vpn-instance vpn1
[PE1-Vlan-interface11] ip address 11.1.1.2 24
[PE1-Vlan-interface11] isis enable 2
[PE1-Vlan-interface11] mpls
[PE1-Vlan-interface11] mpls ldp
[PE1-Vlan-interface11] mpls ldp transport-address interface
[PE1-Vlan-interface11] quit
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import isis 2
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

Configure CE 1.

```
[CE1] interface vlan-interface 11
[CE1-Vlan-interface11] ip address 11.1.1.1 24
[CE1-Vlan-interface11] isis enable 2
[CE1-Vlan-interface11] mpls
[CE1-Vlan-interface11] mpls ldp
[CE1-Vlan-interface11] mpls ldp transport-address interface
[CE1-Vlan-interface11] quit
```

After you complete the configurations, PE 1 and CE 1 can establish the LDP session and IS-IS neighbor relationship between them.

The configurations for PE 2 and CE 2 are similar to those for PE 1 and CE 1. The detailed configuration steps are omitted.

4. Perform configuration to connect CEs of customers to the PEs of the customer carrier.

Configure CE 3.

```
<CE3> system-view
[CE3] interface vlan-interface 11
[CE3-Vlan-interface11] ip address 100.1.1.1 24
[CE3-Vlan-interface11] quit
[CE3] bgp 65410
[CE3-bgp] peer 100.1.1.2 as-number 100
[CE3-bgp] import-route direct
[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 vlan-interface 11
[PE3-Vlan-interface11] ip binding vpn-instance vpn1
[PE3-Vlan-interface11] ip address 100.1.1.2 24
[PE3-Vlan-interface11] quit
[PE3] bgp 100
[PE3-bgp] ipv4-family vpn-instance vpn1
[PE3-bgp-vpn1] peer 100.1.1.1 as-number 65410
[PE3-bgp-vpn1] import-route direct
[PE3-bgp-vpn1] quit
[PE3-bgp] quit
```

The configurations for PE 4 and CE 4 are similar to those for PE 3 and CE 3. The detailed configuration steps are omitted.

5. Configure MP-IBGP peer relationship between PEs of the customer carrier to exchange the VPN routes of the customer carrier's 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] ipv4-family vpnv4
[PE3-bgp-af-vpnv4] peer 6.6.6.9 enable
[PE3-bgp-af-vpnv4] quit
[PE3-bgp] quit
```

The configurations for PE 4 are similar to those for PE 3. The detailed configuration steps are omitted.

6. Verify your configurations

Issue **display ip routing-table** on PE 1 and PE 2. You can see 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        Vlan12
30.1.1.0/24         Direct 0    0         30.1.1.1        Vlan12
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        Vlan12
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
```

Issuing **display ip routing-table vpn-instance** on PE 1 and PE 2, you can see 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        Vlan11
2.2.2.9/32          ISIS   15   10        11.1.1.1        Vlan11
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        Vlan11
11.1.1.0/24         Direct 0    0         11.1.1.1        Vlan11
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        Vlan11
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
```

Issuing **display ip routing-table** on CE 1 and CE 2, you can see 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        Vlan12
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        Vlan11
6.6.6.9/32          ISIS   15   74        11.1.1.2        Vlan11
10.1.1.0/24         Direct 0    0         10.1.1.2        Vlan12
10.1.1.1/32         Direct 0    0         10.1.1.1        Vlan12
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        Vlan1
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        Vlan11
20.1.1.0/24         ISIS   15   74        11.1.1.2        Vlan11
21.1.1.0/24         ISIS   15   74        11.1.1.2        Vlan11
```

```
21.1.1.2/32         ISIS   15   74     11.1.1.2        Vlan11
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
```

Issuing **display ip routing-table** on PE 3 and PE 4, you can see 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        Vlan12
5.5.5.9/32          ISIS   15   84     10.1.1.2        Vlan12
6.6.6.9/32          ISIS   15   84     10.1.1.2        Vlan12
10.1.1.0/24         Direct 0    0      10.1.1.1        Vlan12
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        Vlan12
11.1.1.0/24         ISIS   15   20     10.1.1.2        Vlan12
20.1.1.0/24         ISIS   15   84     10.1.1.2        Vlan12
21.1.1.0/24         ISIS   15   84     10.1.1.2        Vlan12
21.1.1.2/32         ISIS   15   84     10.1.1.2        Vlan12
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
```

Issuing **display ip routing-table vpn-instance** on PE 3 and PE 4, you can see 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       Vlan11
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:

```
[PE3] ping 20.1.1.2
  PING 20.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 20.1.1.2: bytes=56 Sequence=1 ttl=252 time=127 ms
    Reply from 20.1.1.2: bytes=56 Sequence=2 ttl=252 time=97 ms
    Reply from 20.1.1.2: bytes=56 Sequence=3 ttl=252 time=83 ms
    Reply from 20.1.1.2: bytes=56 Sequence=4 ttl=252 time=70 ms
    Reply from 20.1.1.2: bytes=56 Sequence=5 ttl=252 time=60 ms

  --- 20.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 60/87/127 ms
```

CE 3 and CE 4 can ping each other:

```
[CE3] ping 120.1.1.1
```

```
PING 120.1.1.1: 56  data bytes, press CTRL_C to break
  Reply from 120.1.1.1: bytes=56 Sequence=1 ttl=252 time=102 ms
  Reply from 120.1.1.1: bytes=56 Sequence=2 ttl=252 time=69 ms
  Reply from 120.1.1.1: bytes=56 Sequence=3 ttl=252 time=105 ms
  Reply from 120.1.1.1: bytes=56 Sequence=4 ttl=252 time=88 ms
  Reply from 120.1.1.1: bytes=56 Sequence=5 ttl=252 time=87 ms

--- 120.1.1.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 69/90/105 ms
```

Example for configuring nested VPN

Network requirements

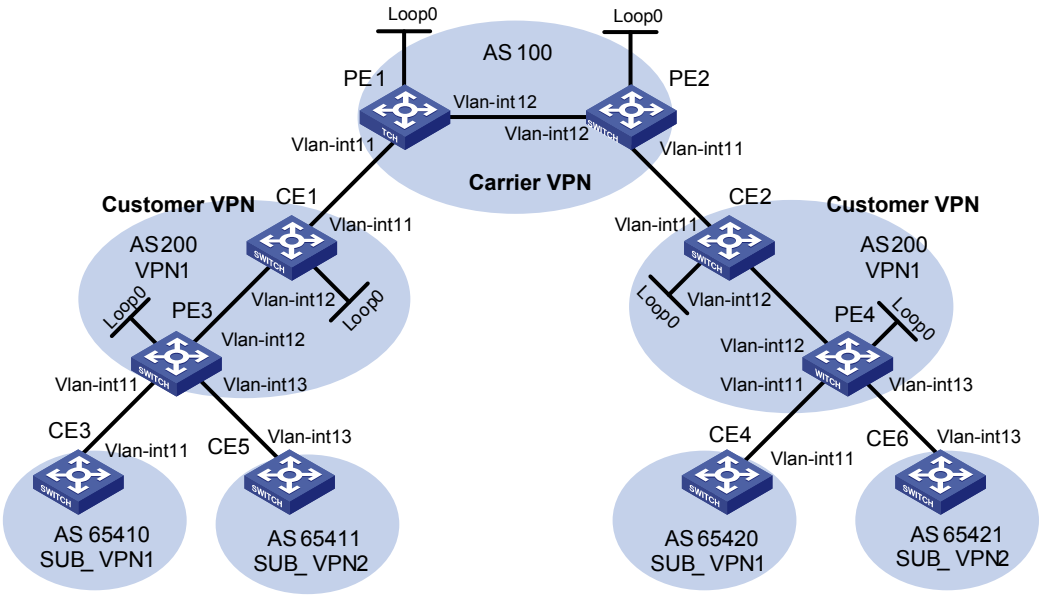
The service provider provides nested VPN services for users, as shown in Figure 77, where:

- 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 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 the sub-VPNs for 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 CE (CE 1 or CE 2 in this example), a service provider PE replaces the RD of the VPNv4 route with the RD of the MPLS VPN on the service provider network where the CE resides, 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 as usual.
- To implement exchange of sub-VPN routes between customer PEs and service provider PEs, MP-EBGP peers should be established between service provider PEs and customer CEs.

Figure 77 Configure nested VPN



Device	Interface	IP address	Device	Interface	IP address
CE 1	Loop0	2.2.2.9/32	CE 2	Loop0	5.5.5.9/32
	Vlan-int12	10.1.1.2/24		Vlan-int11	21.1.1.2/24
	Vlan-int11	11.1.1.1/24		Vlan-int12	20.1.1.1/24
CE 3	Vlan-int11	100.1.1.1/24	CE 4	Vlan-int11	120.1.1.1/24
CE 5	Vlan-int13	110.1.1.1/24	CE 6	Vlan-int13	130.1.1.1/24
PE 1	Loop0	3.3.3.9/32	PE 2	Loop0	4.4.4.9/32
	Vlan-int11	11.1.1.2/24		Vlan-int11	21.1.1.1/24
	Vlan-int12	30.1.1.1/24		Vlan-int12	30.1.1.2/24
PE 3	Loop0	1.1.1.9/32	PE 4	Loop0	6.6.6.9/32
	Vlan-int11	100.1.1.2/24		Vlan-int11	120.1.1.2/24
	Vlan-int12	10.1.1.1/24		Vlan-int12	20.1.1.2/24
	Vlan-int13	110.1.1.2/24		Vlan-int13	130.1.1.2/24

Procedure

1. Configure MPLS L3VPN on the service provider backbone, using IS-IS as the IGP protocol, and enabling LDP and establishing 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
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-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 vlan-interface 12
[PE1-Vlan-interface12] ip address 30.1.1.1 24
[PE1-Vlan-interface12] isis enable 1
[PE1-Vlan-interface12] mpls
[PE1-Vlan-interface12] mpls ldp
[PE1-Vlan-interface12] 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] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 4.4.4.9 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

Configurations on PE 2 are similar to those on PE 1, and are thus omitted here.

After completing the configurations you can execute commands **display mpls ldp session**, **display bgp peer** and **display isis peer** respectively on either PE 1 or PE 2. You can see that the LDP session is established, the BGP peer relationship is established and in the Established state, and the IS-IS neighbor relationship is established and up.

The following uses PE 1 for illustration.

```
[PE1] display mpls ldp session
               LDP Session(s) in Public Network
Total number of sessions: 1
 ------------------------------------------------------------------
 Peer-ID        Status        LAM  SsnRole  FT   MD5  KA-Sent/Rcv
 ------------------------------------------------------------------
 4.4.4.9:0      Operational   DU   Active   Off  Off  378/378
 ------------------------------------------------------------------
 LAM : Label Advertisement Mode         FT  : Fault Tolerance
[PE1] display bgp peer
 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      162      145     0        0  02:12:47 Established
[PE1] display isis peer
                         Peer information for ISIS(1)
                         ---------------------------
  System Id    Interface        Circuit Id  State HoldTime  Type   PRI
0000.0000.0005 Vlan-interface12  001         Up     29s      L1L2   --
```

2. Configure the customer VPN, using IS-IS as the IGP protocol and enabling 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
[PE3-mpls] quit
[PE3] mpls ldp
[PE3-mpls-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 vlan-interface 12
[PE3-Vlan-interface12] ip address 10.1.1.1 24
[PE3-Vlan-interface12] isis enable 2
[PE3-Vlan-interface12] mpls
[PE3-Vlan-interface12] mpls ldp
[PE3-Vlan-interface12] 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
[CE1-mpls] quit
[CE1] mpls ldp
[CE1-mpls-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 vlan-interface 12
[CE1-Vlan-interface12] ip address 10.1.1.2 24
[CE1-Vlan-interface12] isis enable 2
[CE1-Vlan-interface12] mpls
[CE1-Vlan-interface12] mpls ldp
[CE1-Vlan-interface12] quit
```

After the configurations, LDP and IS-IS neighbor relationship can be established between PE 3 and CE 1.

Configurations on PE 4 and CE 2 are similar to those on PE 3 and CE 1 respectively, and are thus omitted here.

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 vlan-interface 11
[PE1-Vlan-interface11] ip binding vpn-instance vpn1
[PE1-Vlan-interface11] ip address 11.1.1.2 24
[PE1-Vlan-interface11] mpls
[PE1-Vlan-interface11] quit
[PE1] bgp 100
[PE1-bgp] ipv4-family 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 vlan-interface 11
[CE1-Vlan-interface11] ip address 11.1.1.1 24
[CE1-Vlan-interface11] mpls
[CE1-Vlan-interface11] quit
[CE1] bgp 200
[CE1-bgp] peer 11.1.1.2 as-number 100
[CE1-bgp] import isis 2
[CE1-bgp] quit
```

Configurations on PE 2 and CE 2 are similar to those on PE 1 and CE 1 respectively, and are thus omitted here.

4. Connect sub-VPN CEs to the customer VPN PEs

Configure CE 3.

```
<CE3> system-view
[CE3] interface vlan-interface11
[CE3-Vlan-interface11] ip address 100.1.1.1 24
[CE3-Vlan-interface11] quit
[CE3] bgp 65410
[CE3-bgp] peer 100.1.1.2 as-number 200
[CE3-bgp] import-route direct
[CE3-bgp] quit
```

Configure CE 5.

```
<CE5> system-view
[CE5] interface vlan-interface 13
[CE5-Vlan-interface13] ip address 110.1.1.1 24
[CE5-Vlan-interface13] quit
[CE5] bgp 65411
[CE5-bgp] peer 110.1.1.2 as-number 200
[CE5-bgp] import-route direct
[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 vlan-interface 11
[PE3-Vlan-interface11] ip binding vpn-instance SUB_VPN1
[PE3-Vlan-interface11] ip address 100.1.1.2 24
[PE3-Vlan-interface11] 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 vlan-interface 13
[PE3-Vlan-interface13] ip binding vpn-instance SUB_VPN2
[PE3-Vlan-interface13] ip address 110.1.1.2 24
[PE3-Vlan-interface13] quit
[PE3] bgp 200
[PE3-bgp] ipv4-family vpn-instance SUB_VPN1
[PE3-bgp-SUB_VPN1] peer 100.1.1.1 as-number 65410
[PE3-bgp-SUB_VPN1] import-route direct
[PE3-bgp-SUB_VPN1] quit
[PE3-bgp] ipv4-family vpn-instance SUB_VPN2
[PE3-bgp-SUB_VPN2] peer 100.1.1.1 as-number 65411
[PE3-bgp-SUB_VPN2] import-route direct
[PE3-bgp-SUB_VPN2] quit
[PE3-bgp] quit
```

Configurations on PE 4, CE 4 and CE 6 are similar to those on PE 3, CE 3 and CE5 respectively, and are thus omitted here.

5. Establish MP-EBGP peer relationship between service provider PEs and their CEs to exchange user VPNv4 routes.

Configure PE 1, enabling nested VPN.

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] nesting-vpn
[PE1-bgp-af-vpnv4] peer 11.1.1.1 vpn-instance vpn1 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

Configure CE 1, enabling VPNv4 capability and establishing VPNv4 neighbor relationship between CE 1 and PE 1.

```
[CE1] bgp 200
[CE1-bgp] ipv4-family vpnv4
[CE1-bgp-af-vpnv4] peer 11.1.1.2 enable
```

Specify to allow the local AS number to appear in the AS-PATH attribute of the routes received.

```
[CE1-bgp-af-vpnv4] peer 11.1.1.2 allow-as-loop 2
```

Specify to receive all VPNv4 routes.

```
[CE1-bgp-af-vpnv4] undo policy vpn-target
```

```
[CE1-bgp-af-vpnv4] quit
[CE1-bgp] quit
```

Configurations on PE 2 and CE 2 are similar to those on PE 1 and CE 1 respectively, and are thus omitted here.

6. Establish MP-IBGP peer relationship 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] ipv4-family vpnv4
[PE3-bgp-af-vpnv4] peer 2.2.2.9 enable
```

Specify to allow the local AS number to appear in the AS-PATH attribute of the routes received.

```
[PE3-bgp-af-vpnv4] peer 2.2.2.9 allow-as-loop 2
[PE3-bgp-af-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] ipv4-family vpnv4
[CE1-bgp-af-vpnv4] peer 1.1.1.9 enable
[CE1-bgp-af-vpnv4] undo policy vpn-target
[CE1-bgp-af-vpnv4] quit
[CE1-bgp] quit
```

Configurations on PE 4 and CE 2 are similar to those on PE 3 and CE 1 respectively, and are thus omitted here.

7. Verify the configurations.

Execute **display ip routing-table** on PE 1 and PE 2. You can see that the public routing tables contain only routes on the service provider network. The following uses PE 1 for illustration.

```
[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        Vlan12
30.1.1.0/24         Direct  0    0            30.1.1.1        Vlan12
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        Vlan12
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 **display ip routing-table vpn-instance** on PE 1 and PE 2. You can that the VPN routing tables contain sub-VPN routes. The following uses PE 1 for illustration.

```
[PE1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 9        Routes : 9

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

11.1.1.0/24         Direct 0    0            11.1.1.1        Vlan11
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        Vlan11
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
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
130.1.1.0/24        BGP    255  0            4.4.4.9         NULL0
```

Execute **display bgp vpnv4 all routing-table** on CE 1 and CE 2. You can that the VPNv4 routing tables on the customer VPN contain internal sub-VPN routes. The following uses CE 1 for illustration.

```
[CE1] display bgp vpnv4 all routing-table
BGP Local router ID is 11.11.11.11
 Status codes: * - valid, ^ - VPNv4 best, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete


 Total number of routes from all PE: 4

 Route Distinguisher: 100:1

    Network            NextHop         In/Out Label     MED        LocPrf

*>  100.1.1.0/24       1.1.1.9         1024/1024

Route Distinguisher: 101:1

    Network            NextHop         In/Out Label     MED        LocPrf

*^  100.1.1.0/24       1.1.1.9         1024/1024

Route Distinguisher: 101:1

    Network            NextHop         In/Out Label     MED        LocPrf

* > 110.1.1.0/24       1.1.1.9         1025/1025
```

```
 Route Distinguisher: 200:1

    Network            NextHop         In/Out Label     MED        LocPrf

* >  120.1.1.0/24       11.1.1.2        1026/1027

Route Distinguisher: 201:1

    Network            NextHop         In/Out Label     MED        LocPrf

* > 130.1.1.0/24       11.1.1.2        1027/1028
```

Execute **display ip routing-table vpn-instance SUB_VPN1** on PE 3 and PE 4. You can see that the VPN routing tables contain routes sent by the provider PE to user sub-VPN. The following uses PE 3 for illustration.

```
[PE3] display ip routing-table vpn-instance SUB_VPN1
Routing Tables: SUB_VPN1
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

100.1.1.0/24        Direct 0    0            100.1.1.2       Vlan11
100.1.1.2/32        Direct 0    0            127.0.0.1       InLoop0
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
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
```

Execute **display ip routing-table** on CE 3 and CE 4. You can see that the routing tables contain routes of remote sub-VPNs. The following uses CE 3 for illustration.

```
[CE3] display ip routing-table
Routing Tables: Public
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

100.1.1.0/24        Direct 0    0            100.1.1.1       Vlan11
100.1.1.1/32        Direct 0    0            127.0.0.1       InLoop0
120.1.1.0/24        BGP    255  0            100.1.1.2       Vlan1
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 **display ip routing-table** on CE5 and CE 6. You can see that the routing tables contain routes of remote sub-VPNs. The following uses CE5 for illustration.

```
[CE5] display ip routing-table
Routing Tables: Public
         Destinations : 5        Routes : 5
```

```
Destination/Mask    Proto   Pre  Cost         NextHop         Interface

110.1.1.0/24        Direct 0    0            110.1.1.1       Vlan11
110.1.1.1/32        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
130.1.1.0/24        BGP    255  0            110.1.1.2       Vlan11
```

CE 3 and CE 4 can ping each other successfully.

```
[CE3] ping 120.1.1.1
  PING 120.1.1.1: 56  data bytes, press CTRL_C to break
    Reply from 120.1.1.1: bytes=56 Sequence=1 ttl=252 time=102 ms
    Reply from 120.1.1.1: bytes=56 Sequence=2 ttl=252 time=69 ms
    Reply from 120.1.1.1: bytes=56 Sequence=3 ttl=252 time=105 ms
    Reply from 120.1.1.1: bytes=56 Sequence=4 ttl=252 time=88 ms
    Reply from 120.1.1.1: bytes=56 Sequence=5 ttl=252 time=87 ms

  --- 120.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 69/90/105 ms
```

CE5 and CE 6 can ping each other successfully.

```
[CE5] ping 130.1.1.1
  PING 130.1.1.1: 56  data bytes, press CTRL_C to break
    Reply from 130.1.1.1: bytes=56 Sequence=1 ttl=252 time=102 ms
    Reply from 130.1.1.1: bytes=56 Sequence=2 ttl=252 time=69 ms
    Reply from 130.1.1.1: bytes=56 Sequence=3 ttl=252 time=105 ms
    Reply from 130.1.1.1: bytes=56 Sequence=4 ttl=252 time=88 ms
    Reply from 130.1.1.1: bytes=56 Sequence=5 ttl=252 time=87 ms

  --- 130.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
round-trip min/avg/max = 69/90/105 ms
```

CE 3 and CE 6 cannot ping each other.

```
[CE3] ping 130.1.1.1
  PING 130.1.1.1: 56  data bytes, press CTRL_C to break
    Request time out
    Request time out
    Request time out
    Request time out
    Request time out

  --- 130.1.1.1 ping statistics ---
    5 packet(s) transmitted
```

```
0 packet(s) received
100.00% packet loss
```

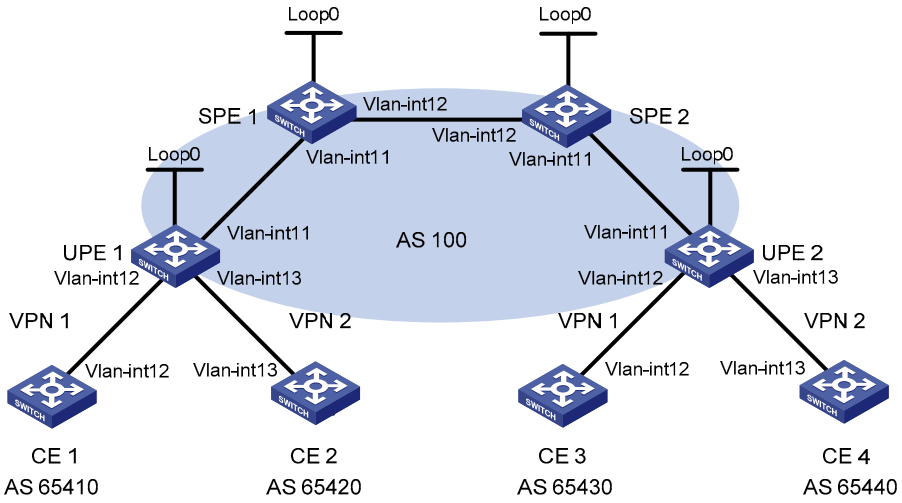
Example for configuring HoVPN

Network requirements

There are two levels of networks, the backbone and the MPLS VPN networks, as shown in Figure 78.

- 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 the 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 78 Configure HoVPN



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int12	10.2.1.1/24	CE 3	Vlan-int12	10.1.1.1/24
CE 2	Vlan-int13	10.4.1.1/24	CE 4	Vlan-int13	10.3.1.1/24
UPE 1	Loop0	1.1.1.9/32	UPE 2	Loop0	4.4.4.9/32
	Vlan-int11	172.1.1.1/24		Vlan-int11	172.2.1.1/24
	Vlan-int12	10.2.1.2/24		Vlan-int12	10.1.1.2/24
	Vlan-int13	10.4.1.2/24		Vlan-int13	10.3.1.2/24
SPE 1	Loop0	2.2.2.9/32	SPE 2	Loop0	3.3.3.9/32
	Vlan-int11	172.1.1.2/24		Vlan-int11	172.2.1.2/24
	Vlan-int12	180.1.1.1/24		Vlan-int12	180.1.1.2/24

Procedure

1. Configure UPE 1

Configure MPLS basic capability 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
[UPE1-mpls] quit
[UPE1] mpls ldp
[UPE1-mpls-ldp] quit
[UPE1] interface vlan-interface 11
[UPE1-Vlan-interface11] ip address 172.1.1.1 24
[UPE1-Vlan-interface11] mpls
[UPE1-Vlan-interface11] mpls ldp
[UPE1-Vlan-interface11] quit
```

Configure the IGP protocol, OSPF, for example.

```
[UPE1] ospf
[UPE1-ospf-1] area 0
[UPE1-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[UPE1-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[UPE1-ospf-1-area-0.0.0.0] quit
[UPE1-ospf-1] quit
```

Configure VPN instances vpn1 and vpn2, allowing CE 1 and CE 2 to access UPE 1.

```
[UPE1] ip vpn-instance vpn1
[UPE1-vpn-instance-vpn1] route-distinguisher 100:1
[UPE1-vpn-instance-vpn1] vpn-target 100:1 both
[UPE1-vpn-instance-vpn1] quit
[UPE1] ip vpn-instance vpn2
[UPE1-vpn-instance-vpn2] route-distinguisher 100:2
[UPE1-vpn-instance-vpn2] vpn-target 100:2 both
[UPE1-vpn-instance-vpn2] quit
[UPE1] interface vlan-interface 12
[UPE1-Vlan-interface12] ip binding vpn-instance vpn1
[UPE1-Vlan-interface12] ip address 10.2.1.2 24
[UPE1-Vlan-interface12] quit
[UPE1] interface vlan-interface 13
[UPE1-Vlan-interface13] ip binding vpn-instance vpn2
[UPE1-Vlan-interface13] ip address 10.4.1.2 24
[UPE1-Vlan-interface13] quit
```

Configure UPE 1 to establish MP-IBGP peer relationship with SPE 1 and to inject VPN routes.

```
[UPE1] bgp 100
[UPE1-bgp] peer 2.2.2.9 as-number 100
[UPE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[UPE1-bgp] ipv4-family vpnv4
[UPE1-bgp-af-vpnv4] peer 2.2.2.9 enable
[UPE1-bgp-af-vpnv4] quit
[UPE1-bgp] ipv4-family vpn-instance vpn1
[UPE1-bgp-vpn1] peer 10.2.1.1 as-number 65410
[UPE1-bgp-vpn1] import-route direct
[UPE1-bgp-vpn1] quit
[UPE1-bgp] ipv4-family vpn-instance vpn2
```

```
[UPE1-bgp-vpn1] peer 10.4.1.1 as-number 65420
[UPE1-bgp-vpn1] import-route direct
[UPE1-bgp-vpn1] quit
[UPE1-bgp] quit
```

2. Configure CE 1

```
<CE1> system-view
[CE1] interface vlan-interface 12
[CE1-Vlan-interface12] ip address 10.2.1.1 255.255.255.0
[CE1-Vlan-interface12] quit
[CE1] bgp 65410
[CE1-bgp] peer 10.2.1.2 as-number 100
[CE1-bgp] import-route direct
[CE1] quit
```

3. Configure CE 2

```
<CE2> system-view
[CE2] interface vlan-interface 13
[CE2-Vlan-interface13] ip address 10.4.1.1 255.255.255.0
[CE2-Vlan-interface13] quit
[CE2] bgp 65420
[CE2-bgp] peer 10.4.1.2 as-number 100
[CE2-bgp] import-route direct
[CE2] quit
```

4. Configure UPE 2

Configure MPLS basic capability 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
[UPE2-mpls] quit
[UPE2] mpls ldp
[UPE2-mpls-ldp] quit
[UPE2] interface vlan-interface 11
[UPE2-Vlan-interface11] ip address 172.2.1.1 24
[UPE2-Vlan-interface11] mpls
[UPE2-Vlan-interface11] mpls ldp
[UPE2-Vlan-interface11] quit
```

Configure the IGP protocol, OSPF, for 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 vlan-interface 12
[UPE2-Vlan-interface12] ip binding vpn-instance vpn1
[UPE2-Vlan-interface12] ip address 10.1.1.2 24
[UPE2-Vlan-interface12] quit
[UPE2] interface vlan-interface 13
[UPE2-Vlan-interface13] ip binding vpn-instance vpn2
[UPE2-Vlan-interface13] ip address 10.3.1.2 24
[UPE2-Vlan-interface13] quit
```

Configure UPE 2 to establish MP-IBGP peer relationship with SPE 2 and to inject VPN routes.

```
[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] ipv4-family vpnv4
[UPE2-bgp-af-vpnv4] peer 3.3.3.9 enable
[UPE2-bgp-af-vpnv4] quit
[UPE2-bgp] ipv4-family vpn-instance vpn1
[UPE2-bgp-vpn1] peer 10.1.1.1 as-number 65430
[UPE2-bgp-vpn1] import-route direct
[UPE2-bgp-vpn1] quit
[UPE2-bgp] ipv4-family vpn-instance vpn2
[UPE2-bgp-vpn1] peer 10.3.1.1 as-number 65440
[UPE2-bgp-vpn1] import-route direct
[UPE2-bgp-vpn1] quit
[UPE2-bgp] quit
```

5. Configure CE 3

```
<CE3> system-view
[CE3] interface vlan-interface 12
[CE3-Vlan-interface12] ip address 10.1.1.1 255.255.255.0
[CE3-Vlan-interface12] quit
[CE3] bgp 65430
[CE3-bgp] peer 10.1.1.2 as-number 100
[CE3-bgp] import-route direct
[CE3] quit
```

6. Configure CE 4

```
<CE4> system-view
[CE4] interface vlan-interface 13
[CE4-Vlan-interface13] ip address 10.3.1.1 255.255.255.0
[CE4-Vlan-interface13] quit
[CE4] bgp 65440
```

```
[CE4-bgp] peer 10.3.1.2 as-number 100
[CE4-bgp] import-route direct
[CE4] quit
```

7. Configure SPE 1

Configure MPLS basic capability 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
[SPE1-mpls] quit
[SPE1] mpls ldp
[SPE1-mpls-ldp] quit
[SPE1] interface vlan-interface 11
[SPE1-Vlan-interface11] ip address 172.1.1.2 24
[SPE1-Vlan-interface11] mpls
[SPE1-Vlan-interface11] mpls ldp
[SPE1-Vlan-interface11] quit
[SPE1] interface vlan-interface 12
[SPE1-Vlan-interface12] ip address 180.1.1.1 24
[SPE1-Vlan-interface12] mpls
[SPE1-Vlan-interface12] mpls ldp
[SPE1-Vlan-interface12] quit
```

Configure the IGP protocol, OSPF, for 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
```

Configure SPE 1 to establish MP-IBGP peer relationship with UPE 1 and to inject VPN routes, and specify UPE 1.

```
[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 1.1.1.9 next-hop-local
[SPE1-bgp] peer 3.3.3.9 as-number 100
[SPE1-bgp] peer 3.3.3.9 connect-interface loopback 0
[SPE1-bgp] ipv4-family vpnv4
[SPE1-bgp-af-vpnv4] peer 3.3.3.9 enable
[SPE1-bgp-af-vpnv4] peer 1.1.1.9 enable
[SPE1-bgp-af-vpnv4] peer 1.1.1.9 upe
[SPE1-bgp-af-vpnv4] quit
[SPE1-bgp]ipv4-family vpn-instance vpn1
[SPE1-bgp-vpn1] quit
[SPE1-bgp]ipv4-family vpn-instance vpn2
[SPE1-bgp-vpn2] quit
[SPE1-bgp] quit
```

Configure SPE 1 to advertise to UPE 1 the routes permitted by a routing policy, or, the routes of CE 3.

```
[SPE1] ip ip-prefix hope index 10 permit 10.1.1.1 24
[SPE1] route-policy hope permit node 0
[SPE1-route-policy] if-match ip-prefix hope
[SPE1-route-policy] quit
[SPE1] bgp 100
[SPE1-bgp] ipv4-family vpnv4
[SPE1-bgp-af-vpnv4] peer 1.1.1.9 upe route-policy hope export
```

8. Configure SPE 2

Configure MPLS basic capability 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
[SPE2-mpls] quit
[SPE2] mpls ldp
[SPE2-mpls-ldp] quit
[SPE2] interface vlan-interface 12
[SPE2-Vlan-interface12] ip address 180.1.1.2 24
[SPE2-Vlan-interface12] mpls
[SPE2-Vlan-interface12] mpls ldp
[SPE2-Vlan-interface12] quit
[SPE2] interface vlan-interface 11
[SPE2-Vlan-interface11] ip address 172.2.1.2 24
[SPE2-Vlan-interface11] mpls
[SPE2-Vlan-interface11] mpls ldp
[SPE2-Vlan-interface11] quit
```

Configure the IGP protocol, OSPF, for 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
```

Configure SPE 2 to establish MP-IBGP peer relationship with UPE 2 and to inject VPN routes, and specify UPE 2.

```
[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 4.4.4.9 next-hop-local
[SPE2-bgp] peer 2.2.2.9 as-number 100
[SPE2-bgp] peer 2.2.2.9 connect-interface loopback 0
[SPE2-bgp] ipv4-family vpnv4
[SPE2-bgp-af-vpnv4] peer 2.2.2.9 enable
[SPE2-bgp-af-vpnv4] peer 4.4.4.9 enable
[SPE2-bgp-af-vpnv4] peer 4.4.4.9 upe
[SPE2-bgp-af-vpnv4] quit
[SPE2-bgp]ipv4-family vpn-instance vpn1
[SPE2-bgp-vpn1] quit
[SPE2-bgp]ipv4-family vpn-instance vpn2
[SPE2-bgp-vpn2] quit
[SPE2-bgp] quit
```

Configure SPE 2 to advertise to UPE 2 the routes permitted by a routing policy, or, the routes of CE 1.

```
[SPE2] ip ip-prefix hope index 10 permit  10.2.1.1 24
[SPE2] route-policy hope permit node 0
[SPE2-route-policy] if-match ip-prefix hope
[SPE2-route-policy] quit
[SPE2] bgp 100
[SPE2-bgp] ipv4-family vpnv4
[SPE2-bgp-af-vpnv4] peer 4.4.4.9 upe route-policy hope export
```

Example for configuring OSPF sham links

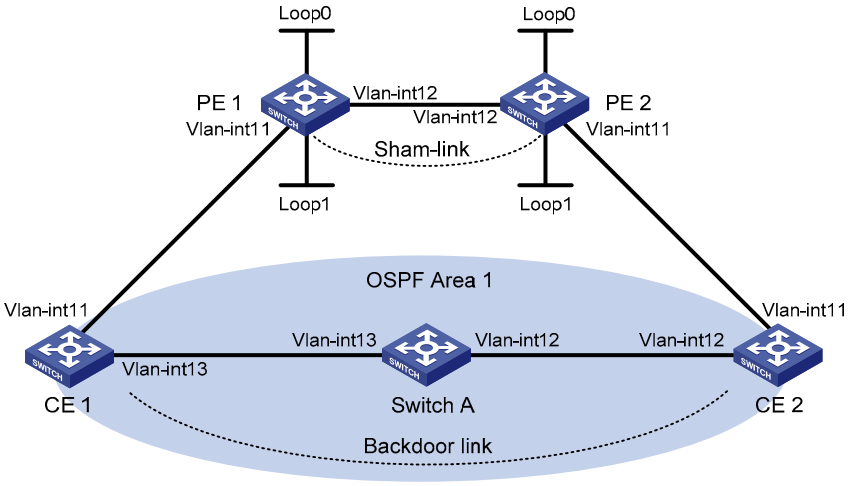
Network requirements

As shown in Figure 79:

- CE 1 and CE 2 belong to VPN 1 and are respectively 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 any route in the OSPF area.

Figure 79 Configure an OSPF sham link



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int11	100.1.1.1/24	CE 2	Vlan-int11	120.1.1.1/24
	Vlan-int13	20.1.1.1/24		Vlan-int12	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
	Vlan-int11	100.1.1.2/24		Vlan-int11	120.1.1.2/24
	Vlan-int12	10.1.1.1/24		Vlan-int12	10.1.1.2/24
Switch A	Vlan-int11	20.1.1.2/24			
	Vlan-int12	30.1.1.1/24			

Procedure

1. Configure OSPF on the customer networks

Configure conventional OSPF on CE 1, Switch A, and CE 2 to advertise segment addresses of the interfaces as shown in Figure 79. The detailed configuration steps are omitted.

After completing the configurations, CE 1 and CE 2 can learn the OSPF route to the VLAN interface 1 of each other. The following uses CE 1 as an example:

```
<CE1> display ip routing-table
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask  Proto  Pre  Cost      NextHop         Interface
20.1.1.0/24       Direct 0    0         20.1.1.1        Vlan13
20.1.1.1/32       Direct 0    0         127.0.0.1       InLoop0
20.1.1.2/32       Direct 0    0         20.1.1.2        Vlan13
30.1.1.0/24       OSPF   10   3124      20.1.1.2        Vlan13
100.1.1.0/24      Direct 0    0         100.1.1.1       Vlan11
100.1.1.1/32      Direct 0    0         127.0.0.1       InLoop0
120.1.1.0/24      OSPF   10   3125      20.1.1.2        Vlan13
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
```

2. Configure MPLS L3VPN on the backbone

Configure MPLS basic capability 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
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
[PE1] interface vlan-interface 12
[PE1-Vlan-interface12] ip address 10.1.1.1 24
[PE1-Vlan-interface12] mpls
[PE1-Vlan-interface12] mpls ldp
[PE1-Vlan-interface12] quit
```

Configure PE 1 to take PE 2 as the 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] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 2.2.2.9 enable
[PE1-bgp-af-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 MPLS basic capability 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
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface vlan-interface 12
[PE2-Vlan-interface12] ip address 10.1.1.2 24
[PE2-Vlan-interface12] mpls
[PE2-Vlan-interface12] mpls ldp
[PE2-Vlan-interface12] quit
```

Configure PE 2 to take PE 1 as the 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] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 1.1.1.9 enable
[PE2-bgp-af-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 PEs to allow CEs to access the network

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 vlan-interface 11
[PE1-Vlan-interface11] ip binding vpn-instance vpn1
[PE1-Vlan-interface11] ip address 100.1.1.2 24
[PE1-Vlan-interface11] 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
[PE2] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import-route ospf 100
[PE1-bgp-vpn1] import-route direct
[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 vlan-interface 11
[PE2-Vlan-interface11] ip binding vpn-instance vpn1
[PE2-Vlan-interface11] ip address 120.1.1.2 24
[PE2-Vlan-interface11] 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] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] import-route ospf 100
[PE2-bgp-vpn1] import-route direct
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

After completing the configurations, Issue **display ip routing-table vpn-instance** on the PEs. You can see that the path to the peer CE is along the OSPF route across the customer networks, instead of the BGP route across the backbone. Take PE 1 as an example:

```
[PE1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 5        Routes : 5
Destination/Mask  Proto  Pre  Cost      NextHop         Interface
20.1.1.0/24       OSPF   10   1563      100.1.1.1       Vlan11
30.1.1.0/24       OSPF   10   3125      100.1.1.1       Vlan11
100.1.1.0/24      Direct 0    0         100.1.1.2       Vlan11
100.1.1.2/32      Direct 0    0         127.0.0.1       InLoop0
120.1.1.0/24      OSPF   10   3126      100.1.1.1       Vlan11
```

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
```

After completing the configurations, issue **display ip routing-table vpn-instance** again on the PEs. You can see that 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. Take PE 1 as an example:

```
[PE1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 6        Routes : 6
Destination/Mask  Proto  Pre  Cost      NextHop         Interface
3.3.3.3/32        Direct 0    0         127.0.0.1       InLoop0
5.5.5.5/32        BGP    255  0         2.2.2.9         NULL0
20.1.1.0/24       OSPF   10   1563      100.1.1.1       Vlan11
100.1.1.0/24      Direct 0    0         100.1.1.2       Vlan11
100.1.1.2/32      Direct 0    0         127.0.0.1       InLoop0
120.1.1.0/24      BGP    255  0         2.2.2.9         NULL0
```

Issuing **display ip routing-table** on the CEs, you can see that the cost of the OSPF route to the peer CE is now 10 (the cost configured for the sham link), and that the next hop is now the VLAN interface 11 connected to the PE. This means that VPN traffic to the peer is forwarded over the backbone. Take CE 1 as an example:

```
[CE1] display ip routing-table
Routing Tables: Public
         Destinations : 9        Routes : 9
Destination/Mask  Proto  Pre  Cost      NextHop         Interface
20.1.1.0/24       Direct 0    0         20.1.1.1        Vlan13
20.1.1.1/32       Direct 0    0         127.0.0.1       InLoop0
20.1.1.2/32       Direct 0    0         20.1.1.2        Vlan13
30.1.1.0/24       OSPF   10   1574      100.1.1.2       Vlan11
100.1.1.0/24      Direct 0    0         100.1.1.1       Vlan11
100.1.1.1/32      Direct 0    0         127.0.0.1       InLoop0
120.1.1.0/24      OSPF   10   12        100.1.1.2       Vlan11
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
```

Issuing **display ospf sham-link** on the PEs, you can see the established sham link. Take PE 1 as an example:

```
[PE1] display ospf sham-link

          OSPF Process 100 with Router ID 100.1.1.2
 Sham Link:
 Area            NeighborId      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
```

Issuing **display ospf sham-link area**, you can see that the status of the peer is Full:

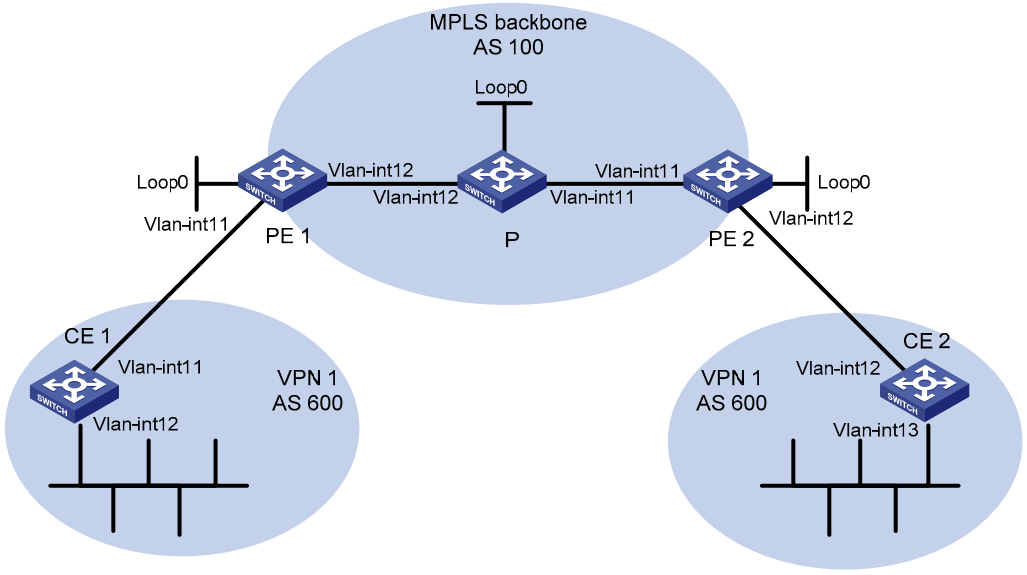
```
[PE1] display ospf sham-link area 1

          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 10, Dead 40, Retransmit 5, Transmit Delay 1
```

Example for configuring BGP AS number substitution

Network requirements

As shown in Figure 80, CE 1 and CE 2 belong to VPN 1 and are connected to PE 1 and PE 2 respectively. In addition, they use the same AS number 600.

Figure 80 Configure BGP AS number substitution



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int11	10.1.1.1/24	P	Loop0	2.2.2.9/32
	Vlan-int12	100.1.1.1/24		Vlan-int11	30.1.1.1/24
PE 1	Loop0	1.1.1.9/32		Vlan-int12	20.1.1.2/24
	Vlan-int11	10.1.1.2/24	PE 2	Loop0	3.3.3.9/32
	Vlan-int12	20.1.1.1/24		Vlan-int11	30.1.1.2/24
CE 2	Vlan-int12	10.2.1.1/24		Vlan-int12	10.2.1.2/24
	Vlan-int13	200.1.1.1/24			

Procedure

1. Configuring 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 MPLS basic capability 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 to inject routes of CEs into PEs.

After completing the configurations, if you issue **display ip routing-table** on CE 2, you can see that CE 2 has learned the route to network segment 10.1.1.0/24, where the interface used by CE 1 to access PE 1

resides; but has not learned the route to the VPN (100.1.1.0/24) behind CE 1. CE 1 has the similar situation.

```
<CE2> display ip routing-table
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto  Pre  Cost       NextHop         Interface
10.1.1.0/24         BGP    255  0          10.2.1.2        Vlan11
10.1.1.1/32         BGP    255  0          10.2.1.2        Vlan11
10.2.1.0/24         Direct 0    0          10.2.1.1        Vlan11
10.2.1.1/32         Direct 0    0          127.0.0.1       InLoop0
10.2.1.2/32         Direct 0    0          10.2.1.2        Vlan11
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
200.1.1.0/24        Direct 0    0          200.1.1.1       InLoop0
200.1.1.1/32        Direct 0    0          127.0.0.1       InLoop0
```

Issuing **display ip routing-table vpn-instance** 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
Routing Tables: vpn1
         Destinations : 7        Routes : 7
Destination/Mask    Proto  Pre  Cost       NextHop         Interface
10.1.1.0/24         BGP    255  0          1.1.1.9         NULL0
10.1.1.1/32         BGP    255  0          1.1.1.9         NULL0
10.2.1.0/24         Direct 0    0          10.2.1.2        Vlan11
10.2.1.1/32         Direct 0    0          10.2.1.1        Vlan11
10.2.1.2/32         Direct 0    0          127.0.0.1       InLoop0
100.1.1.1/32        BGP    255  0          1.1.1.9         NULL0
200.1.1.1/32        BGP    255  0          10.2.1.1        Vlan11
```

Enabling BGP update packet debugging on PE 2, you can see that PE 2 advertises the route to 100.1.1.1/32, and the AS_PATH is 100 600.

```
<PE2> terminal monitor
<PE2> terminal debugging
<PE2> debugging bgp update vpn-instance vpn1 verbose
<PE2> refresh bgp vpn-instance vpn1 all export
*0.4402392 PE2 RM/7/RMDEBUG:
         BGP.vpn1: Send UPDATE to 10.2.1.1 for following destinations :
         Origin    : Incomplete
         AS Path   : 100 600
         Next Hop  : 10.2.1.2
         100.1.1.1/32,
```

Issuing **display bgp routing-table peer received-routes** on CE 2, you can see that CE 2 did not receive the route to 100.1.1.1/32.

```
<CE2> display bgp routing-table peer 10.2.1.2 received-routes
 Total Number of Routes: 4
 BGP Local router ID is 10.2.1.1
 Status codes: * - valid, ^ - VPNv4 best, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
```

```
              Origin : i - IGP, e - EGP, ? - incomplete
     Network          NextHop        MED     LocPrf    PrefVal Path/Ogn
 *>  10.1.1.0/24      10.2.1.2                           0       100?
 *>  10.1.1.1/32      10.2.1.2                           0       100?
 *   10.2.1.0/24      10.2.1.2        0                  0       100?
 *   10.2.1.1/32      10.2.1.2        0                  0       100?
```

2. Configure BGP AS number substitution

Configure BGP AS number substitution on PE 2.

```
<PE2> system-view
[PE2] bgp 100
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] peer 10.2.1.1 substitute-as
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

The following output shows that among the routes advertised by PE 2 to CE 2, the AS_PATH of 100.1.1.1/32 has changed from 100 600 to 100 100:

```
*0.13498737 PE2 RM/7/RMDEBUG:
        BGP.vpn1: Send UPDATE to 10.2.1.1 for following destinations :
        Origin    : Incomplete
        AS Path   : 100 100
        Next Hop  : 10.2.1.2
        100.1.1.1/32
```

Display again the routing information that CE 2 receives and the routing table:

```
<CE2> display bgp routing-table peer 10.2.1.2 received-routes
 Total Number of Routes: 5
 BGP Local router ID is 10.2.1.1
 Status codes: * - valid, ^ - VPNv4 best, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
     Network          NextHop       MED      LocPrf    PrefVal Path/Ogn
 *>  10.1.1.0/24      10.2.1.2                           0       100?
 *>  10.1.1.1/32      10.2.1.2                           0       100?
 *   10.2.1.0/24      10.2.1.2       0                   0       100?
 *   10.2.1.1/32      10.2.1.2       0                   0       100?
 *>  100.1.1.1/32     10.2.1.2                           0       100 100?
<CE2> display ip routing-table
Routing Tables: Public
        Destinations : 9        Routes : 9
Destination/Mask    Proto  Pre  Cost       NextHop         Interface
10.1.1.0/24         BGP    255  0          10.2.1.2        Vlan12
10.1.1.1/32         BGP    255  0          10.2.1.2        Vlan12
10.2.1.0/24         Direct 0    0          10.2.1.1        Vlan12
10.2.1.1/32         Direct 0    0          127.0.0.1       InLoop0
10.2.1.2/32         Direct 0    0          10.2.1.2        Vlan12
100.1.1.1/32        BGP    255  0          10.2.1.2        Vlan12
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
```

```
200.1.1.1/32        Direct 0    0          127.0.0.1       InLoop0
```

After you also configure BGP AS substitution on PE 1, the VLAN interfaces of CE 1 and CE 2 can ping each other:

```
<CE1> ping -a 100.1.1.1 200.1.1.1
  PING 200.1.1.1: 56  data bytes, press CTRL_C to break
    Reply from 200.1.1.1: bytes=56 Sequence=1 ttl=253 time=109 ms
    Reply from 200.1.1.1: bytes=56 Sequence=2 ttl=253 time=67 ms
    Reply from 200.1.1.1: bytes=56 Sequence=3 ttl=253 time=66 ms
    Reply from 200.1.1.1: bytes=56 Sequence=4 ttl=253 time=85 ms
    Reply from 200.1.1.1: bytes=56 Sequence=5 ttl=253 time=70 ms
  --- 200.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 66/79/109 ms
```

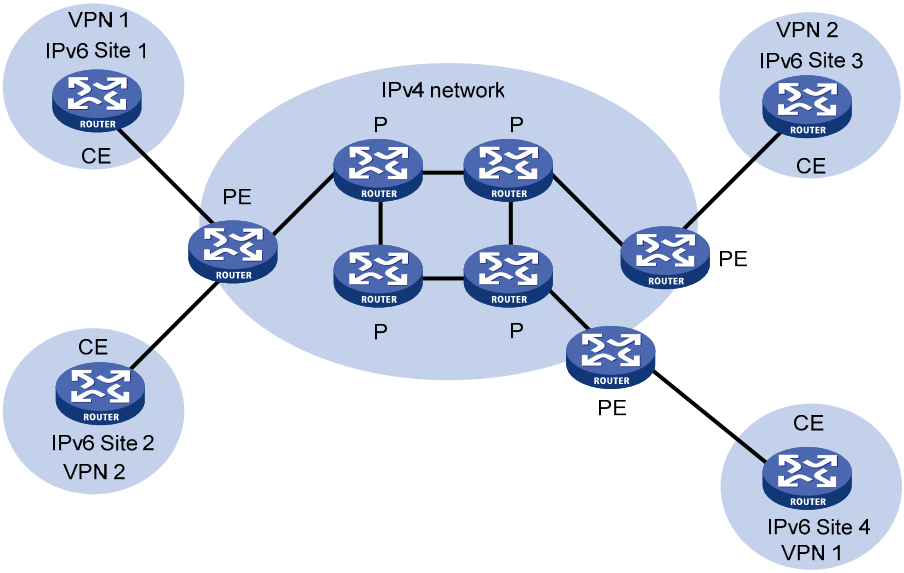
Configuring IPv6 MPLS L3VPN

MPLS L3VPN applies to the IPv4 environment. It uses BGP to advertise IPv4 VPN routes and uses MPLS to forward IPv4 VPN packets on the service provider backbone.

IPv6 MPLS L3VPN functions similarly. It uses BGP to advertise IPv6 VPN routes and uses MPLS to forward IPv6 VPN packets on the service provider backbone.

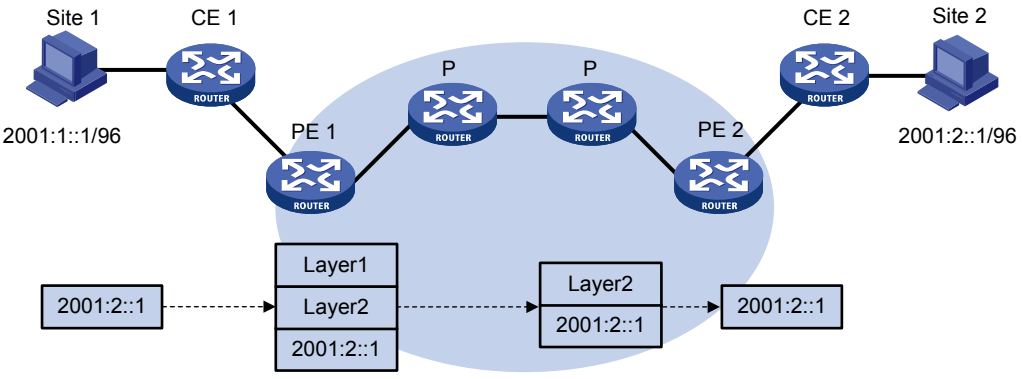
Figure 81 shows the typical IPv6 MPLS L3VPN model. At present, the service provider backbone in the IPv6 MPLS L3VPN model is an IPv4 network. IPv6 runs inside the VPNs and between CEs and PEs. Therefore, PEs must support both IPv4 and IPv6. The PE-CE interfaces of a PE run IPv6 and the PE-P interface of a PE runs IPv4.

Figure 81 Network diagram for the IPv6 MPLS L3VPN model



IPv6 MPLS L3VPN packet forwarding

Figure 82 IPv6 MPLS L3VPN packet forwarding diagram



As shown in Figure 82, 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 searches the routing table of the VPN instance. Finding a matching entry, PE 1 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 the interface to CE 2.
5. CE 2 forwards the packet to the destination by IPv6 forwarding.

IPv6 MPLS L3VPN routing information advertisement

The IPv6 VPN routing information of a local CE is advertised to a remote peer PE in three steps:

1. From the local CE to the ingress PE.
2. From the ingress PE to the egress PE.
3. From the egress PE to the remote peer CE.

Then, a route is available from the local CE to the remote CE.

Routing information exchange from the local CE to the ingress PE

After establishing an adjacency with the directly connected PE, a CE advertises its IPv6 VPN routes to the PE.

The routes between a CE and a PE can be static routes, RIPng routes, OSPFv3 routes, IPv6 IS-IS routes, or eBGP routes. No matter which routing protocol is used, the CE always advertises standard IPv6 routes to the PE.

Routing information exchange from the ingress PE to the egress PE

After learning the IPv6 VPN routes from the CE, the ingress PE adds RDs and VPN targets for these standard IPv6 routes to create VPN-IPv6 routes, saves them to the routing table of the VPN instance created for the CE, and then triggers MPLS to assign VPN labels for them.

Then, the ingress PE advertises the VPN-IPv6 routes to the egress PE through MP-BGP.

Finally, 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.

Routing information exchange from the egress PE to the remote CE

The exchange of routing information between the egress PE and the remote CE is the same as that between the local CE and the ingress PE.

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

Complete the following tasks to configure IPv6 MPLS L3VPN:

Task	Remarks
Configuring basic IPv6 MPLS L3VPN	By configuring basic IPv6 MPLS L3VPN, you can construct simple IPv6 VPN networks over an MPLS backbone. To deploy special IPv6 MPLS L3VPN networks, such as inter-AS VPN, you also must perform some specific configurations in addition to the basic IPv6 MPLS L3VPN configuration. See related sections for details.
Configuring inter-AS IPv6 VPN	

Configuring basic IPv6 MPLS L3VPN

Basic IPv6 MPLS L3VPN configuration task list

The key task in IPv6 MPLS L3VPN configuration is to manage the advertisement of IPv6 VPN routes on the MPLS backbone, including PE-CE route exchange and PE-PE route exchange.

Complete the following tasks to configure basic IPv6 MPLS L3VPN:

Task		Remarks
Configuring VPN instances	Creating a VPN instance	Required.
	Associating a VPN instance with an interface	Required.
	Configuring route related attributes for a VPN instance	Optional.
	Configuring a tunneling policy for a VPN instance	Optional.
	Configuring an LDP instance	Optional.
Configuring routing between PE and CE		Required.
Configuring routing between PEs		Required.
Configuring routing features for the BGP-VPNv6 subaddress family		Optional.

Prerequisites

Before configuring basic IPv6 MPLS L3VPN, complete these tasks:

- Configuring an IGP for the MPLS backbone (on the PEs and Ps) to achieve IP connectivity
- Configuring the MPLS basic capability for the MPLS backbone
- Configuring MPLS LDP for the MPLS backbone so that LDP LSPs can be established

Configuring VPN instances

By configuring VPN instances on a PE, you isolate not only VPN routes from public network routes, but also routes of a VPN from those of another VPN. 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.

A VPN instance only takes effect after you configure an RD for it.

You can configure a description for a VPN instance to record its related information, such as its relationship with a certain VPN.

To create and configure a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a VPN instance and enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required.
3. Configure an RD for the VPN instance.	**route-distinguisher** *route-distinguisher*	Required.
4. Configure a description for the VPN instance.	**description** *text*	Optional.

Associating a VPN instance with an interface

After creating and configuring a VPN instance, you must associate the VPN instance with the interface for connecting the CE. Any LDP-capable interface can be associated with a VPN instance. For information about LDP-capable interfaces, see "MPLS basics configuration."

To associate a VPN instance with an interface:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter interface view.	**interface** *interface-type interface-number*	—
3. Associate a VPN instance with the interface.	**ip binding vpn-instance** *vpn-instance-name*	Required. No VPN instance is associated with an interface by default.

Ip binding vpn-instance clears the IPv6 address of the interface. Be sure to re-configure an IPv6 address for the interface after configuring the command.

Configuring route related attributes for a VPN instance

The control process of VPN route advertisement is as follows:

- When a VPN route learned from a CE gets redistributed into BGP, BGP associates it with a VPN 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 VPN target.
- The VPN instance determines how to change the VPN targets attributes for routes to be advertised according to the **export-extcommunity** in the VPN target.

To configure route related attributes for a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	—
3. Enter IPv6 VPN view.	**ipv6-family**	Optional.
4. Configure VPN targets.	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	Required.
5. Set the maximum number of routes supported.	**routing-table limit** *number* { *warn-threshold* | **simply-alert** }	Optional.
6. Apply an import routing policy.	**import route-policy** *route-policy*	Optional. By default, all routes matching the import target attribute are accepted.
7. Apply an export routing policy.	**export route-policy** *route-policy*	Optional. By default, routes to be advertised are not filtered.

Route related attributes configured in VPN instance view are applicable to both IPv4 VPNs and IPv6 VPNs.

You can configure route related attributes for IPv6 VPNs in both VPN instance view and IPv6 VPN view. Those configured in IPv6 VPN view take precedence.

A single **vpn-target** command can configure up to eight VPN targets. You can configure up to 64 VPN targets for a VPN instance.

You can define the maximum number of routes for a VPN instance to support, preventing too many routes from being redistributed into the PE. The maximum number of routes supported by a PE varies by device.

Create a routing policy before associating it with a VPN instance so that the device can filter the routes to be received and advertised.

Configuring a tunneling policy for a VPN instance

When multiple tunnels exist in a MPLS L3VPN network, you can configure a tunneling policy to specify the type and number of tunnels to be used by using **tunnel select-seq** or the **preferred-path**.

With **tunnel select-seq**, you can specify the tunnel selection preference order and the number of tunnels for load balancing.

With **preferred-path**, you can configure preferred tunnels that each corresponds to a tunnel interface.

After a tunneling policy is applied on a PE, the PE selects tunnels in this order:

- The PE matches the peer PE address against the destination addresses of preferred tunnels, starting from the tunnel with the smallest number. If no match is found, the local PE selects tunnels as configured by **tunnel select-seq** or the default tunneling policy.

- If a matching tunnel is found and the tunnel is available, the local PE stops matching other tunnels and forwards the traffic to the specified tunnel interface.

If the matching tunnel is unavailable (for example, the tunnel is down or the tunnel's ACL does not permit the traffic) and is not specified with the **disable-fallback** keyword, the local PE continues to match other preferred tunnels; if the tunnel is specified with the **disable-fallback** keyword, the local PE stops matching and tunnel selection fails.

To configure a tunneling policy for a VPN instance:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a tunneling policy and enter tunneling policy view.	**tunnel-policy** *tunnel-policy-name*	Required.
3. Configure a preferred tunnel and specify a tunnel interface for it.	**preferred-path** *number* **interface tunnel** *tunnel-number* [**disable-fallback**]	Optional. Not configured by defautl.
4. Specify the tunnel selection preference order and the number of tunnels for load balancing.	**tunnel select-seq** { **cr-lsp** | **lsp** } * **load-balance-number** *number*	Optional. By default, only one tunnel is selected (no load balancing) in this order: LSP tunnel, CR-LSP tunnel.
5. Return to system view.	**quit**	—
6. Enter VPN instance view.	**ip vpn-instance** *vpn-instance-name*	Required.
7. Enter IPv6 VPN view.	**ipv6-family**	Optional.
8. Apply the tunneling policy to the VPN instance	**tnl-policy** *tunnel-policy-name*	Required. By default, only one tunnel is selected (no load balancing) in this order: LSP tunnel, CR-LSP tunnel.

In a tunneling policy, you can configure up to 64 preferred tunnels.

The tunnel interfaces specified for the preferred tunnels can have the same destination address and the tunnel encapsulation type must be MPLS TE.

If multiple types of tunnels are configured and the number of tunnels for load balancing is greater than the number of each type of tunnels, the tunnels of different types are used for laod balancing.

A tunneling policy configured in VPN instance view applies to both IPv4 VPNs and IPv6 VPNs.

You can configure a tunneling policy for IPv6 VPNs in both VPN instance view and IPv6 VPN view. A tunneling policy configured in IPv6 VPN view takes precedence.

Create a tunneling policy before associating it with a VPN instance. Otherwise, the default tunneling policy is used. The default tunneling policy selects only one tunnel in this order: LSP tunnel, CR-LSP tunnel.

Configuring an LDP instance

LDP instances are for carrier's carrier network applications.

This task is to enable LDP for an existing VPN instance, create an LDP instance for the VPN instance, and configure LDP parameters for the LDP instance.

For LDP instance configuration information, see "MPLS L3VPN configuration."

Configuring routing between PE and CE

You can configure static routing, RIPng, OSPFv3, IPv6 IS-IS, or eBGP between PE and CE.

Configuration prerequisites

Before configuring routing between PE and CE, complete these tasks:

- Assign an IPv6 address to the CE-PE interface of the CE
- Assign an IPv6 address to the PE-CE interface of the PE

Configuring static routing between PE and CE

To configure static routing between PE and CE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Configure static routes for a specified VPN instance.	**ipv6 route-static** *ipv6-address prefix-length* { *interface-type interface-number* [*next-hop-address*] | *next-hop-address* | **vpn-instance** *d-vpn-instance-name nexthop-address* } [**preference** *preference-value*]	Required. Use either command. Perform this configuration on PEs. On CEs, configure normal static routes.
	ipv6 route-static vpn-instance *s-vpn-instance-name&<1-6> ipv6-address prefix-length* { *interface-type interface-number* [*next-hop-address*] | *nexthop-address* [**public**] | **vpn-instance** *d-vpn-instance-name nexthop-address* } [**preference** *preference-value*]	

For information about IPv6 static routing, see *Layer 3—IP Routing Configuration Guide*.

Configuring RIPng between PE and 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.

To configure RIPng between PE and CE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create a RIPng process for a VPN instance and enter RIPng view.	**ripng** [*process-id*] **vpn-instance** *vpn-instance-name*	Required. Perform this configuration on PEs. On CEs, create a normal RIPng process.
3. Return to system view.	**quit**	—
4. Enter interface view.	**interface** *interface-type interface-number*	—
5. Enable RIPng on the interface.	**ripng** *process-id* **enable**	Required. By default, RIPng is disabled on an interface.

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

Configuring OSPFv3 between PE and 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.

To configure OSPFv3 between PE and CE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create an OSPFv3 process for a VPN instance and enter the OSPFv3 view.	**ospfv3** [*process-id*] **vpn-instance** *vpn-instance-name*	Required. Perform this configuration on PEs. On CEs, create a normal OSPF process.
3. Set the router ID.	**router-id** *router-id*	Required.
4. Return to system view.	**quit**	—
5. Enter interface view.	**interface** *interface-type interface-number*	—
6. Enable OSPFv3 on the interface.	**ospfv3** *process-id* **area** *area-id* [**instance** *instance-id*]	Required. By default, OSPFv3 is disabled on an interface. Perform this configuration on PEs.

For more information about OSPFv3, see *Layer 3—IP Routing Configuration Guide*.

Configuring IPv6 IS-IS between PE and 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.

To configure IPv6 IS-IS between PE and CE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Create an IPv6 IS-IS process for a VPN instance and enter IS-IS view.	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	Required. Perform this configuration on PEs. On CEs, create a normal IPv6 IS-IS process.
3. Configure a network entity title for the IS-IS process.	**network-entity** *net*	Required. Not configured by default.
4. Enable the IPv6 capacity for the IS-IS process.	**ipv6 enable**	Required. Disabled by default.
5. Return to system view.	**quit**	—
6. Enter interface view.	**interface** *interface-type interface-number*	—
7. Enable the IPv6 capacity for the IS-IS process on the interface.	**isis ipv6 enable** [*process-id*]	Required. Disabled by default.

For more information about IPv6 IS-IS, see *Layer 3—IP Routing Configuration Guide*.

Configuring eBGP between PE and CE

1. Configurations on a PE

To configure eBGP between PE and CE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enable BGP and enter BGP view.	**bgp** *as-number*	—
3. Enter IPv6 BGP-VPN instance view.	**ipv6-family vpn-instance** *vpn-instance-name*	Required.
4. Configure the CE as the VPN eBGP peer.	**peer** *ipv6-address* **as-number** *as-number*	Required.
5. Redistribute the routes of the local CEs.	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	Required. A PE needs to redistribute the routes of the local CEs into its VPN routing table so that it can advertise them to the peer PE.
6. Configure a filtering policy to filter the routes to be advertised.	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **export** [**direct** | **isisv6** *process-id* | **ripng** *process-id* | **static**]	Optional. By default, BGP does not filter routes to be advertised.
7. Configure a filtering policy to filter received routes.	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **import**	Optional. By default, the PE does not filter received routes.

2. Configurations on a CE

To configure PE-CE route exchange through eBGP on a CE:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Enter IPv6 BGP subaddress family view.	**ipv6-family**	Required.
4. Configure the PE as the eBGP peer.	**peer** *ipv6-address* **as-number** *as-number*	Required.
5. Configure route redistribution and advertisement.	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	Optional. A CE needs to advertise its VPN routes to the connected PE so that the PE can advertise them to the peer CE.

After an IPv6 BGP-VPN instance is configured, exchange of BGP routes for the VPN instance is the same as exchange of ordinary BGP routes.

The configuration commands available in IPv6 BGP-VPN instance view are the same as those in IPv6 BGP subaddress family view. For more configuration commands in the two views, see *Layer 3—IP Routing Configuration Guide*.

Configuring routing between PEs

To configure routing between PEs:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	Required.
3. Configure the remote PE as the peer.	**peer** *ip-address* **as-number** *as-number*	Required.
4. Specify the source interface for route update packets.	**peer** { *group-name* | *ip-address* } **connect-interface** *interface-type interface-number*	Required. By default, BGP uses the outbound interface of the best route to the BGP peer.
5. Enter BGP-VPNv6 subaddress family view.	**ipv6-family vpnv6**	Required.
6. Enable the exchange of BGP-VPNv6 routing information with the specified peer.	**peer** *ip-address* **enable**	Required. By default, BGP peers exchange only IPv4 routing information.

Configuring routing features for the BGP-VPNv6 subaddress family

A variety of routing features for the BGP-VPNv6 subaddress family are the same as those for BGP IPv6 unicast routing. You can select any of the features as required.

To configure routing features for the BGP-VPNv6 subaddress family:

Step	Command	Remarks
1. Enter system view.	**system-view**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Configure the remote PE as the peer.	**peer** *ip-address* **as-number** *as-number*	Required.
4. Specify the interface for TCP connections.	**peer** *ip-address* **connect-interface** *interface-type interface-number*	Required.
5. Enter BGP-VPNv6 subaddress family view.	**ipv6-family vpnv6**	—
6. Set the default value of the local preference.	**default local-preference** *value*	Optional. 100 by default.
7. Set the default value for the system MED.	**default med** *med-value*	Optional. By default, the default value of the system MED is 0.
8. Configure a filtering policy to filter routes to be advertised.	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **export** [**direct** | **isisv6** *process-id* | **ripng** *process-id* | **static**]	Optional. By default, the PE does not filter routes to be advertised.

Step	Command	Remarks
9. Configure a filtering policy to filter received routes.	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **import**	Optional. By default, the PE does not filter received routes.
10. Apply a filtering policy for the peer.	**peer** *ip-address* **filter-policy** *acl6-number* { **export** | **import** }	Optional. By default, no filtering policy is applied for a peer.
11. Apply an IPv6-prefix list for the peer to filter received/advertised routes.	**peer** *ip-address* **ipv6-prefix** *prefix-name* { **export** | **import** }	Optional. By default, no IPv6 prefix list is applied for a peer.
12. Specify the preference value for the routes received from the peer.	**peer** *ip-address* **preferred-value** *value*	Optional. 0 by default.
13. Configure BGP updates to the peer to not carry private AS numbers.	**peer** *ip-address* **public-as-only**	Optional. By default, a BGP update carries private AS numbers.
14. Apply a routing policy for the peer.	**peer** *ip-address* **route-policy** *route-policy-name* { **export** | **import** }	Optional. By default, no routing policy is applied for a peer.
15. Enable VPN target filtering for received BGP-VPNv6 subaddress family routes.	**policy vpn-target**	Optional. Enabled by default.
16. Configure the local PE as the route reflector and specify the peer as the client.	**peer** *ip-address* **reflect-client**	Optional. No route reflector or client is configured by default.
17. Enable route reflection between clients.	**reflect between-clients**	Optional. Enabled by default.
18. Configure a cluster ID for the route reflector.	**reflector cluster-id** { *cluster-id* | *ip-address* }	Optional. By default, a route reflector uses its router ID as the cluster ID.
19. Create an RR reflection policy.	**rr-filter** *extended-community-list-number*	Optional.

For information about IPv6 BGP routing features, see *Layer 3—IP Routing Configuration Guide*.

Configuring inter-AS IPv6 VPN

If the MPLS backbone that carries the IPv6 VPN routes spans multiple ASs, you must configure inter-AS IPv6 VPN.

There are three inter-AS VPN solutions (see "MPLS L3VPN configuration" for more information). IPv6 MPLS L3VPN supports only inter-AS VPN option A and option C.

Prerequisites

Before configuring inter-AS IPv6 VPN, complete these tasks:

- Configuring an IGP for the MPLS backbone in each AS to ensure IP connectivity
- Configuring basic MPLS for the MPLS backbone of each AS
- Configuring 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. It is easy to implement.

To configure inter-AS IPv6 option A:

- Perform basic IPv6 MPLS L3VPN configuration on each AS.
- Configure each ASBR, taking the peer ASBR PE as its CE. In other words, configure VPN instances on both PEs and ASBR PEs. The VPN instances on PEs allow CEs to access the network, while those on ASBR PEs are for access of the peer ASBR PEs.

For configuration information, see "Configuring basic IPv6 MPLS L3VPN."

In the inter-AS IPv6 VPN option A solution, for the same IPv6 VPN, the VPN 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.

Configuring inter-AS IPv6 VPN option C

Configuring the PEs

You must establish ordinary iBGP peer relationships between PEs and ASBR PEs in an AS and MP-eBGP peer relationships 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**	—
2. Enter BGP view.	**bgp** *as-number*	—
3. Configure the ASBR PE in the same AS as the iBGP peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
4. Enable the PE to exchange labeled routes with the ASBR PE in the same AS.	**peer** { *group-name* | *ip-address* } **label-route-capability**	Required. By default, the PE does not advertise labeled routes to the IPv4 peer/peer group.
5. Configure the PE of another AS as the eBGP peer.	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	Required.
6. Enter BGP-VPNv6 subaddress family view.	**ipv6-family vpnv6**	—
7. Enable the PE to exchange BGP VPNv6 routing information with the eBGP peer.	**peer** *ip-address* **enable**	Required.

Configuring the ASBR PEs

In the inter-AS IPv6 VPN option C solution, an inter-AS LSP is required, and the routes advertised between the relevant 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 inter-AS VPN option C.

Configuring the routing policy

After you configure and apply a routing policy on an ASBR PE, it:

- 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 inter-AS VPN option C.

Displaying and maintaining IPv6 MPLS L3VPN

Resetting BGP connections

When BGP configuration changes, use the soft reset function or reset BGP connections to make the changes take effect. Soft reset requires that BGP peers have the route refreshment capability, which means supporting Route-Refresh messages.

Task	Command	Remarks
Soft reset the IPv6 BGP connections of a VPN instance.	**refresh bgp ipv6 vpn-instance** *vpn-instance-name* { *ipv6-address* | **all** | **external** } { **export** | **import** }	Available in user view
Soft reset the BGP VPNv6 connections.	**refresh bgp vpnv6** { *ip-address* | **all** | **external** | **internal** } { **export** | **import** }	Available in user view
Reset the IPv6 BGP connections of a VPN instance.	**reset bgp ipv6 vpn-instance** *vpn-instance-name* { *as-number* | *ipv6-address* | **all** | **external** }	Available in user view
Reset BGP VPNv6 connections.	**reset bgp vpnv6** { *as-number* | *ip-address* | **all** | **external** | **internal** }	Available in user view

Displaying information about IPv6 MPLS L3VPN

Task	Command	Remarks
Display information about the IPv6 routing table associated with a VPN instance.	**display ipv6 routing-table vpn-instance** *vpn-instance-name* [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about a specified or all VPN instances.	**display ip vpn-instance** [**instance-name** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

Task	Command	Remarks
Display information about the IPv6 FIB of a VPN instance.	**display ipv6 fib vpn-instance** *vpn-instance-name* [**acl6** *acl6-number* | **ipv6-prefix** *ipv6-prefix-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display a VPN instance's FIB entries that match the specified destination IPv6 address.	**display ipv6 fib vpn-instance** *vpn-instance-name ipv6-address* [*prefix-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about BGP VPNv6 peers established between PEs.	**display bgp vpnv6 all peer** [*ipv4-address* **verbose** | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display information about IPv6 BGP peers established between the PE and CE in a VPN instance.	**display bgp vpnv6 vpn-instance** *vpn-instance-name* **peer** [*ipv6-address* **verbose** | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display all BGP VPNv6 routing information.	**display bgp vpnv6 all routing-table** [*network-address prefix-length* [**longer-prefixes**] | **peer** *ip-address* { **advertised-routes** | **received-routes** } [**statistic**] | **statistic**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display the BGP VPNv6 routing information of a specified RD.	**display bgp vpnv6 route-distinguisher** *route-distinguisher* **routing-table** [*network-address prefix-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view
Display the BGP VPNv6 routing information of a specified VPN instance.	**display bgp vpnv6 vpn-instance** *vpn-instance-name* **routing-table** [*network-address prefix-length* [**longer-prefixes**] | **peer** *ipv6-address* { **advertised-routes** | **received-routes** }] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	Available in any view

For commands that display information about a routing table, see *Layer 3—IP Routing Command Reference*.

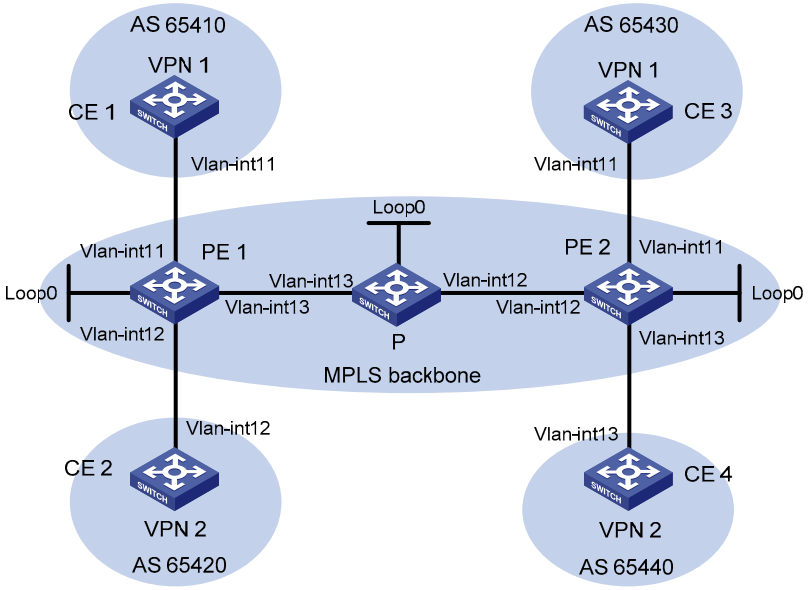
IPv6 MPLS L3VPN configuration examples

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 VPN target attributes 111:1. VPN 2 uses VPN target attributes 222:2. Users of different VPNs cannot access each other.
- eBGP is used to exchange VPN routing information between CE and PE switches.
- PEs use OSPF to communicate with each other and use MP-iBGP to exchange VPN routing information.

Figure 83 Configure IPv6 MPLS L3VPNs



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int11	2001:1::1/64	P	Loop0	2.2.2.9/32
PE 1	Loop0	1.1.1.9/32		Vlan-int12	172.2.1.1/24
	Vlan-int11	2001:1::2/64		Vlan-int13	172.1.1.2/24
	Vlan-int13	172.1.1.1/24	PE 2	Loop0	3.3.3.9/32
	Vlan-int12	2001:2::2/64		Vlan-int12	172.2.1.2/24
CE 2	Vlan-int12	2001:2::1/64		Vlan-int11	2001:3::2/64
CE 3	Vlan-int11	2001:3::1/64		Vlan-int13	2001:4::2/64
CE 4	Vlan-int13	2001:4::1/64			

Procedure

1. Configure OSPF on the MPLS backbone to achieve IP connectivity among the PEs and the P switch.

Configure PE 1.

```
<PE1> system-view
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 1.1.1.9 32
[PE1-LoopBack0] quit
[PE1] interface vlan-interface 13
[PE1-Vlan-interface13] ip address 172.1.1.1 24
[PE1- Vlan-interface13] 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 switch.

```
<P> system-view
[P] interface loopback 0
```

```
[P-LoopBack0] ip address 2.2.2.9 32
[P-LoopBack0] quit
[P] interface vlan-interface 13
[P-Vlan-interface13] ip address 172.1.1.2 24
[P- Vlan-interface13] quit
[P] interface vlan-interface 12
[P-Vlan-interface12] ip address 172.2.1.1 24
[P-Vlan-interface12] 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 vlan-interface 12
[PE2-Vlan-interface12] ip address 172.2.1.2 24
[PE2-Vlan-interface12] 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 you complete the configurations, OSPF adjacency are established between PE 1, P, and PE 2. Issue **display ospf peer**. The output shows that the adjacency status is Full. Issue **display ip routing-table**. The output shows that the PEs have learned the routes to the loopback interfaces of each other. The following uses PE 1 as an example:

```
[PE1] display ip routing-table
Routing Tables: Public
         Destinations : 9        Routes : 9
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        172.1.1.2       Vlan13
3.3.3.9/32        OSPF   10   2        172.1.1.2       Vlan13
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
172.1.1.0/24      Direct 0    0        172.1.1.1       Vlan13
172.1.1.1/32      Direct 0    0        127.0.0.1       InLoop0
172.1.1.2/32      Direct 0    0        172.1.1.2       Vlan13
172.2.1.0/24      OSPF   10   1        172.1.1.2       Vlan13
[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(Vlan-interface13)'s neighbors
 Router ID: 172.1.1.2        Address: 172.1.1.2        GR State: Normal
   State: Full  Mode:Nbr is  Master  Priority: 1
   DR: None   BDR: None   MTU: 1500
   Dead timer due in 38  sec
   Neighbor is up for 00:02:44
   Authentication Sequence: [ 0 ]
   Neighbor state change count: 5
```

2. Configure the MPLS basic capability 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
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
[PE1] interface vlan-interface 13
[PE1-Vlan-interface13] mpls
[PE1-Vlan-interface13] mpls ldp
[PE1-Vlan-interface13] quit
```

Configure the P switch.

```
[P] mpls lsr-id 2.2.2.9
[P] mpls
[P-mpls] quit
[P] mpls ldp
[P-mpls-ldp] quit
[P] interface vlan-interface 13
[P-Vlan-interface13] mpls
[P-Vlan-interface13] mpls ldp
[P-Vlan-interface13] quit
[P] interface vlan-interface 12
[P-Vlan-interface12] mpls
[P-Vlan0interface12] mpls ldp
[P-Vlan-interface12] quit
```

Configure PE 2.

```
[PE2] mpls lsr-id 3.3.3.9
[PE2] mpls
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface vlan-interface 12
[PE2-Vlan-interface12] mpls
[PE2-Vlan-interface12] mpls ldp
[PE2-Vlan-interface12] quit
```

After you complete the configurations, LDP sessions are established between PE 1, P, and PE 2. Issue **display mpls ldp session**. The output shows that the session status is Operational. Issue **display mpls ldp lsp**. The output shows the LSPs established by LDP. The following uses PE 1 as an example:

```
[PE1] display mpls ldp session
               LDP Session(s) in Public Network
Total number of sessions: 1
------------------------------------------------------------------
 Peer-ID         Status        LAM  SsnRole  FT   MD5  KA-Sent/Rcv
 ------------------------------------------------------------------
 2.2.2.9:0       Operational   DU   Passive  Off  Off  5/5
 ------------------------------------------------------------------
 LAM : Label Advertisement Mode         FT  : Fault Tolerance
[PE1] display mpls ldp lsp
                             LDP LSP Information
 ------------------------------------------------------------------
 SN  DestAddress/Mask   In/OutLabel  Next-Hop     In/Out-Interface
 ------------------------------------------------------------------
 1   1.1.1.9/32         3/NULL       127.0.0.1     Vlan-interface13/InLoop0
 2   2.2.2.9/32         NULL/3       172.1.1.2     -------/Vlan-interface13
 3   3.3.3.9/32         NULL/1024    172.1.1.2     -------/Vlan-interface13
------------------------------------------------------------------
 A '*' before an LSP means the LSP is not established
 A '*' before a Label means the USCB or DSCB is stale
```

3. Configure VPN instances on the PEs to allow the CEs to 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 vlan-interface 11
[PE1-Vlan-interface11] ip binding vpn-instance vpn1
[PE1-Vlan-interface11] ipv6 address 2001:1::1 64
[PE1-Vlan-interface11] quit
[PE1] interface vlan-interface 12
[PE1-Vlan-interface12] ip binding vpn-instance vpn2
[PE1-Vlan-interface12] ipv6 address 2001:2::1 64
[PE1-Vlan-interface12] 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 vlan-interface 11
[PE2-Vlan-interface11] ip binding vpn-instance vpn1
[PE2-Vlan-interface11] ipv6 address 2001:3::1 64
[PE2-Vlan-interface11] quit
[PE2] interface vlan-interface 13
[PE2-Vlan-interface13] ip binding vpn-instance vpn2
[PE2-Vlan-interface13] ipv6 address 2001:4::1 64
[PE2-Vlan-interface13] quit
```

Configure IP addresses for the CEs as per Figure 83. The configuration steps are omitted.

After completing the configurations, issue **display ip vpn-instance** on the PEs to view 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. The following uses 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        2006/08/13 09:32:45
  vpn2                  100:2        2006/08/13 09:42:59
[PE1] ping ipv6 -vpn-instance vpn1 2001:1::1
  PING 2001:1::1 : 56  data bytes, press CTRL_C to break
    Reply from 2001:1::1
    bytes=56 Sequence=1 hop limit=64  time = 1 ms
    Reply from 2001:1::1
    bytes=56 Sequence=2 hop limit=64  time = 1 ms
    Reply from 2001:1::1
    bytes=56 Sequence=3 hop limit=64  time = 1 ms
    Reply from 2001:1::1
    bytes=56 Sequence=4 hop limit=64  time = 1 ms
    Reply from 2001:1::1
    bytes=56 Sequence=5 hop limit=64  time = 1 ms

  --- 2001:1::1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 1/1/1 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] ipv6-family
[CE1-bgp-af-ipv6] peer 2001:1::2 as-number 100
[CE1-bgp-af-ipv6] import-route direct
[CE1-bgp-af-ipv6] quit
```

The configurations for the other three CEs (CE 2 through CE 4) are similar.

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ipv6-family vpn-instance vpn1
[PE1-bgp-ipv6-vpn1] peer 2001:1::1 as-number 65410
[PE1-bgp-ipv6-vpn1] import-route direct
[PE1-bgp-ipv6-vpn1] quit
[PE1-bgp] ipv6-family vpn-instance vpn2
[PE1-bgp-ipv6-vpn2] peer 2001:2::1 as-number 65420
[PE1-bgp-ipv6-vpn2] import-route direct
[PE1-bgp-ipv6-vpn2] quit
[PE1-bgp] quit
```

The configurations for PE 2 are similar to those for PE 1.

After completing the configurations, issue **display bgp vpnv6 vpn-instance peer** on the PEs. The output shows that BGP peer relationship has been established between PE and CE switches, and has reached the Established state. The following uses the PE 1-CE 1 BGP peer relationship as an example:

```
[PE1] display bgp vpnv6 vpn-instance vpn1 peer
 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     11        9     0        1   00:06:37   Established
```

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] ipv6-family vpnv6
[PE1-bgp-af-vpnv6] peer 3.3.3.9 enable
[PE1-bgp-af-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] ipv6-family vpnv6
[PE2-bgp-af-vpnv6] peer 1.1.1.9 enable
[PE2-bgp-af-vpnv6] quit
[PE2-bgp] quit
```

After completing the configurations, issue **display bgp peer** or the **display bgp vpnv6 all peer** on the PEs. The output shows a BGP peer relationship has been established between the PEs, and has reached the Established state.

```
[PE1] display bgp peer
 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        2        6     0          0  00:00:12 Established
```

6. Verify your configurations

Issue **display ipv6 routing-table vpn-instance** on the PEs. The output shows the routes to the CEs. The following uses PE 1 as an example:

```
[PE1] display ipv6 routing-table vpn-instance vpn1
Routing Table :
         Destinations : 3        Routes : 3

Destination: 2001:1::/64                            Protocol  : Direct
NextHop    : 2001:1::2                              Preference: 0
Interface  : Vlan11                                 Cost      : 0

Destination: 2001:1::2/128                          Protocol  : Direct
NextHop    : ::1                                    Preference: 0
Interface  : InLoop0                                Cost      : 0

Destination: 2001:2::/64                            Protocol  : BGP4+
NextHop    : ::FFFF:303:309                         Preference: 0
Interface  : NULL0                                  Cost      : 0
[PE1] display ipv6 routing-table vpn-instance vpn2
Routing Table :
         Destinations : 3        Routes : 3

Destination: 2001:3::/64                            Protocol  : Direct
NextHop    : 2001:3::2                              Preference: 0
Interface  : Vlan12                                 Cost      : 0

Destination: 2001:3::2/128                          Protocol  : Direct
NextHop    : ::1                                    Preference: 0
Interface  : InLoop0                                Cost      : 0

Destination: 2001:4::/64                            Protocol  : BGP4+
NextHop    : ::FFFF:303:309                         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 are not. For example, CE 1 can ping CE 3 (2001:3::1), but cannot ping CE 4 (2001:4::1):

```
[CE1] ping ipv6 2001:3::1
  PING 2001:3::1 : 56  data bytes, press CTRL_C to break
    Reply from 2001:3::1
    bytes=56 Sequence=1 hop limit=64  time = 1 ms
    Reply from 2001:3::1
    bytes=56 Sequence=2 hop limit=64  time = 1 ms
    Reply from 2001:3::1
    bytes=56 Sequence=3 hop limit=64  time = 1 ms
    Reply from 2001:3::1
```

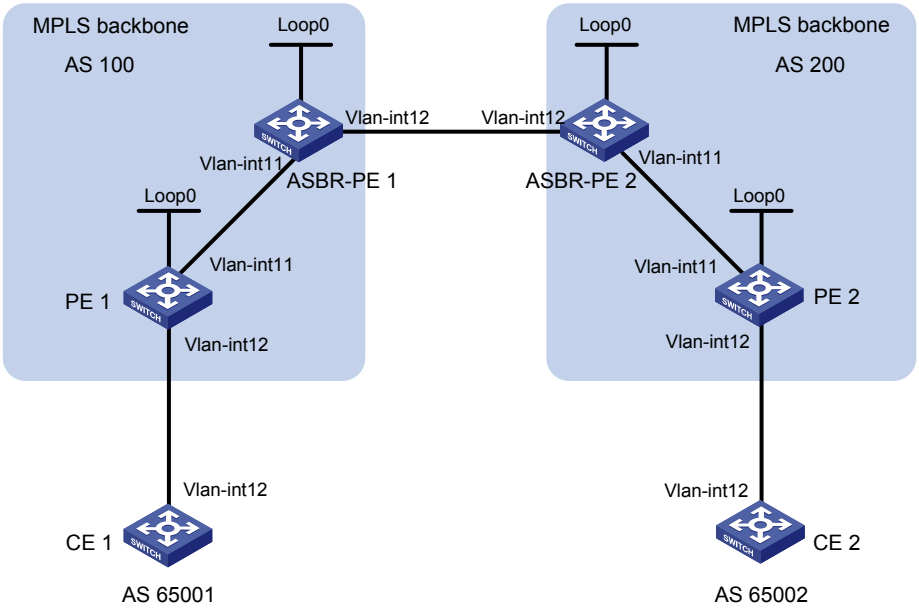
```
  bytes=56 Sequence=4 hop limit=64  time = 1 ms
  Reply from 2001:3::1
  bytes=56 Sequence=5 hop limit=64  time = 1 ms

--- 2001:3::1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 1/1/1 ms
[CE1] ping ipv6 2001:4::1
PING 2001:4::1 : 56  data bytes, press CTRL_C to break
  Request time out
  Request time out
  Request time out
  Request time out
  Request time out

--- 2001:4::1 ping statistics ---
  5 packet(s) transmitted
  0 packet(s) received
  100.00% packet loss
  round-trip min/avg/max = 0/0/0 ms
```

Configuring inter-AS IPv6 VPN 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.
- An inter-AS IPv6 MPLS L3VPN is implemented using option A, where the VRF-to-VRF method is used to manage VPN routes.
- The MPLS backbone in each AS runs OSPF.

Figure 84 Configure inter-AS IPv6 VPN option A



Device	Interface	IP address	Device	Interface	IP address
CE 1	Vlan-int12	2001:1::1/64	CE 2	Vlan-int12	2001:2::1/64
PE 1	Loop0	1.1.1.9/32	PE 2	Loop0	4.4.4.9/32
	Vlan-int12	2001:1::2/64		Vlan-int12	2001:2::2/64
	Vlan-int11	172.1.1.2/24		Vlan-int11	162.1.1.2/24
ASBR-PE 1	Loop0	2.2.2.9/32	ASBR-PE 2	Loop0	3.3.3.9/32
	Vlan-int11	172.1.1.1/24		Vlan-int11	162.1.1.1/24
	Vlan-int12	2002:1::1/64		Vlan-int12	2002:1::2/64

Procedure

1. Configure an IGP on each MPLS backbone to ensure IP connectivity within the backbone.

This example uses OSPF. The configuration steps are omitted.

Be sure to advertise the 32-bit loopback interface address of each router through OSPF. The loopback interface address of a switch is to be used as the switch's LSR ID.

After you complete the configurations, each ASBR PE and the PE in the same AS can establish OSPF adjacencies. Issue **display ospf peer**. The output shows that the adjacencies reach the Full state, and that PE and ASBR PE routers in the same AS 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 the MPLS basic capability and enable MPLS LDP on each MPLS backbone to establish LDP LSPs

Configure the MPLS basic capability on PE 1 and enable MPLS LDP for PE 1 and for the interface connected to ASBR-PE 1.

```
<PE1> system-view
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls
[PE1-mpls] quit
[PE1] mpls ldp
```

```
[PE1-mpls-ldp] quit
[PE1] interface vlan-interface 12
[PE1-Vlan-interface12] mpls
[PE1-Vlan-interface12] mpls ldp
[PE1-Vlan-interface12] quit
```

Configure the MPLS basic capability on ASBR-PE 1 and enable MPLS LDP for ASBR-PE 1 and for the interface connected to PE 1.

```
<ASBR-PE1> system-view
[ASBR-PE1] mpls lsr-id 2.2.2.9
[ASBR-PE1] mpls
[ASBR-PE1-mpls] quit
[ASBR-PE1] mpls ldp
[ASBR-PE1-mpls-ldp] quit
[ASBR-PE1] interface vlan-interface 11
[ASBR-PE1-Vlan-interface11] mpls
[ASBR-PE1-Vlan-interface11] mpls ldp
[ASBR-PE1-Vlan-interface11] quit
```

Configure the MPLS basic capability on ASBR-PE 2 and enable MPLS LDP for ASBR-PE 2 and for the interface connected to PE 2.

```
<ASBR-PE2> system-view
[ASBR-PE2] mpls lsr-id 3.3.3.9
[ASBR-PE2] mpls
[ASBR-PE2-mpls] quit
[ASBR-PE2] mpls ldp
[ASBR-PE2-mpls-ldp] quit
[ASBR-PE2] interface vlan-interface 11
[ASBR-PE2-Vlan-interface11] mpls
[ASBR-PE2-Vlan-interface11] mpls ldp
[ASBR-PE2-Vlan-interface11] quit
```

Configure the MPLS basic capability on PE 2 and enable MPLS LDP for PE 2 for the interface connected to ASBR-PE 2.

```
<PE2> system-view
[PE2] mpls lsr-id 4.4.4.9
[PE2] mpls
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface vlan-interface 11
[PE2-Vlan-interface11] mpls
[PE2-Vlan-interface11] mpls ldp
[PE2-Vlan-interface11] quit
```

After you complete the configurations, each PE and the ASBR PE in the same AS can establish LDP neighbor relationship. Issue **display mpls ldp session** on the switches. The output shows that the session status is Operational in the output information.

3. Configure a VPN instance on the PEs to allow the CEs to access

For the same VPN, the VPN targets for the VPN instance on the PE must match those for the VPN instance of the ASBR-PE in the same AS. This is not required for PEs in different ASs.

Configure CE 1.

```
<CE1> system-view
[CE1] interface vlan-interface 12
[CE1-Vlan-interface12] ipv6 address 2001:1::1 64
[CE1-Vlan-interface12] 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 vlan-interface 12
[PE1-Vlan-interface12] ip binding vpn-instance vpn1
[PE1-Vlan-interface12] ipv6 address 2001:1::2 64
[PE1-Vlan-interface12] quit
```

Configure CE 2.

```
<CE2> system-view
[CE2] interface vlan-interface 12
[CE2-Vlan-interface12] ipv6 address 2001:2::1 64
[CE2-Vlan-interface12] quit
```

Configure PE 2.

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance] route-distinguisher 200:2
[PE2-vpn-instance] vpn-target 100:1 both
[PE2-vpn-instance] quit
[PE2] interface vlan-interface 12
[PE2-Vlan-interface12] ip binding vpn-instance vpn1
[PE2-Vlan-interface12] ipv6 address 2001:2::2 64
[PE2-Vlan-interface12] 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-instance-vpn1] route-distinguisher 100:1
[ASBR-PE1-vpn-instance-vpn1] vpn-target 100:1 both
[ASBR-PE1-vpn-instance-vpn1] quit
[ASBR-PE1] interface vlan-interface 12
[ASBR-PE1-Vlan-interface12] ip binding vpn-instance vpn1
[ASBR-PE1-Vlan-interface12] ip address 192.1.1.1 24
[ASBR-PE1-Vlan-interface12] 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-vpn-vpn1] route-distinguisher 200:1
[ASBR-PE2-vpn-vpn-vpn1] vpn-target 100:1 both
[ASBR-PE2-vpn-vpn-vpn1] quit
```

```
[ASBR-PE2] interface vlan-interface 12
[ASBR-PE2-Vlan-interface12] ip binding vpn-instance vpn1
[ASBR-PE2-Vlan-interface12] ip address 192.1.1.2 24
[ASBR-PE2-Vlan-interface12] quit
```

After completing the configurations, you can view the VPN instance configurations by issuing **display ip vpn-instance**.

Each PE can ping its attached CE, and ASBR-PE 1 and ASBR-PE 2 can ping each other.

4. Establish eBGP peer relationship between PE and CE switches to allow VPN routes to be redistributed.

Configure CE 1.

```
[CE1] bgp 65001
[CE1-bgp] ipv6-family
[CE1-bgp-af-ipv6] peer 2001:1::2 as-number 100
[CE1-bgp-af-ipv6] import-route direct
[CE1-bgp-af-ipv6] quit
```

Configure PE 1.

```
[PE1] bgp 100
[PE1-bgp] ipv6-family vpn-instance vpn1
[PE1-bgp-ipv6-vpn1] peer 2001:1::1 as-number 65001
[PE1-bgp-ipv6-vpn1] import-route direct
[PE1-bgp-ipv6-vpn1] quit
[PE1-bgp] quit
```

Configure CE 2.

```
[CE2] bgp 65002
[CE1-bgp] ipv6-family
[CE2-bgp-af-ipv6] peer 2001:2::2 as-number 200
[CE2-bgp-af-ipv6] import-route direct
[CE2-bgp-af-ipv6] quit
```

Configure PE 2.

```
[PE2] bgp 200
[PE2-bgp] ipv6-family vpn-instance vpn1
[PE2-bgp-ipv6-vpn1] peer 2001:2::1 as-number 65002
[PE2-bgp-ipv6-vpn1] import-route direct
[PE2-bgp-ipv6-vpn1] quit
[PE2-bgp] quit
```

5. Establish iBGP peer relationship between each PE and the ASBR-PE in the same AS and 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] ipv6-family vpnv6
[PE1-bgp-af-vpnv6] peer 2.2.2.9 enable
[PE1-bgp-af-vpnv6] quit
```

Configure ASBR-PE 1.

```
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] ipv6-family vpn-instance vpn1
[ASBR-PE1-bgp-ipv6-vpn1] peer 2002:1::2 as-number 200
[ASBR-PE1-bgp-ipv6-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] ipv6-family vpnv6
[ASBR-PE1-bgp-af-vpnv6] peer 1.1.1.9 enable
[ASBR-PE1-bgp-af-vpnv6] quit
[ASBR-PE1-bgp] quit
```

Configure ASBR-PE 2.

```
[ASBR-PE2] bgp 200
[ASBR-PE2-bgp] ipv6-family vpn-instance vpn1
[ASBR-PE2-bgp-ipv6-vpn1] peer 2002:1::1 as-number 100
[ASBR-PE2-bgp-ipv6-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] ipv6-family vpnv6
[ASBR-PE2-bgp-af-vpnv6] peer 4.4.4.9 enable
[ASBR-PE2-bgp-af-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] ipv6-family vpnv6
[PE2-bgp-af-vpnv6] peer 3.3.3.9 enable
[PE2-bgp-af-vpnv6] quit
[PE2-bgp] quit
```

6. Verify your configurations

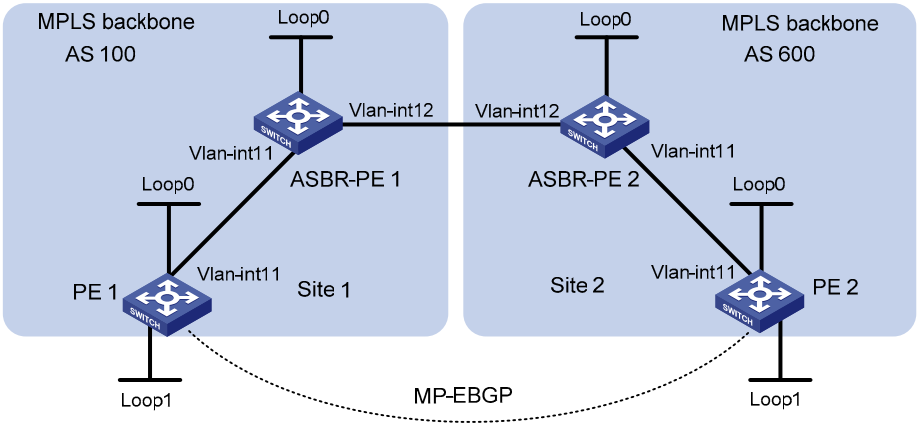
After you complete the configurations, display the routing table and use the ping command. The CEs have learned the route to each other and can ping each other.

Configuring inter-AS IPv6 VPN 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 MP-iBGP.
- PE 2 and ASBR-PE 2 exchange labeled IPv4 routes by MP-iBGP.
- PE 1 and PE 2 are MP-eBGP peers.
- 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 MP-eBGP to exchange labeled IPv4 routes.

Figure 85 Configure inter-AS IPv6 VPN option C



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
	Vlan-int11	1.1.1.2/8		Vlan-int11	9.1.1.2/8
ASBR-PE 1	Loop0	3.3.3.9/32	ASBR-PE 2	Loop0	4.4.4.9/32
	Vlan-int11	1.1.1.1/8		Vlan-int11	9.1.1.1/8
	Vlan-int12	11.0.0.2/8		Vlan-int12	11.0.0.1/8

Procedure

1. Configure PE 1

Run 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 an LSR ID, and enable MPLS and LDP.

```
[PE1] mpls lsr-id 2.2.2.9
[PE1] mpls
[PE1-mpls] label advertise non-null
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

Configure interface VLAN-interface 11, and start IS-IS and enable MPLS and LDP on the interface.

```
[PE1] interface vlan-interface 11
[PE1-Vlan-interface11] ip address 1.1.1.2 255.0.0.0
[PE1-Vlan-interface11] isis enable 1
[PE1-Vlan-interface11] mpls
[PE1-Vlan-interface11] mpls ldp
[PE1-Vlan-interface11] 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 VPN target attributes for it.

```
[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] ipv6 address 2001:1::1 128
[PE1-LoopBack1] quit
```

Start BGP.

```
[PE1] bgp 100
```

Configure the capability to advertise labeled routes to and receive labeled routes from the 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] peer 3.3.3.9 label-route-capability
```

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] ipv6-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] ipv6-family vpn-instance vpn1
[PE1-bgp-ipv6-vpn1] import-route direct
[PE1-bgp-ipv6-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 10.222.222.222.222.00
[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
[ASBR-PE1-mpls] label advertise non-null
[ASBR-PE1-mpls] quit
```

```
[ASBR-PE1] mpls ldp
[ASBR-PE1-mpls-ldp] quit
```

Configure interface VLAN-interface 11, and start IS-IS and enable MPLS and LDP on the interface.

```
[ASBR-PE1] interface vlan-interface 11
[ASBR-PE1-Vlan-interface11] ip address 1.1.1.1 255.0.0.0
[ASBR-PE1-Vlan-interface11] isis enable 1
[ASBR-PE1-Vlan-interface11] mpls
[ASBR-PE1-Vlan-interface11] mpls ldp
[ASBR-PE1-Vlan-interface11] quit
```

Configure interface VLAN-interface 12 and enable MPLS on it.

```
[ASBR-PE1] interface vlan-interface 12
[ASBR-PE1-Vlan-interface12] ip address 11.0.0.2 255.0.0.0
[ASBR-PE1-Vlan-interface12] mpls
[ASBR-PE1-Vlan-interface12] 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-policy1] apply mpls-label
[ASBR-PE1-route-policy1] quit
[ASBR-PE1] route-policy policy2 permit node 1
[ASBR-PE1-route-policy2] if-match mpls-label
[ASBR-PE1-route-policy2] apply mpls-label
[ASBR-PE1-route-policy2] quit
```

Start BGP on ASBR-PE 1 and redistribute routes from IS-IS process 1.

```
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] import-route isis 1
```

Apply routing policy policy2 to filter routes advertised to iBGP peer 2.2.2.9.

```
[ASBR-PE1-bgp] peer 2.2.2.9 as-number 100
[ASBR-PE1-bgp] peer 2.2.2.9 route-policy policy2 export
```

Configure the capability to advertise labeled routes to and receive labeled routes from iBGP peer 2.2.2.9.

```
[ASBR-PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[ASBR-PE1-bgp] peer 2.2.2.9 label-route-capability
```

Apply routing policy policy1 to filter routes advertised to eBGP peer 11.0.0.1.

```
[ASBR-PE1-bgp] peer 11.0.0.1 as-number 600
[ASBR-PE1-bgp] peer 11.0.0.1 route-policy policy1 export
```

Configure the capability to advertise labeled routes to and receive labeled routes from eBGP peer 11.0.0.1.

```
[ASBR-PE1-bgp] peer 11.0.0.1 label-route-capability
[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 10.222.222.222.222.00
[ASBR-PE2-isis-1] quit
```

Configure an LSR ID, enable MPLS and LDP.

```
[ASBR-PE2] mpls lsr-id 4.4.4.9
[ASBR-PE2] mpls
[ASBR-PE2-mpls] label advertise non-null
[ASBR-PE2-mpls] quit
[ASBR-PE2] mpls ldp
[ASBR-PE2-mpls-ldp] quit
```

Configure interface VLAN-interface 11, start IS-IS and enable MPLS and LDP on the interface.

```
[ASBR-PE2] interface vlan-interface 11
[ASBR-PE2-Vlan-interface11] ip address 9.1.1.1 255.0.0.0
[ASBR-PE2-Vlan-interface11] isis enable 1
[ASBR-PE2-Vlan-interface11] mpls
[ASBR-PE2-Vlan-interface11] mpls ldp
[ASBR-PE2-Vlan-interface11] 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 VLAN-interface 12 and enable MPLS on it.

```
[ASBR-PE2] interface vlan-interface 12
[ASBR-PE2-Vlan-interface12] ip address 11.0.0.1 255.0.0.0
[ASBR-PE2-Vlan-interface12] mpls
[ASBR-PE2-Vlan-interface12] quit
```

Create routing policies.

```
[ASBR-PE2] route-policy policy1 permit node 1
[ASBR-PE2-route-policy1] apply mpls-label
[ASBR-PE2-route-policy1] quit
[ASBR-PE2] route-policy policy2 permit node 1
[ASBR-PE2-route-policy2] if-match mpls-label
[ASBR-PE2-route-policy2] apply mpls-label
[ASBR-PE2-route-policy2] quit
```

Start BGP on ASBR-PE 2 and redistribute routes from IS-IS process 1.

```
[ASBR-PE2] bgp 600
[ASBR-PE2-bgp] import-route isis 1
```

Configure the capability to advertise labeled routes to and receive labeled routes from iBGP peer 5.5.5.9.

```
[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] peer 5.5.5.9 label-route-capability
```

Apply routing policy policy2 to filter routes advertised to iBGP peer 5.5.5.9.

```
[ASBR-PE2-bgp] peer 5.5.5.9 route-policy policy2 export
```

Apply routing policy policy1 to filter routes advertised to eBGP peer 11.0.0.2.

```
[ASBR-PE2-bgp] peer 11.0.0.2 as-number 100
[ASBR-PE2-bgp] peer 11.0.0.2 route-policy policy1 export
```

Configure the capability to advertise labeled routes to and receive labeled routes from eBGP peer 11.0.0.2.

```
[ASBR-PE2-bgp] peer 11.0.0.2 label-route-capability
[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 10.111.111.111.111.00
[PE2-isis-1] quit
```

Configure an LSR ID, and enable MPLS and LDP.

```
[PE2] mpls lsr-id 5.5.5.9
[PE2] mpls
[PE2-mpls] label advertise non-null
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

Configure interface VLAN-interface 11, and start IS-IS and enable MPLS and LDP on the interface.

```
[PE2] interface vlan-interface 11
[PE2-Vlan-interface11] ip address 9.1.1.2 255.0.0.0
[PE2-Vlan-interface11] isis enable 1
[PE2-Vlan-interface11] mpls
[PE2-Vlan-interface11] mpls ldp
[PE2-Vlan-interface11] 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 VPN target attributes for it.

```
[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] ipv6 address 2001:1::2 128
[PE2-LoopBack1] quit
```

Start BGP on PE 2.

```
[PE2] bgp 600
```

Configure 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] peer 4.4.4.9 label-route-capability
```

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] ipv6-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 vpn1.

```
[PE2-bgp] ipv6-family vpn-instance vpn1
[PE2-bgp-ipv6-vpn1] import-route direct
[PE2-bgp-ipv6-vpn1] quit
[PE2-bgp] quit
```

5. Verify your configurations

From each PE, ping the other PE. PE 1 and PE 2 can ping each other:

```
[PE2] ping ipv6 -vpn-instance vpn1 2001:1::1
  PING 2001:1::1 : 56  data bytes, press CTRL_C to break
    Reply from 2001:1::1
    bytes=56 Sequence=1 hop limit=64  time = 1 ms
    Reply from 2001:1::1
    bytes=56 Sequence=2 hop limit=64  time = 1 ms
    Reply from 2001:1::1
    bytes=56 Sequence=3 hop limit=64  time = 1 ms
    Reply from 2001:1::1
    bytes=56 Sequence=4 hop limit=64  time = 1 ms
    Reply from 2001:1::1
    bytes=56 Sequence=5 hop limit=64  time = 1 ms
  --- 2001:1::1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 1/1/1 ms
[PE1] ping ipv6 -vpn-instance vpn1 2001:1::2
  PING 2001:1::2 : 56  data bytes, press CTRL_C to break
    Reply from 2001:1::2
    bytes=56 Sequence=1 hop limit=64  time = 1 ms
    Reply from 2001:1::2
    bytes=56 Sequence=2 hop limit=64  time = 1 ms
    Reply from 2001:1::2
```

```
  bytes=56 Sequence=3 hop limit=64  time = 1 ms
  Reply from 2001:1::2
  bytes=56 Sequence=4 hop limit=64  time = 1 ms
  Reply from 2001:1::2
  bytes=56 Sequence=5 hop limit=64  time = 1 ms
--- 2001:1::2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 1/1/1 ms
```

Configuring carrier's carrier

Network requirements

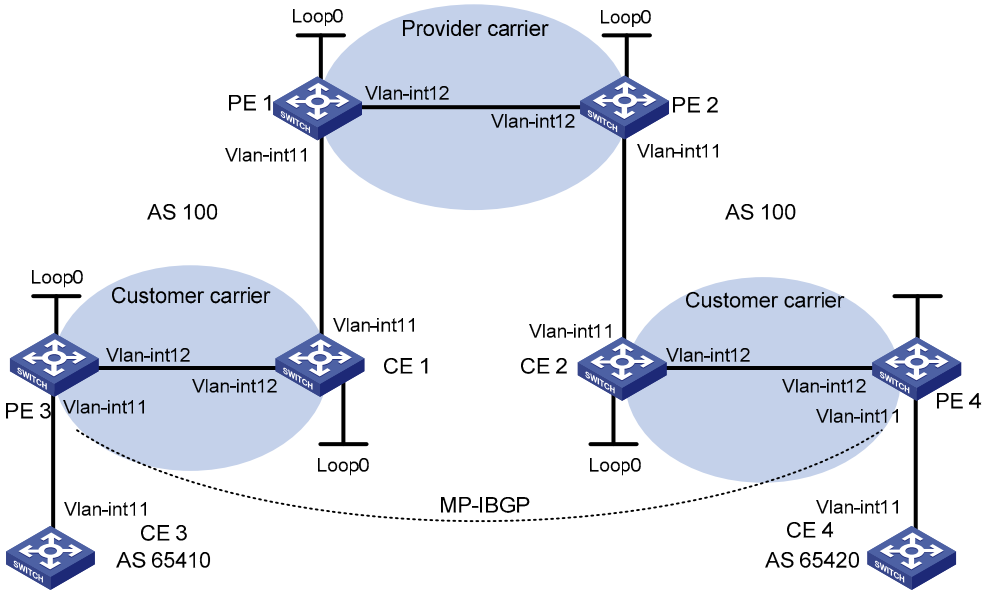
Configure carrier's carrier for the scenario shown in Figure 86. In this scenario:

- PE 1 and PE 2 are the provider carrier's PE switches. They provide VPN services for the customer carrier.
- CE 1 and CE 2 are the customer carrier's switches. They connect to the provider carrier's backbone as CE switches.
- PE 3 and PE 4 are the customer carrier's PE switches. They provide IPv6 MPLS L3VPN services for the 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 86 Configure carrier's carrier



Device	Interface	IP address	Device	Interface	IP address
CE 3	Vlan-int11	2001:1::1/64	CE 4	Vlan-int11	2001:2::1/64
PE 3	Loop0	1.1.1.9/32	PE 4	Loop0	6.6.6.9/32
	Vlan-int11	2001:1::2/64		Vlan-int11	2001:2::2/64
	Vlan-int12	10.1.1.1/24		Vlan-int12	20.1.1.2/24
CE 1	Loop0	2.2.2.9/32	CE 2	Loop0	5.5.5.9/32
	Vlan-int12	10.1.1.2/24		Vlan-int11	21.1.1.2/24
	Vlan-int11	11.1.1.1/24		Vlan-int12	20.1.1.1/24
PE 1	Loop0	3.3.3.9/32	PE 2	Loop0	4.4.4.9/32
	Vlan-int11	11.1.1.2/24		Vlan-int12	30.1.1.2/24
	Vlan-int12	30.1.1.1/24		Vlan-int11	21.1.1.1/24

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 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
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-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 vlan-interface 12
[PE1-Vlan-interface12] ip address 30.1.1.1 24
[PE1-Vlan-interface12] isis enable 1
[PE1-Vlan-interface12] mpls
[PE1-Vlan-interface12] mpls ldp
[PE1-Vlan-interface2] mpls ldp transport-address interface
[PE1-Vlan-interface2] 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] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 4.4.4.9 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

The configurations for PE 2 are similar to those for PE 1.

After completing the configurations, issue **display mpls ldp session** on PE 1 or PE 2; the output shows that the LDP session has been established successfully. Issue **display bgp peer**; the output shows that a BGP peer relationship has been established and has reached the Established state. Issue **display isis peer**; the output shows that an IS-IS neighbor relationship has been set up. Take PE 1 as an example:

```
[PE1] display mpls ldp session
               LDP Session(s) in Public Network
Total number of sessions: 1
 ------------------------------------------------------------------
 Peer-ID        Status        LAM  SsnRole  FT   MD5  KA-Sent/Rcv
 ------------------------------------------------------------------
 4.4.4.9:0      Operational   DU   Active   Off  Off  378/378
 ------------------------------------------------------------------
 LAM : Label Advertisement Mode         FT  : Fault Tolerance
[PE1] display bgp peer
 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      162      145     0        0  02:12:47 Established
[PE1] display isis peer
                         Peer information for ISIS(1)
                         ---------------------------
  System Id    Interface        Circuit Id  State HoldTime  Type   PRI
0000.0000.0005 Vlan-interface12 001         Up     29s      L1L2   --
```

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
[PE3-mpls] quit
[PE3] mpls ldp
[PE3-mpls-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 vlan-interface 12
[PE3-Vlan-interface12] ip address 10.1.1.1 24
[PE3-Vlan-interface12] isis enable 2
[PE3-Vlan-interface12] mpls
[PE3-Vlan-interface12] mpls ldp
[PE3-Vlan-interface12] mpls ldp transport-address interface
[PE3-Vlan-interface12] 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
[CE1-mpls] quit
[CE1] mpls ldp
[CE1-mpls-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 vlan-interface 12
[CE1-Vlan-interface12] ip address 10.1.1.2 24
[CE1-Vlan-interface12] isis enable 2
[CE1-Vlan-interface12] mpls
[CE1-Vlan-interface12] mpls ldp
[CE1-Vlan-interface12] mpls ldp transport-address interface
[CE1-Vlan-interface12] quit
```

After you complete the configurations, PE 3 and CE 1 can establish an LDP session and IS-IS neighbor relationship between them.

The configurations for PE 4 and CE 2 are similar to those for PE 3 and CE 1.

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 vpn-instance vpn1
[PE1-mpls-ldp-vpn-instance-vpn1] 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 vlan-interface11
[PE1-Vlan-interface11] ip binding vpn-instance vpn1
[PE1-Vlan-interface11] ip address 11.1.1.2 24
[PE1-Vlan-interface11] isis enable 2
[PE1-Vlan-interface11] mpls
[PE1-Vlan-interface11] mpls ldp
[PE1-Vlan-interface11] mpls ldp transport-address interface
[PE1-Vlan-interface11] quit
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import isis 2
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

Configure CE 1.

```
[CE1] interface vlan-interface11
[CE1-Vlan-interface11] ip address 11.1.1.1 24
[CE1-Vlan-interface11] isis enable 2
[CE1-Vlan-interface11] mpls
[CE1-Vlan-interface11] mpls ldp
[CE1-Vlan-interface11] mpls ldp transport-address interface
[CE1-Vlan-interface11] quit
```

After you complete the configurations, PE 1 and CE 1 can establish the LDP session and IS-IS neighbor relationship between them.

The configurations for PE 2 and CE 2 are similar to those for PE 1 and CE 1. The configuration steps are omitted.

4. Connect end customers to the customer carrier.

Configure CE 3.

```
<CE3> system-view
[CE3] interface vlan-interface11
[CE3-Vlan-interface11] ipv6 address 2001:1::1 64
[CE3-Vlan-interface11] quit
[CE3] bgp 65410
[CE3-bgp] ipv6-family
[CE3-bgp-af-ipv6] peer 2001:1::2 as-number 100
[CE3-bgp-af-ipv6] import-route direct
```

```
[CE3-bgp-af-ipv6] 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 Vlan-interface11
[PE3-Vlan-interface11] ip binding vpn-instance vpn1
[PE3-Vlan-interface11] ipv6 address 2001:1::2 64
[PE3-Vlan-interface11] quit
[PE3] bgp 100
[PE3-bgp] ipv6-family vpn-instance vpn1
[PE3-bgp-ipv6-vpn1] peer 2001:1::1 as-number 65410
[PE3-bgp-ipv6-vpn1] import-route direct
[PE3-bgp-ipv6-vpn1] quit
[PE3-bgp] quit
```

The configurations for PE 4 and CE 4 are similar to those for PE 3 and CE 3. The configuration steps are omitted.

5. Configure MP-iBGP peer relationship between PEs of the customer carrier to exchange the VPN routes of the customer carrier's 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] ipv6-family vpnv6
[PE3-bgp-af-vpnv6] peer 6.6.6.9 enable
[PE3-bgp-af-vpnv6] quit
[PE3-bgp] quit
```

The configurations for PE 4 are similar to those for PE 3. The configuration steps are omitted.

6. Verify your configurations

Issue **display ip routing-table** 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        Vlan12
30.1.1.0/24         Direct 0    0         30.1.1.1        Vlan12
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        Vlan12
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
```

Issue **display ip routing-table vpn-instance** 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. Issue **display ipv6 routing-table**

vpn-instance on PE 1 and PE 2. The output shows that their VPN routing tables do not contain the VPN routes that the customer carrier maintains. 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        Vlan11
2.2.2.9/32          ISIS   15   10        11.1.1.1        Vlan11
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        Vlan11
11.1.1.0/24         Direct 0    0         11.1.1.1        Vlan11
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        Vlan11
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
```

Issue **display ip routing-table** 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. Issue **display ipv6 routing-table vpn-instance** on CE 1 and CE 2. The output shows that the VPN routing tables do not contain the VPN routes that the customer carrier maintains. 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        Vlan12
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        Vlan11
6.6.6.9/32          ISIS   15   74        11.1.1.2        Vlan11
10.1.1.0/24         Direct 0    0         10.1.1.2        Vlan12
10.1.1.1/32         Direct 0    0         10.1.1.1        Vlan12
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        Vlan11
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        Vlan11
20.1.1.0/24         ISIS   15   74        11.1.1.2        Vlan11
21.1.1.0/24         ISIS   15   74        11.1.1.2        Vlan11
21.1.1.2/32         ISIS   15   74        11.1.1.2        Vlan11
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
```

Issue **display ip routing-table** 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        Vlan12
```

```
5.5.5.9/32          ISIS   15   84     10.1.1.2        Vlan12
6.6.6.9/32          ISIS   15   84     10.1.1.2        Vlan12
10.1.1.0/24         Direct 0    0      10.1.1.1        Vlan12
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        Vlan12
11.1.1.0/24         ISIS   15   20     10.1.1.2        Vlan12
20.1.1.0/24         ISIS   15   84     10.1.1.2        Vlan12
21.1.1.0/24         ISIS   15   84     10.1.1.2        Vlan12
21.1.1.2/32         ISIS   15   84     10.1.1.2        Vlan12
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
```

PE 3 and PE 4 can ping each other:

```
[PE3] ping 20.1.1.2
  PING 20.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 20.1.1.2: bytes=56 Sequence=1 ttl=252 time=127 ms
    Reply from 20.1.1.2: bytes=56 Sequence=2 ttl=252 time=97 ms
    Reply from 20.1.1.2: bytes=56 Sequence=3 ttl=252 time=83 ms
    Reply from 20.1.1.2: bytes=56 Sequence=4 ttl=252 time=70 ms
    Reply from 20.1.1.2: bytes=56 Sequence=5 ttl=252 time=60 ms

  --- 20.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 60/87/127 ms
```

CE 3 and CE 4 can ping each other:

```
[CE3] ping ipv6 2001:2::1
  PING 2001:2::1 : 56  data bytes, press CTRL_C to break
    Reply from 2001:2::1
    bytes=56 Sequence=1 hop limit=64  time = 1 ms
    Reply from 2001:2::1
    bytes=56 Sequence=2 hop limit=64  time = 1 ms
    Reply from 2001:2::1
    bytes=56 Sequence=3 hop limit=64  time = 1 ms
    Reply from 2001:2::1
    bytes=56 Sequence=4 hop limit=64  time = 1 ms
    Reply from 2001:2::1
    bytes=56 Sequence=5 hop limit=64  time = 1 ms

  --- 2001:2::1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 1/1/1 ms
```

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 A-Series 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

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.
Italic	*Italic* 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.

Port numbering in examples

The port numbers in this document are for illustration only and might be unavailable on your device.

Index