H3C S5500-HI 系列以太网交换机

MPLS 配置指导

杭州华三通信技术有限公司
http://www.h3c.com.cn

资料版本：6W102-20131220
产品版本：Release 52xx

Copyright © 2013 杭州华三通信技术有限公司及其许可者 版权所有，保留一切权利。
未经本公司书面许可，任何单位和个人不得擅自摘抄、复制本书内容的部分或全部，并不得以任何形式传播。

H3C、H3C、H3CS、H3CIE、H3CNE、Aolynk、Aolynk、H3Care、H3care、IRF、NetPilot、Netflow、SecEngine、SecPath、SecCenter、SecBlade、Comware、ITCMM、HUASAN、华三均为杭州华三通信技术有限公司的商标。对于本手册中出现的其它公司的商标、产品标识及商品名称，由各自权利人拥有。

由于产品版本升级或其他原因，本手册内容有可能变更。H3C 保留在没有任何通知或者提示的情况下对本手册的内容进行修改的权利。本手册仅作为使用指导，H3C 尽全力在本手册中提供准确的信息，但是 H3C 并不确保手册内容完全没有错误，本手册中的所有陈述、信息和建议也不构成任何明示或暗示的担保。

前 言

H3C S5500-HI 系列以太网交换机配置指导共分为十一本手册，介绍了 S5500-HI 系列以太网交换机 Release52xx 系列软件版本各软件特性的原理及其配置方法，包含原理简介、配置任务描述和配置举例。《MPLS 配置指导》主要介绍 MPLS 协议及其扩展功能的原理和配置，包括 MPLS 基本配置、建立 MPLS TE 隧道，以及利用 MPLS 标签实现二层 VPN 和三层 VPN 的配置。

前言部分包含如下内容：

- 读者对象
- 新增及修改特性说明
- 本书约定
- 产品配套资料
- 资料获取方式
- 技术支持
- 资料意见反馈

读者对象

本手册主要适用于如下工程师：

- 网络规划人员
- 现场技术支持与维护人员
- 负责网络配置和维护的网络管理员

新增及修改特性说明

本手册对应 S5500-HI 系列以太网交换机的 Release52xx 系列软件版本，各版本间特性差异如下：

- Release5206 与 Release5203 版本相比，无新增、修改特性。
- Release5203 与Release5101 版本相比，新增、修改了部分特性，具体请参见 表 1。

表1 Release5203 与 Release5101 版本间特性差异

配置指导	新增及修改特性
MCE	无
MPLS基本配置	新增特性：配置发送的LDP报文的DSCP优先级
MPLS TE	新增特性：配置发送的RSVP报文的DSCP优先级
VPLS	新增特性： • 配置 MAC 地址迁移功能 • 检测 PW
MPLS L2VPN	无
MPLS L3VPN	新增特性：在隧道策略中配置优选路径

本书约定

1. 命令行格式约定

格　式	意　义
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**加粗**字体表示。
斜体	命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
{ x | y | ... }	表示从多个选项中仅选取一个。
[x | y | ...]	表示从多个选项中选取一个或者不选。
{ x | y | ... }*	表示从多个选项中至少选取一个。
[x | y | ...]*	表示从多个选项中选取一个、多个或者不选。
&<1-n>	表示符号&前面的参数可以重复输入1～n次。
#	由“#”号开始的行表示为注释行。

2. 图形界面格式约定

格　式	意　义
< >	带尖括号“< >”表示按钮名，如“单击<确定>按钮”。
[ ]	带方括号“[ ]”表示窗口名、菜单名和数据表，如“弹出[新建用户]窗口”。
/	多级菜单用“/”隔开。如[文件/新建/文件夹]多级菜单表示[文件]菜单下的[新建]子菜单下的[文件夹]菜单项。

3. 各类标志

本书还采用各种醒目标志来表示在操作过程中应该特别注意的地方，这些标志的意义如下：

警告	该标志后的注释需给予格外关注，不当的操作可能会对人身造成伤害。
注意	提醒操作中应注意的事项，不当的操作可能会导致数据丢失或者设备损坏。
提示	为确保设备配置成功或者正常工作而需要特别关注的操作或信息。
说明	对操作内容的描述进行必要的补充和说明。
窍门	配置、操作、或使用设备的技巧、小窍门。

4. 图标约定

本书使用的图标及其含义如下：

	该图标及其相关描述文字代表一般网络设备，如路由器、交换机、防火墙等。
	该图标及其相关描述文字代表一般意义下的路由器，以及其他运行了路由协议的设备。

	该图标及其相关描述文字代表二、三层以太网交换机，以及运行了二层协议的设备。

5. 端口编号示例约定

本手册中出现的端口编号仅作示例，并不代表设备上实际具有此编号的端口，实际使用中请以设备上存在的端口编号为准。

产品配套资料

H3C S5500-HI 系列以太网交换机的配套资料包括如下部分：

大类	资料名称	内容介绍
产品知识介绍	产品彩页	帮助您了解产品的主要规格参数及亮点
	技术白皮书	帮助您了解产品和特性功能，对于特色及复杂技术从细节上进行介绍
硬件介绍及安装	安全兼容性手册	列出产品的兼容性声明，并对兼容性和安全的细节进行说明
	H3C 设备防雷安装指导手册	帮助您了解防雷接地设计和工程安装方法，以保证交换机具有良好的抗雷击性能
	快速安装指南	指导您对设备进行初始安装，通常针对最常用的情况，减少您的检索时间
	安装指导	帮助您详细了解设备硬件规格和安装方法，指导您对设备进行安装
	风扇安装手册	帮助您了解产品支持的可插拔风扇模块的外观、功能、规格、安装及拆卸方法
	电源手册	帮助您了解产品支持的可插拔电源模块的外观、功能、规格、安装及拆卸方法
	RPS电源用户手册	帮助您了解产品支持的RPS电源的外观、功能、规格
	H3C低端系列以太网交换机RPS电源选购指南	帮助您了解各种RPS电源适用的交换机产品型号及RPS电源配套电缆的相关规格
	接口模块扩展卡用户手册	帮助您了解该接口模块扩展卡的外观、规格、安装及拆卸方法
	H3C低端系列以太网交换机可插拔模块手册	帮助您了解产品支持的可插拔模块类型、外观和规格
	H3C可插拔SFP[SFP+][XFP]模块安装指南	帮助您掌握SFP/SFP+/XFP模块的正确安装方法，避免因操作不当而造成器件损坏
业务配置	配置指导	帮助您掌握设备软件功能的配置方法及配置步骤
	命令参考	详细介绍设备的命令，相当于命令字典，方便您查阅各个命令的功能
运行维护	故障处理手册	指导您快速定位并处理软件故障

大类	资料名称	内容介绍
	版本说明书	帮助您了解产品版本的相关信息（包括：版本配套说明、兼容性说明、特性变更说明、技术支持信息）及软件升级方法

资料获取方式

您可以通过H3C网站（www.h3c.com.cn）获取最新的产品资料：

H3C 网站与产品资料相关的主要栏目介绍如下：

- [服务支持/文档中心]：可以获取硬件安装类、软件升级类、配置类或维护类等产品资料。
- [产品技术]：可以获取产品介绍和技术介绍的文档，包括产品相关介绍、技术介绍、技术白皮书等。
- [解决方案]：可以获取解决方案类资料。
- [服务支持/软件下载]：可以获取与软件版本配套的资料。

技术支持

用户支持邮箱：service@h3c.com

技术支持热线电话：400-810-0504（手机、固话均可拨打）

010-62982107

网址：http://www.h3c.com.cn

资料意见反馈

如果您在使用过程中发现产品资料的任何问题，可以通过以下方式反馈：

E-mail：info@h3c.com

感谢您的反馈，让我们做得更好！

目 录

1 MCE配置 ····· 1-1
1.1 MCE简介 ····· 1-1
1.1.1 MPLS L3VPN简介 ····· 1-1
1.1.2 MPLS L3VPN的基本概念 ····· 1-2
1.1.3 MCE简介 ····· 1-4
1.1.4 MCE工作原理 ····· 1-4
1.1.5 在隧道环境中使用MCE ····· 1-5
1.2 MCE的路由信息交换 ····· 1-6
1.2.1 MCE与私网间的路由交换 ····· 1-6
1.2.2 MCE与PE间的路由交换 ····· 1-8
1.3 配置MCE ····· 1-8
1.3.1 配置VPN实例 ····· 1-8
1.4 配置MCE的路由交换 ····· 1-10
1.4.1 配置准备 ····· 1-11
1.4.2 配置MCE与站点之间的路由交换 ····· 1-11
1.4.3 配置MCE与PE之间的路由交换 ····· 1-16
1.5 MCE显示和维护 ····· 1-21
1.5.1 复位BGP连接 ····· 1-21
1.5.2 显示MCE的运行状态 ····· 1-21
1.6 MCE典型配置举例 ····· 1-22
1.6.1 MCE和PE间使用OSPF引入VPN路由 ····· 1-22
1.6.2 MCE和PE间使用BGP引入VPN路由 ····· 1-27
1.6.3 在隧道组网中使用MCE功能发布VPN路由 ····· 1-30
2 IPv6 MCE配置 ····· 2-1
2.1 IPv6 MCE简介 ····· 2-1
2.2 配置IPv6 MCE ····· 2-1
2.2.1 配置VPN实例 ····· 2-1
2.3 配置IPv6 MCE的路由交换 ····· 2-3
2.3.1 配置准备 ····· 2-3
2.3.2 配置IPv6 MCE与站点之间的路由交换 ····· 2-3
2.3.3 配置IPv6 MCE与PE之间的路由交换 ····· 2-7
2.4 IPv6 MCE显示和维护 ····· 2-10
2.4.1 复位BGP连接 ····· 2-10
2.4.2 显示IPv6 MCE的运行状态 ····· 2-10

2.5 IPv6 MCE典型配置举例 ·····2-11
2.5.1 配置IPv6 MCE典型配置举例 ·····2-11

1 MCE配置

说明

- 本章所指的路由器代表了一般意义下的路由器，以及运行了路由协议的三层交换机。为提高可读性，在手册的描述中将不另行说明。
- 本章只涉及 MCE 的介绍和配置，涉及到的各种路由协议的介绍和配置请参见“三层技术-IP 路由配置指导”中的相关内容。
- MCE 功能中所指的“接口”为三层口，包括 VLAN 接口、三层以太网接口等。三层以太网接口是指工作模式被配置成三层模式的以太网端口，有关以太网端口工作模式切换的操作，请参见“二层技术-以太网交换配置指导”中的“以太网端口配置”部分。

1.1 MCE简介

1.1.1 MPLS L3VPN简介

MPLS L3VPN 是服务提供商 VPN 解决方案中一种基于 PE 的 L3VPN 技术，它使用 BGP 在服务提供商骨干网上发布 VPN 路由，使用 MPLS 在服务提供商骨干网上转发 VPN 报文。

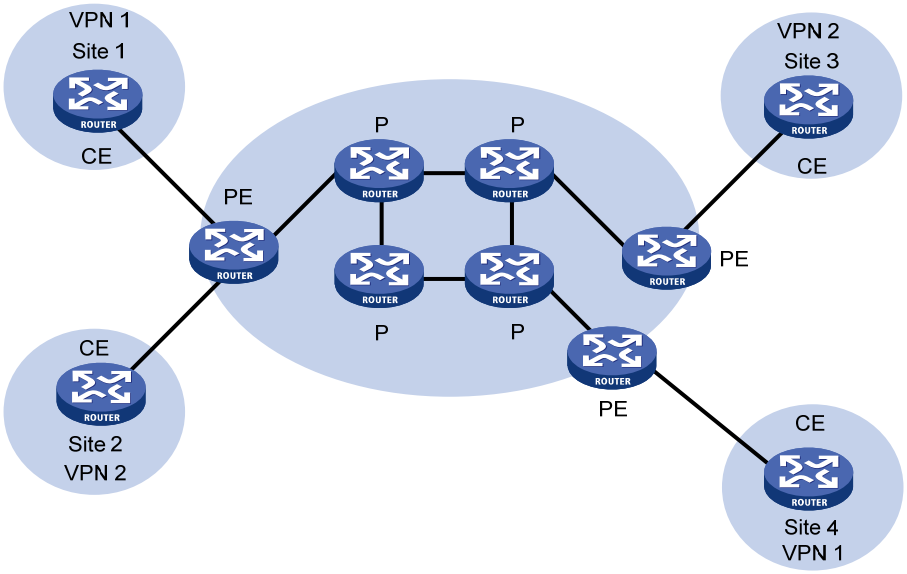
MPLS L3VPN 组网方式灵活、可扩展性好，并能够方便地支持 MPLS QoS 和 MPLS TE，因此得到越来越多的应用。

MPLS L3VPN 模型由三部分组成：CE、PE 和 P。

- CE（Customer Edge）设备：用户网络边缘设备，有接口直接与 SP（Service Provider，服务提供商）相连。CE 可以是路由器或交换机，也可以是一台主机。CE“感知”不到 VPN 的存在，也不需要必须支持 MPLS。
- PE（Provider Edge）路由器：服务提供商边缘路由器，是服务提供商网络的边缘设备，与用户的 CE 直接相连。在 MPLS 网络中，对 VPN 的所有处理都发生在 PE 上。
- P（Provider）路由器：服务提供商网络中的骨干路由器，不与 CE 直接相连。P 设备只需要具备基本 MPLS 转发能力。

图 1-1 是一个MPLS L3VPN组网方案的示意图。

图1-1 MPLS L3VPN 组网



CE 和 PE 的划分主要是根据 SP 与用户的管理范围，CE 和 PE 是两者管理范围的边界。

CE 设备通常是一台路由器，当 CE 与直接相连的 PE 建立邻接关系后，CE 把本站点的 VPN 路由发布给 PE，并从 PE 学到远端 VPN 的路由。CE 与 PE 之间使用 BGP/IGP 交换路由信息，也可以使用静态路由。

PE 从 CE 学到 CE 本地的 VPN 路由信息后，通过 BGP 与其它 PE 交换 VPN 路由信息。PE 路由器只维护与它直接相连的 VPN 的路由信息，不维护服务提供商网络中的所有 VPN 路由。

P 路由器只维护到 PE 的路由，不需要了解任何 VPN 路由信息。

当在 MPLS 骨干网上传输 VPN 流量时，入口 PE 做为 Ingress LSR（Label Switch Router，标签交换路由器），出口 PE 做为 Egress LSR，P 路由器则做为 Transit LSR。

1.1.2 MPLS L3VPN的基本概念

1. 站点

在介绍 VPN 时经常会提到“站点”，站点（Site）的含义可以从下述几个方面理解：

- 站点是指相互之间具备 IP 连通性的一组 IP 系统，并且，这组 IP 系统的 IP 连通性不需通过服务提供商网络实现；
- 站点的划分是根据设备的拓扑关系，而不是地理位置，尽管在大多数情况下一个站点中的设备地理位置相邻；
- 一个站点中的设备可以属于多个 VPN，换言之，一个站点可以属于多个 VPN；
- 站点通过 CE 连接到服务提供商网络，一个站点可以包含多个 CE，但一个 CE 只属于一个站点。

对于多个连接到同一服务提供商网络的站点，通过制定策略，可以将它们划分为不同的集合（set），只有属于相同集合的站点之间才能通过服务提供商网络互访，这种集合就是 VPN。

2. 地址空间重叠

VPN 是一种私有网络，不同的 VPN 独立管理自己使用的地址范围，也称为地址空间（Address Space）。

不同 VPN 的地址空间可能会在一定范围内重合，比如，VPN 1 和 VPN 2 都使用了 10.110.10.0/24 网段的地址，这就发生了地址空间重叠（Overlapping Address Spaces）。

3. VPN实例

在 MPLS VPN 中，不同 VPN 之间的路由隔离通过 VPN 实例（VPN-instance）实现。

PE 为每个直接相连的站点建立并维护专门的 VPN 实例。VPN 实例中包含对应站点的 VPN 成员关系和路由规则。如果一个站点中的用户同时属于多个 VPN，则该站点的 VPN 实例中将包括所有这些 VPN 的信息。

为保证 VPN 数据的独立性和安全性，PE 上每个 VPN 实例都有相对独立的路由表和 LFIB（Label Forwarding Information Base，标签转发表）。

具体来说，VPN 实例中的信息包括：标签转发表、IP 路由表、与 VPN 实例绑定的接口以及 VPN 实例的管理信息。VPN 实例的管理信息包括 RD（Route Distinguisher，路由标识符）、路由过滤策略、成员接口列表等。

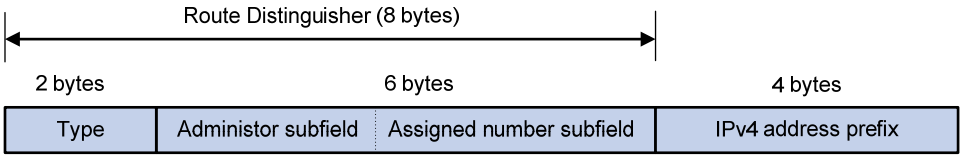
4. VPN-IPv4 地址

传统 BGP 无法正确处理地址空间重叠的 VPN 的路由。假设 VPN 1 和 VPN 2 都使用了 10.110.10.0/24 网段的地址，并各自发布了一条去往此网段的路由，BGP 将只会选择其中一条路由，从而导致去往另一个 VPN 的路由丢失。

PE 路由器之间使用 MP-BGP 来发布 VPN 路由，并使用 VPN-IPv4 地址族来解决上述问题。

VPN-IPv4 地址共有 12 个字节，包括 8 字节的RD和 4 字节的IPv4 地址前缀，如 图 1-2 所示。

图1-2 VPN-IPv4 地址结构



PE 从 CE 接收到普通 IPv4 路由后，需要将这些私网 VPN 路由发布给对端 PE。私网路由的独立性是通过为这些路由附加 RD 实现的。

SP 可以独立地分配 RD，但必须保证 RD 的全局唯一性。这样，即使来自不同服务提供商的 VPN 使用了同样的 IPv4 地址空间，PE 路由器也可以向各 VPN 发布不同的路由。

建议为 PE 上每个 VPN 实例配置专门的 RD，以保证到达同一 CE 的路由都使用相同的 RD。RD 为 0 的 VPN-IPv4 地址相当于全局唯一的 IPv4 地址。

RD 的作用是添加到一个特定的 IPv4 前缀，使之成为全局唯一的 VPN IPv4 前缀。

RD 或者是与自治系统号（ASN）相关的，在这种情况下，RD 是由一个自治系统号和一个任意的数组成；或者是与 IP 地址相关的，在这种情况下，RD 是由一个 IP 地址和一个任意的数组成。

RD 有三种格式，通过 2 字节的 Type 字段区分：

- Type 为 0 时，Administrator 子字段占 2 字节，Assigned number 子字段占 4 字节，格式为：16bits 自治系统号:32bits 用户自定义数字。例如：100:1
- Type 为 1 时，Administrator 子字段占 4 字节，Assigned number 子字段占 2 字节，格式为：32bitsIPv4 地址:16bits 用户自定义数字。例如：172.1.1.1:1
- Type 为 2 时，Administrator 子字段占 4 字节，Assigned number 子字段占 2 字节，格式为：32bits 自治系统号：16bits 用户自定义数字，其中的自治系统号最小值为 65536。例如：65536:1

为保证 RD 的全局唯一性，建议不要将 Administrator 子字段的值设置为私有 AS 号或私有 IP 地址。

5. VPN Target属性

MPLS L3VPN 使用 BGP 扩展团体属性——VPN Target（也称为 Route Target）来控制 VPN 路由信息的发布。

PE 路由器上的 VPN 实例有两类 VPN Target 属性：

- Export Target 属性：在本地 PE 将从与自己直接相连的站点学到的 VPN-IPv4 路由发布给其它 PE 之前，为这些路由设置 Export Target 属性；
- Import Target 属性：PE 在接收到其它 PE 路由器发布的 VPN-IPv4 路由时，检查其 Export Target 属性，只有当此属性与 PE 上 VPN 实例的 Import Target 属性匹配时，才把路由加入到相应的 VPN 路由表中。

也就是说，VPN Target 属性定义了一条 VPN-IPv4 路由可以为哪些站点所接收，PE 路由器可以接收哪些站点发送来的路由。

与 RD 类似，VPN Target 也有三种格式：

- 16bits 自治系统号:32bits 用户自定义数字，例如：100:1。
- 32bits IPv4 地址:16bits 用户自定义数字，例如：172.1.1.1:1。
- 32bits 自治系统号：16bits 用户自定义数字，其中的自治系统号最小值为 65536。例如：65536:1。

1.1.3 MCE简介

BGP/MPLS VPN以隧道的方式解决了在公网中传送私网数据的问题，但传统的BGP/MPLS VPN架构要求每个VPN实例单独使用一个CE与PE相连，如 图 1-1 所示。

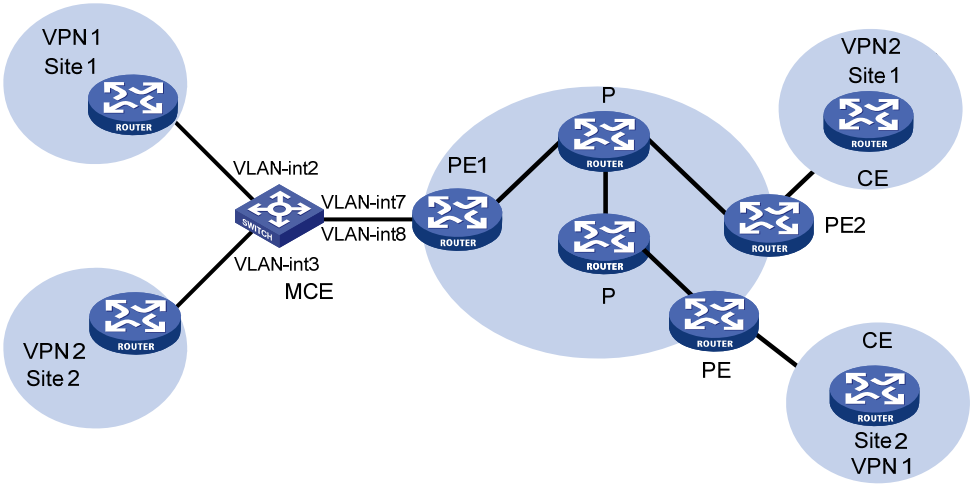
随着用户业务的不断细化和安全需求的提高，很多情况下一个私有网络内的用户需要划分成多个 VPN，不同 VPN 用户间的业务需要完全隔离。此时，为每个 VPN 单独配置一台 CE 将加大用户的设备开支和维护成本；而多个 VPN 共用一台 CE，使用同一个路由表项，又无法保证数据的安全性。

使用本系列交换机提供的 MCE（Multi-VPN-Instance CE，具备多 VPN 实例功能的 CE）功能，可以有效解决多 VPN 网络带来的用户数据安全与网络成本之间的矛盾，它使用 CE 设备本身的 VLAN 接口编号与网络内的 VPN 进行绑定，并为每个 VPN 创建和维护独立的路由转发表（Multi-VRF）。这样不但能够隔离私网内不同 VPN 的报文转发路径，而且通过与 PE 间的配合，也能够将每个 VPN 的路由正确发布至对端 PE，保证 VPN 报文在公网内的传输。

1.1.4 MCE工作原理

下面以 图 1-3 为例介绍MCE对多个VPN的路由表项进行维护，并与PE交互VPN路由的过程。

图1-3 MCE 工作原理示意图



如 图 1-3 所示，左侧私网内有两个VPN站点：站点 1 和站点 2，分别通过MCE设备接入MPLS骨干网，其中VPN1 和VPN2 的用户，需要分别与远端站点 2 内的VPN1 用户和站点 1 内的VPN2 用户建立VPN隧道。

通过配置 MCE 功能，可以在 MCE 设备上为 VPN1 和 VPN2 创建不同的 VPN 实例，为其维护各自独立的路由表，并使用 Vlan-interface2 接口与 VPN1 进行绑定、Vlan-interface3 与 VPN2 进行绑定。在接收路由信息时，MCE 设备根据接收接口的编号，即可判断该路由信息的来源，并将其维护到对应 VPN 实例的路由转发表中。

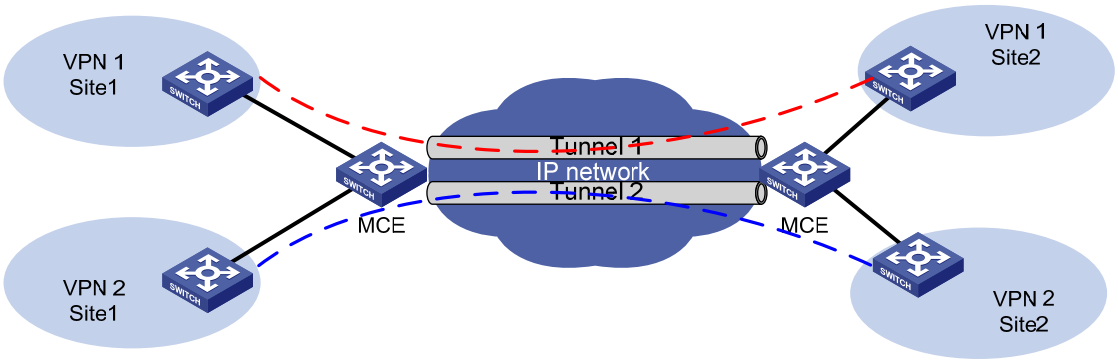
同时，MCE 需要通过不同的路由协议或进程来向 PE 发布 VPN1 和 VPN2 的路由，以便 PE 能够识别并分别维护不同 VPN 的路由。因此，MCE 需要通过两个接口连接到 PE，并创建两个 VPN 实例与两个接口分别绑定。在向 PE 发布 VPN 路由时，将不同的路由协议或进程与 VPN 实例绑定，并在不同的协议或进程中分别引入 VPN1 和 VPN2 内的路由。

通过上述过程，MCE 即可将两个 VPN 内的路由信息分别发布到 PE 设备。

1.1.5 在隧道环境中使用MCE

除MPLS L3VPN外，目前能够提供VPN功能的还有隧道技术。本交换机提供的MCE功能在通过隧道实现VPN的组网中，也可以进行应用。具体的实现方式如 图 1-4 所示：

图1-4 在隧道组网中使用 MCE 的组网方式一

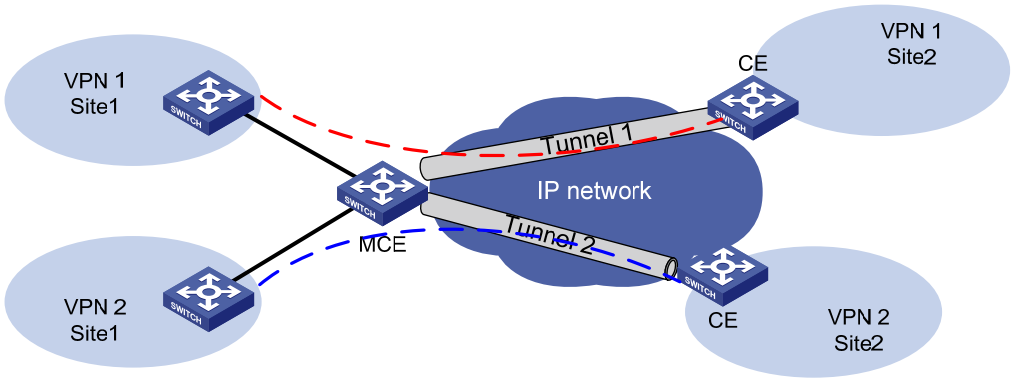


在两台MCE设备间建立多条隧道，并将隧道接口与指定的VPN实例进行绑定，可以使VPN实例的路由信息和数据通过绑定的隧道接口传输至对端设备，对端MCE设备根据接收路由的隧道接口来判断

该路由所属的VPN实例，并将其发布给相应的站点。例如在 图 1-4 中将隧道 1 与VPN1 实例绑定，便可以使MCE设备在该隧道上传输VPN1 的路由信息和数据。

另外一种隧道环境的组网方式是将MCE设备通过隧道与远端多个CE进行连接，该方式中CE设备只需按正常方式接收和发布路由，MCE设备根据隧道接口与VPN的绑定关系来发送和接收VPN路由信息。组网示意如 图 1-5 所示：

图1-5 在隧道组网中使用 MCE 的组网方式二



说明

- 在隧道环境中使用MCE时，MCE设备可以直接与对端MCE设备（或多个CE设备）交互VPN路由信息，交互方式与在MPLS L3VPN环境下MCE与PE的交互方式相同，请参见 1.2.2 MCE与PE间的路由交换。
- 目前支持 MCE 功能的隧道类型有 GRE 隧道、IPv4 over IPv4 隧道、IPv4 over IPv6 隧道。
- 关于隧道种类的介绍和相关配置，请参见“三层技术-IP 业务配置指导”中的“隧道配置”部分。

1.2 MCE的路由信息交换

通过接口与 VPN 实例的绑定，MCE 与 PE 已经能够正确判断报文的来源，参考对应 VPN 实例的路由信息对报文进行转发。下面介绍一下 MCE 设备如何将多个 VPN 实例的私网路由信息准确传播到 PE 设备。

1.2.1 MCE与私网间的路由交换

MCE 可以使用如下的路由协议与站点交换 VPN 私网路由：

- 静态路由
- RIP
- OSPF
- IS-IS
- IBGP
- EBGP

说明

下文只介绍各路由协议与 MCE 功能配合的配置思路，有关路由协议的基本原理，请参见“三层技术-IP 路由配置指导”中的相关介绍。

1. 使用静态路由

MCE 可以通过静态路由与站点连接。传统 CE 配置的静态路由对全局生效，无法解决多 VPN 间的地址重叠问题。本系列交换机提供的 MCE 功能可以将静态路由与 VPN 实例相绑定，将各 VPN 之间的静态路由进行隔离。

2. 使用RIP

本系列交换机提供了将 RIP 进程与 VPN 实例绑定的方法，通过在 MCE 上配置 RIP 进程与 VPN 实例的绑定关系，使不同 VPN 内的私网路由可以通过不同的 RIP 进程在站点和 MCE 间进行交互，保证了私网路由的隔离和安全。

3. 使用OSPF

本系列交换机提供了将 OSPF 进程与 VPN 实例绑定的功能，用来在 MCE 上隔离不同 VPN 的路由。

需要注意的是：

在 VPN 实例所绑定的 OSPF 进程中，将不会使用在系统视图下配置的公网 Router ID，需要用户在启动进程时手工配置 Router ID。

一个 OSPF 进程只能绑定一个 VPN 实例，但一个 VPN 实例可以使用多个 OSPF 进程为其传播私网路由。同一 VPN 实例内的 OSPF 进程应配置有相同的域 ID，以保证路由发布的正确性。

说明

对于标准的 BGP/OSPF 互相引用功能，当在 MCE 设备上配置引入 BGP 路由到 OSPF 中时，该路由的原 OSPF 属性将无法恢复，造成该路由与从其他域引入的路由无法区分。为了区分原属于不同 OSPF 域的路由，需要在远端 PE 将 OSPF 路由引入到 BGP 时携带标识域的属性，即 OSPF 的域 ID（Domain ID）。OSPF 进程的域 ID 包含在此进程生成的路由中，在将 OSPF 路由引入 BGP 中时，域 ID 被附加到 BGP VPN 路由上，作为 BGP 的扩展团体属性传递。

在某些情况下，同一个 VPN 可能会连接多个 MCE 设备，当其中一个 MCE 将从 BGP 学到的路由向 VPN 内发布时，可能会被另外的 MCE 设备学到，造成路由环。为避免路由环路，可以在 MCE 上为不同的 VPN 实例配置 Route Tag，建议在多个 MCE 上为同一个 VPN 配置相同的 Route Tag。

4. 使用IS-IS

MCE 和站点之间使用 IS-IS 传播私网路由的方式与使用 OSPF 时类似，将进程与 VPN 实例进行绑定。一个 IS-IS 进程只能属于一个 VPN 实例。

5. 使用IBGP

当 MCE 和站点之间使用 IBGP 传播私网路由时，需要在 MCE 上为每个 VPN 实例配置 IBGP 对等体，并引入相应 VPN 内的 IGP 路由信息。当 MCE 同时与属于同一 VPN 的多个站点进行连接时，可以将 MCE 设备配置为路由反射器，站点的出口路由器配置为客户端，使 MCE 在多个站点之间传递（反射）路由信息，而站点之间不需要建立 BGP 连接，减少了 BGP 连接数量，简化了网络配置。

6. 使用EBGP

当 MCE 和站点之间使用 EBGP 传播私网路由时，需要在 MCE 上为每个 VPN 实例配置 EBGP 对等体，并引入相应 VPN 内的 IGP 路由信息。由于一般情况下各个站点处于不同的 AS 内，因此应使用 EBGP 进行路由的传播。

(1) 配置 EBGP 引入各站点内的 IGP 路由

为正确的将私网路由发布到 PE，需要 MCE 首先将与其直连站点内的 IGP 路由引入到自己的 BGP 路由表中。

(2) 为每个 VPN 实例配置对等体组

为准确地与各站点交换路由信息，可以在 BGP 的 IPv4 地址族视图下，为每个 VPN 实例配置对等体组，并指定对等体组的 AS 编号。

(3) 通过 Filter-policy 对路由进行过滤

为正确地将路由信息传播到站点和 PE 设备，在指定对等体后，还需要使用 Filter-policy 对接收/发布的路由进行过滤。

1.2.2 MCE与PE间的路由交换

由于在 MCE 设备上已经将路由信息与 VPN 实例进行了绑定，因此，只需要在 MCE 与 PE 之间将接口与 VPN 实例进行绑定，再进行简单的路由配置，便可以将 MCE 的 VPN 路由表项引入到 MCE-PE 间的路由协议中，实现私网 VPN 路由信息的传播。

MCE 与 PE 之间可以使用以下的路由协议进行路由交换：

- 静态路由
- RIP
- OSPF
- IS-IS
- IBGP
- EBGP

各路由协议的配置方法及引入路由的操作，请参考“三层技术-IP 路由配置指导”中的介绍。

1.3 配置MCE

1.3.1 配置VPN实例

VPN 实例用于将 VPN 私网路由与公网路由隔离，在所有 MCE 组网方案中，都需要配置 VPN 实例。

VPN 实例不仅可以将 VPN 私网路由与公网路由隔离，不同 VPN 实例的路由之间也是相互隔离的，这一特点使得 VPN 实例的使用不限于 MCE。

1. 创建VPN实例

VPN 实例在实现中与站点关联。VPN 实例不是直接对应于 VPN，一个 VPN 实例综合了和它所对应站点的 VPN 成员关系和路由规则。

一个 VPN 实例只有配置了 RD 后才生效。在配置 RD 之前，除了描述信息外，不能配置 VPN 实例的其他任何参数。

描述信息用于描述 VPN 实例，可以用来记录 VPN 实例与某个 VPN 的关系等信息。

表1-1 创建 VPN 实例

操作	命令	说明
进入系统视图	**system-view**	-
创建VPN实例，并进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	必选
配置VPN实例的RD	**route-distinguisher** *route-distinguisher*	必选
配置VPN实例的描述信息	**description** *text*	可选

注意

为方便记忆和管理，建议用户在 MCE 设备和 PE 设备上为同一个 VPN 实例配置相同的 RD。

2. 配置VPN实例与接口关联

VPN 实例配置完成后，还需要与连接各个 VPN 站点或 PE 设备的接口进行关联。

说明

- 在隧道环境中，需要将 VPN 实例与连接对端 MCE 设备（或 CE 设备）的隧道接口进行关联。
- 本系列交换机还支持将管理用以太网口与 VPN 实例进行绑定，使管理用以太网口上的 IP 地址仅参与 VPN 实例内部的路由计算。
- 配置三层聚合接口与 VPN 实例关联时，还需要同时配置其所有成员端口与相同的 VPN 实例关联，否则会影响流量的正常转发。
- 执行 **ip binding vpn-instance** 命令将删除接口上已经配置的 IP 地址，因此需要重新配置接口的 IP 地址。

表1-2 配置 VPN 实例与接口关联

操作	命令	说明
进入系统视图	**system-view**	-
进入要关联接口的接口视图	**interface** *interface-type interface-number*	-
将当前接口与VPN实例关联	**ip binding vpn-instance** *vpn-instance-name*	必选 缺省情况下，接口不关联任何VPN实例

3. 配置VPN实例的路由相关属性

MCE 设备对 VPN 路由的发布控制过程如下：

- 当从 VPN 站点内学习到的一条 VPN 路由引入 BGP 时，BGP 为它关联一个 VPN Target 扩展团体属性列表，通常这个列表是与该站点相关联的 VPN 实例的输出路由属性列表。
- VPN 实例根据 VPN Target 中 **import-extcommunity** 决定可被接受并引入此 VPN 实例的路由。

- VPN 实例根据 VPN Target 中的 **export-extcommunity** 对向外发布的路由进行 VPN Target 属性的修改。

表1-3 配置 VPN 实例的路由相关属性

操作	命令	说明
进入系统视图	**system-view**	-
进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	必选
进入IPv4 VPN视图	**ipv4-family**	可选
将当前VPN实例与一个或多个VPN Target相关联	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	必选
配置VPN实例支持的最大路由数	**routing-table limit** *number* { *warn-threshold* | **simply-alert** }	可选 缺省情况下，没有配置当前VPN实例支持的最多路由数
对当前VPN实例应用入方向路由策略	**import route-policy** *route-policy*	可选 缺省情况下，允许所有VPN Target属性匹配的路由通过
对当前VPN实例应用出方向路由策略	**export route-policy** *route-policy*	可选 缺省情况下，允许所有VPN Target属性匹配的路由通过

说明

- 只有当 MCE 与 PE 间运行 BGP 时，VPN Target 属性才会随路由发布到 PE，否则配置该属性没有意义。
- 既可以在 VPN 实例视图下，也可以在 IPv4 VPN 视图下，配置 IPv4 VPN 的路由相关属性。如果同时在两个视图下配置了路由相关属性，则 IPv4 VPN 采用 IPv4 VPN 视图下配置的路由相关属性。
- 命令 **vpn-target** 最多可以配置 8 个 VPN Target；一个 VPN 实例最多可以配置 64 个 VPN Target。
- 可以配置一个 VPN 实例可以支持的最大路由数，以防止 PE 路由器的入接口有过多的路由。
- 为 VPN 实例配置路由策略之前必须已经创建了路由策略，否则将采用缺省路由策略。

1.4 配置MCE的路由交换

MCE 可以看作一种通过路由隔离实现业务隔离的组网方案。配置 MCE 的关键为：

- 配置 MCE 与站点之间的路由交换
- 配置 MCE 与 PE 之间的路由交换

在 MCE 组网方案中，路由计算时需要关闭 PE 上的路由环路检测功能，防止路由丢失；同时禁止各路由协议互操作功能，以节省系统资源。

1.4.1 配置准备

在配置 MCE 之前，需完成以下任务：

- 创建 VPN 实例、在 MCE 上将连接站点和 PE 的接口与 VPN 实例绑定
- 配置相关接口的链路层协议和网络层协议，保证路由可达

1.4.2 配置MCE与站点之间的路由交换

1. 配置MCE与站点之间使用静态路由

MCE 可以通过静态路由与 Site 连接。传统 CE 配置的静态路由对全局生效，无法解决多 VPN 间的地址重叠问题。以太网交换机提供的 MCE 功能可以将静态路由与 VPN 实例相绑定，将各 VPN 之间的静态路由进行隔离。

表1-4 配置 MCE 与站点之间使用静态路由

操作	命令	说明
进入系统视图	**system-view**	-
为指定VPN实例配置静态路由	**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*]	二者必选其一 该配置在MCE上进行，站点上的配置方法与普通静态路由相同
	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*]	
配置静态路由的缺省优先级	**ip route-static default-preference** *default-preference-value*	可选 缺省情况下，静态路由的缺省优先级为60

2. 配置MCE与站点之间使用RIP

一个 RIP 进程只能属于一个 VPN 实例。如果在启动 RIP 进程时不绑定到 VPN 实例，则该进程属于公网进程。通过在 MCE 上将 RIP 进程与 VPN 实例绑定，可以使不同 VPN 内的私网路由通过不同的 RIP 进程在 Site 和 MCE 间进行交互，保证了私网路由的隔离和安全。

表1-5 配置 MCE 与站点之间使用 RIP

操作	命令	说明
进入系统视图	**system-view**	-
创建MCE与站点间的RIP实例，并进入RIP视图	**rip** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在MCE上进行，站点上配置普通RIP即可
在指定网段接口上使能RIP	**network** *network-address*	必选 缺省情况下，接口上的RIP功能处于关闭状态

操作	命令	说明
引入由PE发布的远端站点的路由	**import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | **route-policy** *route-policy-name* | **tag** *tag*] *	必选 缺省情况下，RIP未引入其它路由
配置引入路由的缺省度量值	**default cost** *value*	可选 缺省情况下，引入路由的缺省度量值为0

说明

有关 RIP 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“RIP”。

3. 配置MCE与站点之间使用OSPF

一个 OSPF 进程只能属于一个 VPN 实例。如果在启动 OSPF 进程时不绑定到 VPN 实例，则该进程属于公网进程。

通过在 MCE 上将 OSPF 进程与 VPN 实例绑定，可以使不同 VPN 内的私网路由通过不同的 OSPF 进程在 Site 和 MCE 间进行交互，保证了私网路由的隔离和安全。

表1-6 配置 MCE 与站点之间使用 OSPF

操作	命令	说明
进入系统视图	**system-view**	-
创建MCE与站点间的OSPF实例，并进入OSPF视图	**ospf** [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*] *	必选 该配置在MCE上进行，站点上配置普通OSPF即可
配置OSPF域标识符	**domain-id** *domain-id* [**secondary**]	可选 缺省情况下，OSPF域标识符为0 该配置在MCE上进行，Site上配置普通OSPF即可
配置OSPF扩展团体属性的类型编码	**ext-community-type** { **domain-id** *type-code1* | **router-id** *type-code2* | **route-type** *type-code3* }	可选 缺省情况下，OSPF扩展团体属性Domain ID的类型编码是0x0005，Router ID的类型编码是0x0107，Route Type的类型编码是0x0306
配置route tag	**route-tag** *tag-value*	可选 缺省情况下，未配置route tag 在某些组网环境中，同一个VPN可能会连接多个MCE设备，当其中一个MCE将从BGP学到的路由向VPN内发布时，可能会被另外的MCE设备学到，造成路由环路。为避免路由环路，可以在MCE上为不同的VPN实例配置Route Tag，建议在多个MCE上为同一个VPN配置相同的route tag

操作	命令	说明
引入由PE发布的远端站点的路由	**import-route** *protocol* [*process-id* | **allow-ibgp**] [**cost** *cost* | **type** *type* | **tag** *tag* | **route-policy** *route-policy-name*] *	必选 缺省情况下，没有引入其他协议的路由信息
配置OSPF引入缺省路由	**default-route-advertise summary cost** *cost*	可选 缺省情况下，不引入缺省路由 该命令用于以Type-3 LSA引入缺省路由，MCE设备会将引入的缺省路由发布给站点内部
配置OSPF区域，进入OSPF区域视图	**area** *area-id*	必选 缺省情况下，没有配置OSPF区域
配置区域所包含的网段并在指定网段的接口上使能OSPF	**network** *ip-address wildcard-mask*	必选 缺省情况下，接口不属于任何区域且OSPF功能处于关闭状态

说明

- VPN 实例绑定的 OSPF 进程中不使用系统视图下配置的公网 Router ID，因此用户需要在启动进程时手工配置 Router ID。同一 VPN 的所有 OSPF 进程应配置相同的域 ID，以保证路由发布的正确性。
- 一个 OSPF 进程只能属于一个 VPN 实例，但一个 VPN 实例可以使用多个 OSPF 进程为其传播私网路由。同一 VPN 实例内的 OSPF 进程应配置有相同的域 ID，以保证路由发布的正确性。
- 有关 OSPF 协议的配置，请参见“三层技术-IP 路由配置指导”中的“OSPF”。

4. 配置MCE与站点之间使用IS-IS

一个 IS-IS 进程只能属于一个 VPN 实例。如果在启动 IS-IS 进程时不绑定到 VPN 实例，则该进程属于公网进程。

通过在 MCE 上将 IS-IS 进程与 VPN 实例绑定，可以使不同 VPN 内的私网路由通过不同的 IS-IS 进程在 Site 和 MCE 间进行交互，保证了私网路由的隔离和安全。

表1-7 配置 MCE 与站点之间使用 IS-IS

操作	命令	说明
进入系统视图	**system-view**	-
创建MCE与站点间的IS-IS实例，并进入IS-IS视图	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在MCE上进行，站点上配置普通IS-IS即可
配置网络实体名称	**network-entity** *net*	必选 缺省情况下，没有配置网络实体名称

操作	命令	说明
引入由PE发布的远端站点的路由	**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*] *	可选 缺省情况下，IS-IS不引入其它协议的路由信息 如果**import-route**命令中不指定引入的级别，则默认为引入路由到Level-2路由表中
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
使能接口IS-IS并指定要关联的IS-IS进程号	**isis enable** [*process-id*]	必选 缺省情况下，接口上没有使能IS-IS

说明

有关 IS-IS 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“IS-IS”。

5. 配置MCE与站点之间使用EBGP

MCE 与站点间使用 EBGP 交换路由信息时，需要在 MCE 上为每个 VPN 实例配置 BGP 对等体，并在站点上引入相应 VPN 内的 IGP 路由信息。

MCE 使用 EBGP 交换路由信息时，还可以通过 Filter-policy 对接收/发布的路由进行过滤。

(1) MCE 上的配置

表1-8 配置 MCE 与站点之间使用 EBGP

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	必选
配置EBGP对等体	**peer** { *group-name* | *ip-address* } [**as-number** *as-number*]	二者必选其一
配置允许本地AS号在所接收的路由的AS_PATH属性中出现，并可同时配置允许重复的次数	**peer** { *group-name* | *ip-address* } **allow-as-loop** [*number*]	
引入由PE发布的远端站点的路由	**import-route** *protocol* [*process-id* | **all-processes**] [**med** *med-value* | **route-policy** *route-policy-name*] *	必选 缺省情况下，BGP不引入且不通告其它协议的路由
配置对发布的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct | isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤

说明

- 通常情况下，BGP 通过 AS 号检测路由环路。如果在 MCE 和 Site 之间运行 EBGP，当 MCE 将路由信息通告给 Site 时带上本自治系统的 AS 号，再从 Site 接收路由更新时，路由更新消息中会带有本自治系统的 AS 号，这样 MCE 就不能接收这条路由更新信息，此时需要配置允许路由环路。
- 对于标准的 BGP/OSPF 互相引用功能，当在 MCE 设备上配置引入 BGP 路由到 OSPF 中时，该路由的原 OSPF 属性将无法恢复，造成该路由与从其他域引入的路由无法区分。为了区分原属于不同 OSPF 域的路由，需要在远端 PE 将 OSPF 路由引入到 BGP 时携带标识域的属性，即 OSPF 的域 ID（Domain ID）。OSPF 进程的域 ID 包含在此进程生成的路由中，在将 OSPF 路由引入 BGP 中时，域 ID 被附加到 BGP VPN 路由上，作为 BGP 的扩展团体属性传递。
- 配置 BGP VPN 实例后，VPN 实例中的 BGP 路由交换与普通的 BGP 相同。有关 BGP 协议的配置，请参见“三层技术-IP 路由配置指导”中的“BGP”。

(2) 站点上的配置

表1-9 站点上的配置

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
将MCE配置为EBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
配置引入VPN内的IGP路由	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	可选 站点需要将自己所能到达的VPN网段地址发布给接入的MCE

6. 配置MCE与站点间使用IBGP

MCE 与站点间使用 IBGP 交换路由信息时，需要在 MCE 上为每个 VPN 实例配置 BGP 对等体，并在站点上引入相应 VPN 内的 IGP 路由信息。

(1) MCE 上的配置

表1-10 配置 MCE 与站点之间使用 IBGP

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	必选
配置IBGP对等体	**peer** { *group-name* | *ip-address* } [**as-number** *as-number*]	必选

操作	命令	说明
配置本地设备作为路由反射器，对端设备作为路由反射器的客户端	**peer** { *group-name* | *ip-address* } **reflect-client**	可选 缺省情况下，没有配置路由反射器及其客户端
引入由PE发布的远端站点的路由	**import-route** *protocol* [*process-id* | **all-processes**] [**med** *med-value* | **route-policy** *route-policy-name*] *	必选 缺省情况下，BGP不引入且不通告其它协议的路由
配置对发布的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct** | **isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤

说明

将站点配置为 IBGP 对等体时，从该站点学习的 BGP 路由不会向其它 IBGP 对等体（包括 VPNv4 对等体）发送。只有将站点配置为路由反射器的客户端后，才能向其它 IBGP 对等体发送从该站点学习的路由。

(2) 站点上的配置

表1-11 站点上的配置

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
将MCE配置为IBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
配置引入VPN内的IGP路由	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	可选 站点需要将自己所能到达的VPN网段地址发布给接入的MCE

1.4.3 配置MCE与PE之间的路由交换

由于在 MCE 设备上已经将站点内的私网路由信息与 VPN 实例进行了绑定，因此，只需要在 MCE 与 PE 之间将接口与 VPN 实例进行绑定、进行简单的路由配置、并将 MCE 上维护的站点内的 VPN 路由引入到 MCE-PE 间的路由协议中，便可以实现私网 VPN 路由信息的传播。

1. 配置MCE与PE之间使用静态路由

表1-12 配置 MCE 与 PE 之间使用静态路由

操作	命令	说明
进入系统视图	**system-view**	-
为指定VPN实例配置静态路由	**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*]	必选
	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*]	
配置静态路由的缺省优先级	**ip route-static default-preference** *default-preference-value*	可选 缺省情况下，静态路由的缺省优先级为60

2. 配置MCE与PE之间使用RIP

表1-13 配置 MCE 与 PE 之间使用 RIP

操作	命令	说明
进入系统视图	**system-view**	-
创建MCE与PE间的RIP实例，并进入RIP视图	**rip** [*process-id*] **vpn-instance** *vpn-instance-name*	必选
在指定网段接口上使能RIP	**network** *network-address*	必选 缺省情况下，接口上的RIP功能处于关闭状态
引入站点内的VPN路由	**import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | **route-policy** *route-policy-name* | **tag** *tag*] *	必选 缺省情况下，RIP未引入其它路由
配置引入路由的缺省度量值	**default cost** *value*	可选 缺省情况下，引入路由的缺省度量值为0

说明

有关 RIP 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“RIP”。

3. 配置MCE与PE之间使用OSPF

表1-14 配置 MCE 与 PE 之间使用 OSPF

操作	命令	说明
进入系统视图	**system-view**	-
创建MCE与PE间的OSPF实例，并进入OSPF视图	**ospf** [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*] *	必选
关闭OSPF实例的路由环路检测功能	**vpn-instance-capability simple**	必选 缺省情况下，OSPF实例的路由环路检测功能处于开启状态 需要在MCE上通过本命令关闭OSPF实例的路由环路检测功能。否则，MCE不会接收PE发送过来的OSPF路由，导致路由丢失
配置OSPF域标识符	**domain-id** *domain-id* [**secondary**]	可选 缺省情况下，OSPF域标识符为0
引入站点内的VPN路由	**import-route** *protocol* [*process-id* | **allow-ibgp**] [**cost** *cost* | **type** *type* | **tag** *tag* | **route-policy** *route-policy-name*] *	必选 缺省情况下，没有引入其他协议的路由信息
配置对发布的路由进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [*protocol* [*process-id*]]	可选 缺省情况下，没有对发布的路由信息进行过滤
配置引入外部路由时的参数缺省值（开销、路由数量、标记、类型）	**default** { **cost** *cost* | **limit** *limit* | **tag** *tag* | **type** *type* } *	可选 缺省情况下，OSPF引入外部路由的度量值为1，一次引入外部路由数量的上限为1000，外部路由标记值为1，引入的外部路由类型为Type2
配置OSPF区域，进入OSPF区域视图	**area** *area-id*	必选 缺省情况下，没有配置OSPF区域
配置区域所包含的网段并在指定网段的接口上使能OSPF	**network** *ip-address wildcard-mask*	必选 缺省情况下，接口不属于任何区域且OSPF功能处于关闭状态

说明

有关 OSPF 协议的配置，请参见“三层技术-IP 路由配置指导”中的“OSPF”。

4. 配置MCE与PE之间使用IS-IS

表1-15 配置 MCE 与 PE 之间使用 IS-IS

操作	命令	说明
进入系统视图	**system-view**	-
创建MCE与PE间的IS-IS实例，并进入IS-IS视图	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	必选

操作	命令	说明
配置网络实体名称	**network-entity** *net*	必选 缺省情况下，没有配置网络实体名称
引入站点内的VPN路由	**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*] *	可选 缺省情况下，IS-IS不引入其它协议的路由信息 如果**import-route**命令中不指定引入的级别，则默认为引入路由到Level-2路由表中
对发布的路由进行过滤	**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**]	可选 缺省情况下，IS-IS不对发布的路由信息进行过滤
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
使能接口IS-IS并指定要关联的IS-IS进程号	**isis enable** [*process-id*]	必选 缺省情况下，接口上没有使能IS-IS

说明

有关 IS-IS 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“IS-IS”。

5. 配置MCE与PE之间使用EBGP

表1-16 配置 MCE 与 PE 之间使用 EBGP

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	必选
将PE配置为EBGP对等体	**peer** { *group-name* | *ip-address* } [**as-number** *as-number*]	必选
引入站点内的VPN路由	**import-route** *protocol* [*process-id* | **all-processes**] [**med** *med-value* | **route-policy** *route-policy-name*] *	必选 缺省情况下，BGP不引入且不通告其它协议的路由
配置对发布的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct | isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤

说明

配置 BGP VPN 实例后，VPN 实例中的 BGP 路由交换与普通的 BGP 相同。有关 BGP 协议的配置，请参见“三层技术-IP 路由配置指导”中的“BGP”。

6. 配置MCE与PE之间使用IBGP

表1-17 配置 MCE 与 PE 之间使用 IBGP

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	必选
将PE配置为IBGP对等体	**peer** { *group-name* | *ip-address* } [**as-number** *as-number*]	必选
引入站点内的VPN路由	**import-route** *protocol* [*process-id* | **all-processes**] [**med** *med-value* | **route-policy** *route-policy-name*] *	必选 缺省情况下，BGP不引入且不通告其它协议的路由
将站点出口路由器配置为路由反射器的客户端	**peer** { *group-name* | *ip-address* } **reflect-client**	可选 缺省情况下，没有配置路由反射器及其客户端
配置允许客户端到客户端的路由反射	**reflect between-clients**	可选 缺省情况下，允许客户端到客户端的路由反射 如果客户之间是全连接的，则不需要配置路由反射
配置路由反射器的集群ID	**reflector cluster-id** *cluster-id*	可选 缺省情况下，每个路由反射器都使用自己的Router ID作为群ID 如果一个群中配置了多个路由反射器，请使用本命令为所有的路由反射器配置相同的群ID，以避免产生路由环路
配置对发布的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct** | **isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤

1.5 MCE显示和维护

1.5.1 复位BGP连接

当 BGP 配置变化后，可以通过软复位或复位 BGP 连接使新的配置生效。软复位需要 BGP 对等体具备路由刷新能力（支持 ROUTE-REFRESH 消息）。

请在用户视图下进行下列操作。

表1-18 复位 BGP 连接

操作	命令
软复位指定VPN实例的BGP连接	**refresh bgp vpn-instance** *vpn-instance-name* { *ip-address* | **all** | **external** | **group** *group-name* } { **export** | **import** }
复位指定VPN实例的BGP连接	**reset bgp vpn-instance** *vpn-instance-name* { *as-number* | *ip-address* | **all** | **external** | **group** *group-name* }

1.5.2 显示MCE的运行状态

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 MPLS L3VPN 的运行情况，通过查看显示信息验证配置的效果。

在用户视图下执行 **reset** 命令可以清除路由振荡衰减信息。

表1-19 显示 MCE 的运行状态

操作	命令
显示与VPN实例相关联的IP路由表	**display ip routing-table vpn-instance** *vpn-instance-name* [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例信息	**display ip vpn-instance** [**instance-name** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例的FIB信息	**display fib vpn-instance** *vpn-instance-name* [**acl** *acl-number* | **ip-prefix** *ip-prefix-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例中与指定目的IP地址匹配的FIB信息	**display fib vpn-instance** *vpn-instance-name ip-address* [*mask* | *mask-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示BGP VPNv4对等体组信息	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **group** [*group-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示引入的BGP VPNv4路由信息	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **network** [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示BGP VPNv4的AS路径信息	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **paths** [*as-regular-expression* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]
显示BGP VPNv4对等体信息	**display bgp vpnv4 vpn-instance** *vpn-instance-name* **peer** [*group-name* **log-info** | *ip-address* { **log-info** | **verbose** } | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]

操作	命令
显示指定VPN实例的BGP VPNv4路由信息	**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*]
清除指定VPN实例的路由振荡衰减信息	**reset bgp vpn-instance** *vpn-instance-name* **dampening** [*network-address* [*mask* | *mask-length*]
清除指定VPN实例的BGP对等体路由振荡历史信息	**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*]

说明

有关查看路由表的命令介绍请参见“三层技术-IP 路由命令参考”中的“IP 路由基础”。

1.6 MCE典型配置举例

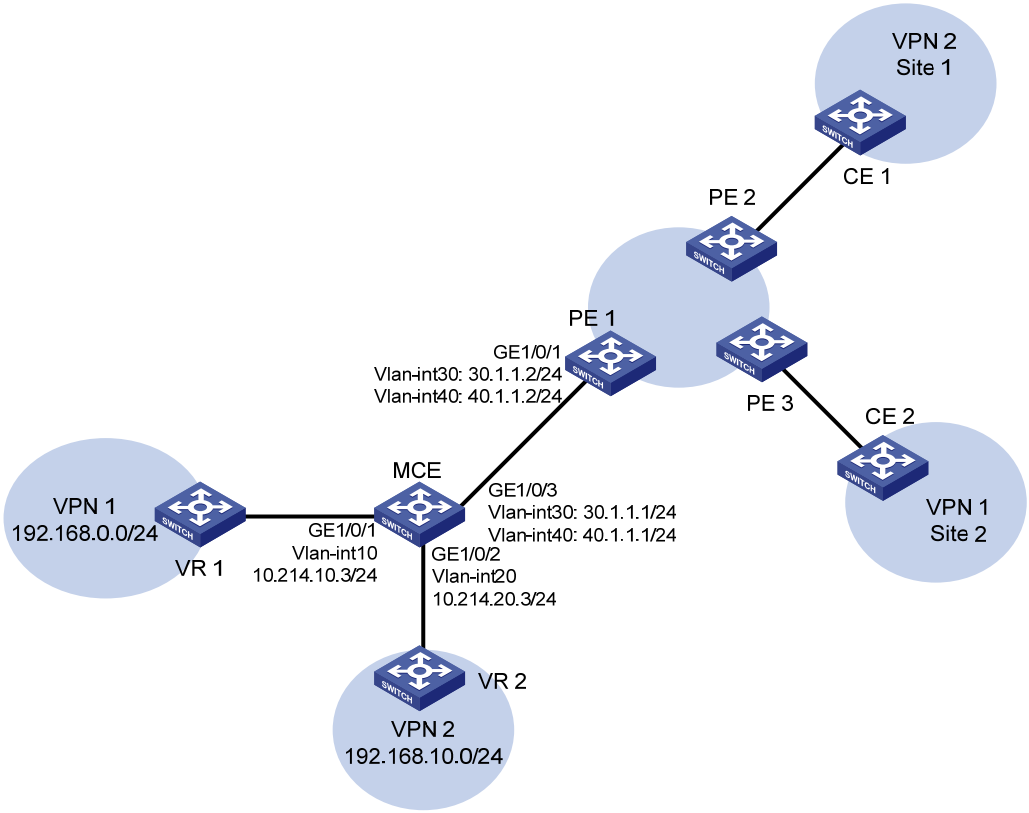
1.6.1 MCE和PE间使用OSPF引入VPN路由

1. 组网需求

- MCE 设备通过 Vlan-interface10 接口（IP 地址 10.214.10.3）连接到地址范围为 192.168.0.0/24 的 VPN 1，通过 Vlan-interface20 接口（IP 地址 10.214.20.3）连接到地址范围为 192.168.10.0/24 的 VPN 2，其中 VPN 2 内运行 RIP 路由协议。
- 要求 MCE 设备能够将 VPN 之间的路由隔离，并通过 OSPF 将各 VPN 的路由发布到 PE 1。

2. 组网图

图1-6 MCE 和 PE 间使用 OSPF 引入 VPN 路由组网示意图



3. 配置步骤

为区分设备，假设 MCE 系统名为“MCE”，VPN 1 和 VPN 2 的边缘路由器分别名为“VR1”和“VR2”，PE 1 系统名为 “PE1”。

(1) 在 MCE 和 PE 1 上配置 VPN 实例

在 MCE 设备上配置 VPN 实例，名称分别为 vpn1 和 vpn2，RD 分别取值为 10:1 和 20:1，VPN Target 取值与 RD 取相同数值，Export 和 Import 均取此值。

```
<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
```

创建 VLAN10，将端口 GigabitEthernet1/0/1 加入 VLAN10，并创建 Vlan-interface10 接口。

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

配置 Vlan-interface10 接口与 VPN 实例 vpn1 绑定，并配置接口的 IP 地址。

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

```
[MCE-Vlan-interface10] ip address 10.214.10.3 24
```
使用类似步骤配置 VLAN20，将端口 GigabitEthernet1/0/2 加入 VLAN20，配置接口与 VPN 实例 vpn2 绑定并配置 IP 地址。
```
[MCE-Vlan-interface10] quit
[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] ip address 10.214.20.3 24
[MCE-Vlan-interface20] quit
```
在 PE 1 上配置 VPN 实例，名称分别为 VPN1 和 VPN2，RD 分别取值为 30:1 和 40:1，VPN Target 取值分别为 10:1 和 20:1，Export 和 Import 均取此值。
```
<PE1> system-view
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 30:1
[PE1-vpn-instance-vpn1] vpn-target 10:1
[PE1-vpn-instance-vpn1] quit
[PE1] ip vpn-instance vpn2
[PE1-vpn-instance-vpn2] route-distinguisher 40:1
[PE1-vpn-instance-vpn2] vpn-target 20:1
[PE1-vpn-instance-vpn2] quit
```
(2) MCE 与 Site 间路由配置

MCE 与 VPN 1 直接相连，且 VPN 1 内未使用路由协议，因此可以使用静态路由进行配置。

配置 VR1 与 MCE 连接的接口地址为 10.214.10.2/24，连接 VPN1 接口的地址为 192.168.0.1/24。向 VLAN 中增加端口和配置接口 IP 地址的过程省略。

在 VR1 上配置缺省路由，指定出方向报文的下一跳地址为 10.214.10.3。
```
<VR1> system-view
[VR1] ip route-static 0.0.0.0 0.0.0.0 10.214.10.3
```
在 MCE 上指定静态路由，去往 192.168.0.0/24 网段的报文，下一跳地址为 10.214.10.2，并将此路由与 VPN 实例 vpn1 绑定。
```
[MCE] ip route-static vpn-instance vpn1 192.168.0.0 24 10.214.10.2
```
显示 MCE 上为 VPN 实例 vpn1 维护的路由信息。
```
[MCE] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

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
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
192.168.0.0/24      Static 60   0            10.214.10.2     Vlan10
```
可以看到，已经在 MCE 上为 VPN 1 指定了静态路由。

VPN 2 内运行 RIP，在 MCE 上配置 RIP 进程 20，并与 VPN 实例 vpn2 绑定，以便将 VPN 2 内的路由学习到 VPN 实例 vpn2 的路由表中。
```
[MCE] rip 20 vpn-instance vpn2
```

发布网段 10.214.20.0 的路由。

```
[MCE-rip-20] network 10.214.20.0
[MCE-rip-20] quit
```

在 VR2 上，配置与 MCE 连接的接口地址为 10.214.20.2/24，连接 VPN 2 接口的地址为 192.168.10.1/24。（配置过程略）

配置 RIP，发布网段 192.168.10.0 和 10.214.20.0 的路由。

```
<VR2> system-view
[VR2] rip 20
[VR2-rip-20] network 192.168.10.0
[VR2-rip-20] network 10.214.20.0
```

在 MCE 上查看 VPN 实例 vpn2 的路由信息。

```
[MCE] 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
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
```

可以看到，MCE 已经通过 RIP 学习到了 VPN 2 内的私网路由，并与 VPN 1 内的 192.168.0.0 路由信息分别维护在两个路由表内，有效进行了隔离。

(3) MCE 与 PE 间路由配置

MCE 使用 GigabitEthernet1/0/3 端口连接到 PE 1 的 GigabitEthernet1/0/1 端口，需要配置这两个端口为 Trunk 端口，并允许 VLAN 30 和 VLAN 40 的报文携带 Tag 通过。

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

配置 PE 1 的 GigabitEthernet1/0/1 端口。

```
[PE1] interface gigabitethernet 1/0/1
[PE1-GigabitEthernet1/0/1] port link-type trunk
[PE1-GigabitEthernet1/0/1] port trunk permit vlan 30 40
[PE1-GigabitEthernet1/0/1] quit
```

在 MCE 上创建 VLAN 30 和接口 Vlan-interface30，配置接口与 VPN 实例 vpn1 绑定，并配置接口的 IP 地址。

```
[MCE] vlan 30
[MCE-vlan30] quit
[MCE] interface vlan-interface 30
[MCE-Vlan-interface30] ip binding vpn-instance vpn1
[MCE-Vlan-interface30] ip address 30.1.1.1 24
[MCE-Vlan-interface30] quit
```

在 MCE 上创建 VLAN 40 和接口 Vlan-interface40，配置接口与 VPN 实例 vpn2 绑定，并配置接口的 IP 地址。

```
[MCE] vlan 40
[MCE-vlan40] quit
```

```
[MCE] interface vlan-interface 40
[MCE-Vlan-interface40] ip binding vpn-instance vpn2
[MCE-Vlan-interface40] ip address 40.1.1.1 24
[MCE-Vlan-interface40] quit
```

在 PE 1 上创建 VLAN 30 和接口 Vlan-interface30，配置接口与 VPN 实例 vpn1 绑定，并配置接口的 IP 地址。

```
[PE1] vlan 30
[PE1-vlan30] quit
[PE1] interface vlan-interface 30
[PE1-Vlan-interface30] ip binding vpn-instance vpn1
[PE1-Vlan-interface30] ip address 30.1.1.2 24
[PE1-Vlan-interface30] quit
```

在 PE 1 上创建 VLAN 40 和接口 Vlan-interface40，配置接口与 VPN 实例 vpn2 绑定，并配置接口的 IP 地址。

```
[PE1] vlan 40
[PE1-vlan40] quit
[PE1] interface vlan-interface 40
[PE1-Vlan-interface40] ip binding vpn-instance vpn2
[PE1-Vlan-interface40] ip address 40.1.1.2 24
[PE1-Vlan-interface40] quit
```

配置 MCE 和 PE 1 的 Loopback0 接口，用于指定 MCE 和 PE 1 的 Router ID，地址分别为 101.101.10.1 和 100.100.10.1。配置步骤这里省略。

配置 MCE 启动 OSPF 进程 10，配置绑定到 VPN 实例 vpn1，域 ID 设置为 10。

```
[MCE] ospf 10 router-id 101.101.10.1 vpn-instance vpn1
[MCE-ospf-10] vpn-instance-capability simple
[MCE-ospf-10] domain-id 10
```

在 Area0 区域发布 30.1.1.0 网段，并引入 VPN 1 的静态路由。

```
[MCE-ospf-10] area 0
[MCE-ospf-10-area-0.0.0.0] network 30.1.1.0 0.0.0.255
[MCE-ospf-10-area-0.0.0.0] quit
[MCE-ospf-10] import-route static
```

配置 PE 1 启动 OSPF 进程 10，绑定到 VPN 实例 vpn1，域 ID 为 10，在 Area0 区域发布 30.1.1.0 网段。

```
[PE1] ospf 10 router-id 100.100.10.1 vpn-instance vpn1
[PE1-ospf-10] domain-id 10
[PE1-ospf-10] area 0
[PE1-ospf-10-area-0.0.0.0] network 30.1.1.0 0.0.0.255
[PE1-ospf-10-area-0.0.0.0] quit
[PE1-ospf-10] quit
```

显示 PE 1 上的 VPN 1 路由信息。

```
[PE1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

30.1.1.0/24         Direct 0    0            30.1.1.2        Vlan30
30.1.1.2/32         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
192.168.0.0/24      O_ASE  150  1           30.1.1.1        Vlan30
```

可以看到，VPN 1 内的静态路由已经引入到 MCE 与 PE 1 间的 OSPF 路由表中。

MCE 与 PE 1 间配置 OSPF 进程 20，导入 VPN 实例 vpn2 的路由信息的过程与上面介绍的配置基本一致，不同的是在 MCE 的 OSPF 中配置导入的是 RIP 进程 20 的路由，这里不再赘述，只以显示信息为例表示导入成功后的结果。

```
[PE1] display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

40.1.1.0/24         Direct 0    0           40.1.1.2        Vlan40
40.1.1.2/32         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     O_ASE  150  1           40.1.1.1        Vlan40
```

至此，通过配置，已经将两个 VPN 实例内的路由信息完整地传播到 PE 1 中，配置完成。

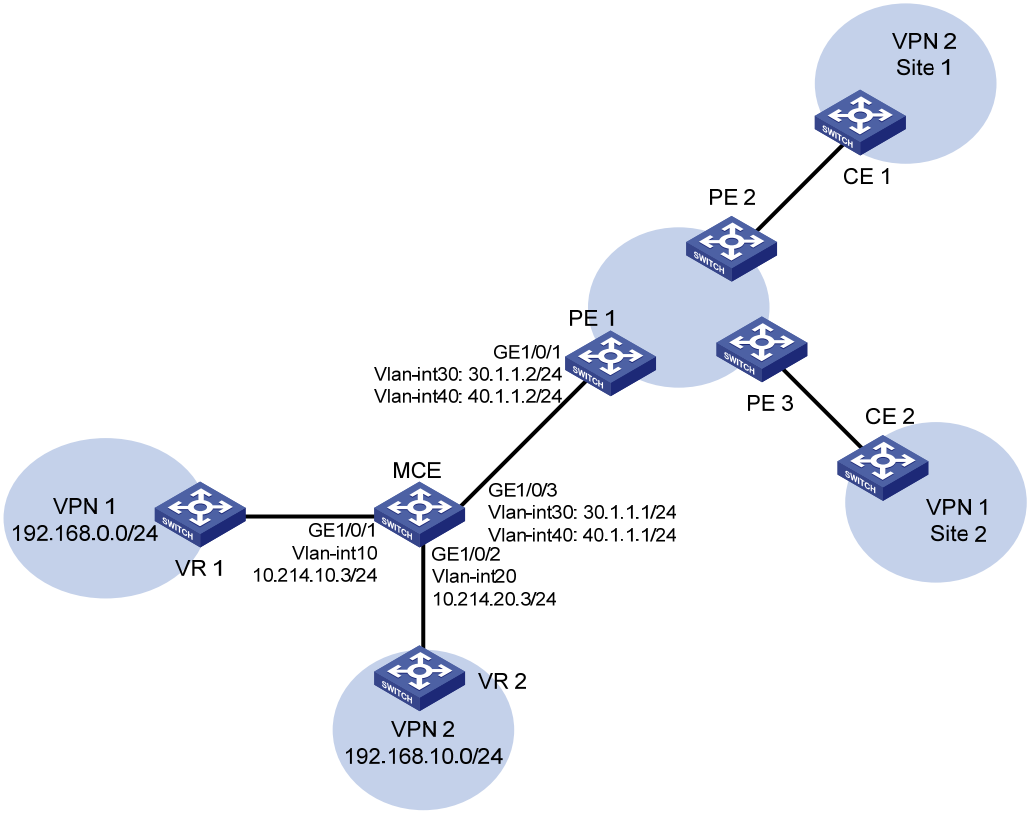
1.6.2 MCE和PE间使用BGP引入VPN路由

1. 组网需求

- 使用本系列交换机作为 MCE 设备，将站点 1 和站点 2 内的 VPN 路由发布到 PE 1，使 MPLS 骨干网络两端的 VPN 能够正常通信。
- 站点 1 和站点 2 内部均使用 OSPF 协议，MCE 与 PE 间使用 EBGP 协议。

2. 组网图

图1-7 MCE 和 PE 间使用 BGP 引入 VPN 路由组网示意图



3. 配置步骤

(1) VPN 实例配置

在MCE和PE 1 上创建VPN实例，并与接口绑定的配置与“1.6.1 MCE和PE间使用OSPF引入VPN路由”中的配置类似，这里不再赘述。

(2) MCE 与 Site 间路由配置

配置两个 VPN 实例内的设备启动 OSPF 进程，发布各网段的路由，操作与普通 OSPF 配置相同，这里不再赘述。

配置 MCE 的 OSPF 协议，进程 10 与 VPN 实例 vpn1 绑定，学习 VPN 1 内的路由。

```
<MCE> system-view
[MCE] ospf router-id 10.214.10.3 10 vpn-instance vpn1
[MCE-ospf-10] area 0
[MCE-ospf-10-area-0.0.0.0] network 10.214.10.0 0.0.0.255
```

显示 VPN 1 的路由信息。

```
[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

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
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
192.168.0.0/24      OSPF   10   1           10.214.10.2     Vlan10
```

可以看到，MCE 已经通过 OSPF 进程 10 学习到了 VPN 1 内的路由。

配置 MCE 的 OSPF 进程 20 与 VPN 实例 vpn2 绑定，学习 VPN 2 内的路由，配置与上面配置 OSPF 进程 10 类似。这里直接显示配置结果。

```
[MCE] 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
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      OSPF   10   1           10.214.20.2     Vlan20
```

(3) MCE 与 PE 间路由配置

配置MCE与PE 1 间通过Trunk端口进行连接，配置与“1.6.1 MCE和PE间使用OSPF引入VPN路由”中的配置类似，这里不再赘述。

配置 MCE 启动 BGP 进程 100，并进入 VPN 实例 vpn1 的 IPv4 地址族视图。

```
[MCE] bgp 100
[MCE-bgp] ipv4-family vpn-instance vpn1
```

指定 PE 1（PE 1 与实例 vpn1 绑定的接口地址为 30.1.1.2/24，BGP 进程为 200）为 EBGP 对等体，并引入 OSPF 进程 10 的路由信息。

```
[MCE-bgp-vpn1] peer 30.1.1.2 as-number 200
[MCE-BGP-vpn1] import-route ospf 10
```

在 PE 1 上配置 BGP200，并指定 MCE 为 EBGP 对等体。

```
<PE1> system-view
[PE1] bgp 200
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] peer 30.1.1.1 as-number 100
```

显示 PE 1 上 VPN 实例 vpn1 的路由信息。

```
[PE1-bgp-vpn1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 5        Routes : 5

Destination/Mask    Proto  Pre  Cost         NextHop         Interface

30.1.1.0/24         Direct 0    0           30.1.1.2        Vlan30
30.1.1.2/32         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
192.168.0.0/24      BGP    255  2           30.1.1.1        Vlan30
```

对于 VPN 2，在 MCE 和 PE 1 上进行类似的配置，将 VPN 实例 vpn2 的 OSPF 路由信息引入到 EBGP 路由中。这里不再叙述配置过程，直接显示配置结果。

```
[PE1-bgp-vpn1] display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 5        Routes : 5
```

```
Destination/Mask    Proto   Pre  Cost        NextHop         Interface

40.1.1.0/24         Direct  0    0           40.1.1.2        Vlan40
40.1.1.2/32         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     BGP     255  2           40.1.1.1        Vlan40
```

至此，MCE 设备已经将两个 VPN 实例内的 OSPF 路由全部引入 PE 1 的 EBGP 路由表中，配置完成。

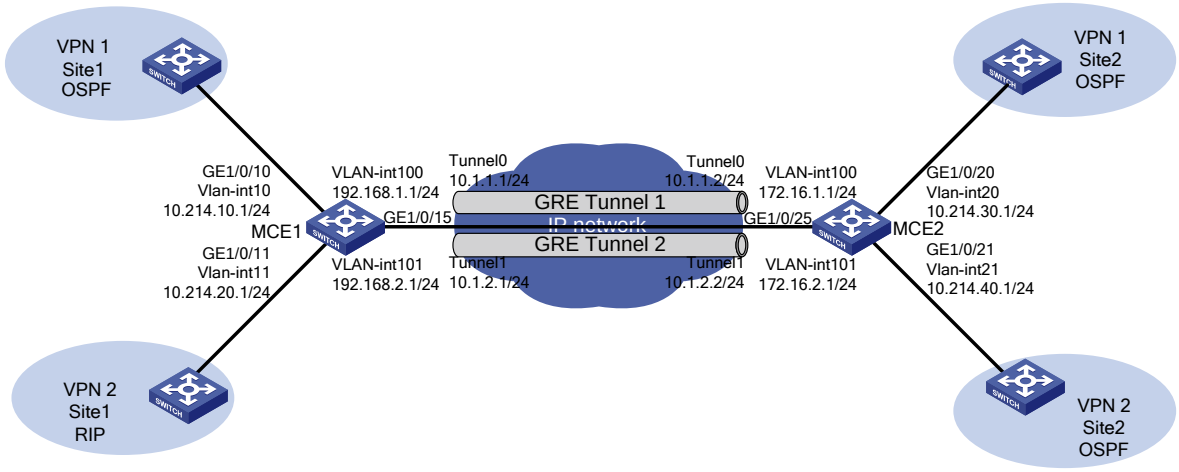
1.6.3 在隧道组网中使用MCE功能发布VPN路由

1. 组网需求

如 图 1-8 所示，MCE1 和MCE2 设备通过GRE隧道进行通信，并分别连接到VPN1 和VPN2 的站点。其中VPN1 的站点全部使用OSPF作为路由协议，且均只存在骨干区域，即区域 0。而VPN2 的站点分别使用RIP和OSPF，其中OSPF部分仅存在骨干区域。现要求MCE1 和MCE2 设备能够通过隧道将两个VPN网络中的路由信息进行准确发布。

2. 组网图

图1-8 在隧道组网中使用 MCE 功能发布 VPN 路由组网示意图



3. 配置思路

由于已经为每个VPN网络单独配置了一条GRE隧道，用于传输该VPN网络中的数据和路由信息，因此可以通过创建VPN实例并将特定的接口与VPN实例进行绑定，使原有网络简化为两个独立的拓扑结构（如 图 1-9 和 图 1-10 所示），从而使MCE设备可以将每个VPN网络的路由通过独立路径进行发布。

图1-9 VPN1 的站点与 MCE 设备的网络拓扑示意图

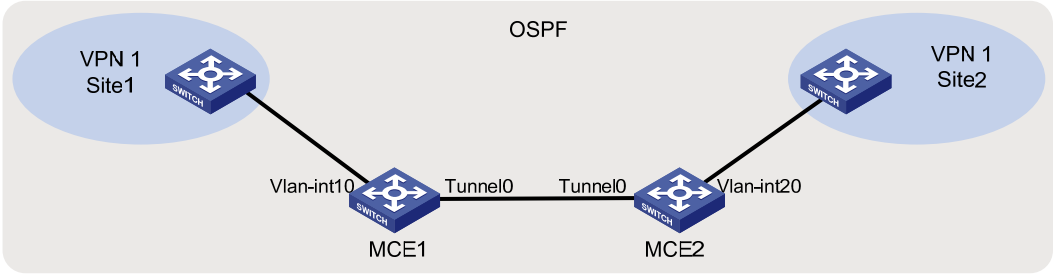
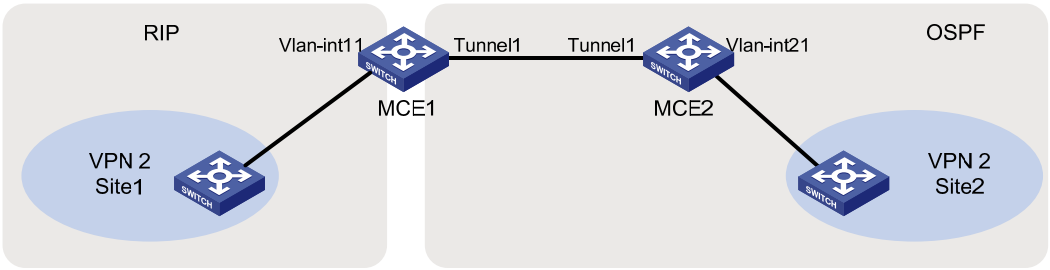


图1-10 VPN2 的站点与 MCE 设备的网络拓扑示意图



根据以上描述，本例中需要在 MCE 设备上为每个 VPN 网络创建 VPN 实例，并将 MCE 与 VPN 站点相连的接口和隧道接口与 VPN 实例进行绑定。对于 VPN1 网络，可以将两个 MCE 设备上的接口全部加入到站点的 OSPF 计算中，将整个网络作为一个 OSPF 域进行维护；对于 VPN2 网络，则还需要在 MCE1 上进行 OSPF 和 RIP 的相互引用配置。

4. 配置步骤

(1) 配置隧道

- 配置 MCE1

创建 VLAN100 和 VLAN101，并将端口 GigabitEthernet1/0/15 配置为 Trunk 类型，加入这两个 VLAN。

```
<MCE1> system-view
[MCE1] vlan 100 to 101
[MCE1] interface Gigabitethernet 1/0/15
[MCE1-GigabitEthernet1/0/15] port link-type trunk
[MCE1-GigabitEthernet1/0/15] port trunk permit vlan 100 101
[MCE1-GigabitEthernet1/0/15] quit
```

创建 VLAN 接口，并配置相应的 IP 地址。

```
[MCE1] interface vlan-interface 100
[MCE1-Vlan-interface100] ip address 192.168.1.1 255.255.255.0
[MCE1-Vlan-interface100] quit
[MCE1] interface vlan-interface 101
[MCE1-Vlan-interface101] ip address 192.168.2.1 255.255.255.0
[MCE1-Vlan-interface101] quit
```

创建 Tunnel0 接口。

```
[MCE1] interface tunnel 0
```

配置 Tunnel0 接口的 IP 地址。

```
[MCE1-Tunnel0] ip address 10.1.1.1 255.255.255.0
```

配置 Tunnel 封装模式。

```
[MCE1-Tunnel0] tunnel-protocol gre
```

配置 Tunnel0 接口的源地址。

```
[MCE1-Tunnel0] source vlan-interface 100
```

配置 Tunnel0 接口的目的地址。

```
[MCE1-Tunnel0] destination 172.16.1.1
[MCE1-Tunnel0] quit
```

创建并配置业务环回组 1，服务类型为 tunnel。

```
[MCE1] service-loopback group 1 type tunnel
```

将任意未使用端口（此处以 GigabitEthernet1/0/3 为例）加入业务环回组 1。

```
[MCE1] interface Gigabitethernet 1/0/3
[MCE1-GigabitEthernet1/0/3] undo stp enable
[MCE1-GigabitEthernet1/0/3] port service-loopback group 1
```

在 Tunnel 接口视图下指定隧道引用的业务环回组 1。

```
[MCE1-GigabitEthernet1/0/3] quit
[MCE1] interface tunnel 0
[MCE1-Tunnel0] service-loopback-group 1
[MCE1-Tunnel0] quit
```

创建 Tunnel1 接口。

```
[MCE1] interface tunnel 1
```

配置 Tunnel1 接口的 IP 地址。

```
[MCE1-Tunnel1] ip address 10.1.2.1 255.255.255.0
```

配置 Tunnel 封装模式。

```
[MCE1-Tunnel1] tunnel-protocol gre
```

配置 Tunnel1 接口的源地址。

```
[MCE1-Tunnel1] source vlan-interface 101
```

配置 Tunnel1 接口的目的地址。

```
[MCE1-Tunnel1] destination 172.16.2.1
```

在 Tunnel 接口视图下指定隧道引用的业务环回组 1。

```
[MCE1-Tunnel1] service-loopback-group 1
[MCE1-Tunnel1] quit
```

- 配置 MCE2

创建 VLAN100 和 VLAN101，并将端口 GigabitEthernet1/0/25 配置为 Trunk 类型，加入这两个 VLAN。

```
<MCE2> system-view
[MCE2] vlan 100 to 101
[MCE2] interface gigabitethernet 1/0/25
[MCE2-GigabitEthernet1/0/25] port link-type trunk
[MCE2-GigabitEthernet1/0/25] port trunk permit vlan 100 101
[MCE2-GigabitEthernet1/0/25] quit
```

创建 VLAN 接口，并配置相应的 IP 地址。

```
[MCE2] interface vlan-interface 100
[MCE2-Vlan-interface100] ip address 172.16.1.1 255.255.255.0
[MCE2-Vlan-interface100] quit
[MCE2] interface vlan-interface 101
[MCE2-Vlan-interface101] ip address 172.16.2.1 255.255.255.0
[MCE2-Vlan-interface101] quit
```

创建 Tunnel0 接口。

```
[MCE2] interface tunnel 0
```

配置 Tunnel0 接口的 IP 地址。

```
[MCE2-Tunnel0] ip address 10.1.1.2 255.255.255.0
```

配置 Tunnel 封装模式。

```
[MCE2-Tunnel0] tunnel-protocol gre
```

配置 Tunnel0 接口的源地址。

```
[MCE2-Tunnel0] source vlan-interface 100
```

配置 Tunnel0 接口的目的地址。

```
[MCE2-Tunnel0] destination 192.168.1.1
[MCE2-Tunnel0] quit
```

创建并配置业务环回组 1，服务类型为 tunnel。

```
[MCE2] service-loopback group 1 type tunnel
```

将任意未使用端口（此处以 GigabitEthernet1/0/3 为例）加入业务环回组 1。

```
[MCE2] interface gigabitethernet 1/0/3
[MCE2-GigabitEthernet1/0/3] undo stp enable
[MCE2-GigabitEthernet1/0/3] port service-loopback group 1
```

在 Tunnel 接口视图下指定隧道引用的业务环回组 1。

```
[MCE2-GigabitEthernet1/0/3] quit
[MCE2] interface tunnel 0
[MCE2-Tunnel0] service-loopback-group 1
[MCE2-Tunnel0] quit
```

创建 Tunnel1 接口。

```
[MCE2] interface tunnel 1
```

配置 Tunnel1 接口的 IP 地址。

```
[MCE2-Tunnel1] ip address 10.1.2.2 255.255.255.0
```

配置 Tunnel 封装模式。

```
[MCE2-Tunnel1] tunnel-protocol gre
```

配置 Tunnel1 接口的源地址。

```
[MCE2-Tunnel1] source vlan-interface 101
```

配置 Tunnel1 接口的目的地址。

```
[MCE2-Tunnel1] destination 192.168.2.1
```

在 Tunnel 接口视图下指定隧道引用的业务环回组 1。

```
[MCE2-Tunnel1] service-loopback-group 1
```

(1) 配置 VPN 实例

- MCE1 设备的配置

为 VPN1 网络创建 VPN 实例，命名为“vpn1”，并配置 RD 属性。

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

为 VPN2 网络创建 VPN 实例，命名为“vpn2”，并配置 RD 属性。

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

```
[MCE1-vpn-instance-vpn2] quit
```

配置 Vlan-interface 10 接口和 Tunnel 0 接口与 VPN 实例“vpn1”进行绑定，并配置 IP 地址。

```
[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
```

配置 Vlan-interface 11 接口和 Tunnel 1 接口与 VPN 实例“vpn2”进行绑定，并配置 IP 地址。

```
[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
```

- MCE2 设备的配置

为 VPN1 网络创建 VPN 实例，命名为“vpn1”，并配置 RD 属性（与 MCE1 设备保持一致）。

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

为 VPN2 网络创建 VPN 实例，命名为“vpn2”，并配置 RD 属性（与 MCE1 设备保持一致）。

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

配置 Vlan-interface 20 接口和 Tunnel 0 接口与 VPN 实例“vpn1”进行绑定，并配置 IP 地址。

```
[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
```

配置 Vlan-interface 21 接口和 Tunnel 1 接口与 VPN 实例“vpn2”进行绑定，并配置 IP 地址。

```
[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
```

(2) 配置路由协议

- 发布 VPN1 网络的路由

在 MCE1 设备上配置 OSPF 路由协议并与 VPN 实例“vpn1”进行绑定，开启 OSPF 支持 MCE 功能。需要注意的是，下面配置中的 OSPF 区域要与站点内的 OSPF 配置保持一致，此处以区域 0 为例进行配置。

```
[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]
```

将 MCE1 设备的 Vlan-interface10 和 Tunnel 0 接口的地址进行发布。

```
[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
```

在 MCE2 设备上配置 OSPF 路由协议并与 VPN 实例“vpn1”进行绑定，开启 OSPF 支持 MCE 功能。

```
[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]
```

将 MCE2 设备的 Vlan-interface20 和 Tunnel 0 接口的地址进行发布。

```
[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
```

- 发布 VPN2 网络的路由

在 MCE1 设备上配置 OSPF 路由协议并与 VPN 实例“vpn2”进行绑定，开启 OSPF 支持 MCE 功能。需要注意的是，下面配置中的 OSPF 区域要与站点 2 内的 OSPF 配置保持一致，此处以区域 0 为例进行配置。

```
[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]
```

将 MCE1 设备 Tunnel 1 接口的地址进行发布，使其参与 OSPF 域中的路由计算。

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

在 MCE1 设备上配置 RIP 路由协议并与 VPN 实例“vpn2”进行绑定，此处以 RIP 进程号 1 为例进行配置。

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

将 MCE1 设备 Vlan-interface11 接口的地址进行发布，使其参与站点 1 的 RIP 计算。

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

在当前 RIP 进程中引入 OSPF 进程 2 学到的路由信息。

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

在 OSPF 进程 2 中引入 RIP 进程 1 学到的路由信息。

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

在 MCE2 设备上配置 OSPF 路由协议并与 VPN 实例“vpn2”进行绑定，开启 OSPF 支持 MCE 功能。需要注意的是，下面配置中的 OSPF 区域要与站点 2 内的 OSPF 配置保持一致，此处以区域 0 为例进行配置。

```
[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]
```

将 MCE2 设备的 Vlan-interface21 和 Tunnel 1 接口的地址进行发布。

```
[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
```

2 IPv6 MCE配置

2.1 IPv6 MCE简介

在 MPLS L3VPN 组网环境中，MCE 利用静态或动态路由协议在 VPN 站点和 PE 之间发布 IPv4 路由，并交互 IPv4 报文。

在IPv6 MPLS L3VPN组网环境中，IPv6 MCE在VPN站点和PE之间发布IPv6 路由，并交互IPv6 报文。IPv6 MCE的工作原理与MCE相同，详细内容请参见 1.1.4 MCE工作原理。

2.2 配置IPv6 MCE

2.2.1 配置VPN实例

VPN 实例不仅可以隔离 VPN 私网路由与公网路由，还可以隔离不同 VPN 实例的路由，这一特点使得 VPN 实例的使用不限于 MPLS L3VPN。

1. 创建VPN实例

VPN 实例在实现中与 Site 关联。VPN 实例不是直接对应于 VPN，一个 VPN 实例综合了和它所对应 Site 的 VPN 成员关系和路由规则。

一个 VPN 实例只有配置了 RD 后才生效。

描述信息用于描述 VPN 实例，可以用来记录 VPN 实例与某个 VPN 的关系等信息。

表2-1 创建 VPN 实例

操作	命令	说明
进入系统视图	**system-view**	-
创建VPN实例，并进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	必选
配置VPN实例的RD	**route-distinguisher** *route-distinguisher*	必选
配置VPN实例的描述信息	**description** *text*	可选

2. 配置VPN实例与接口关联

VPN 实例配置完成后，需要与连接各个 VPN 站点以及 PE 设备的接口进行关联。

表2-2 配置 VPN 实例与接口关联

操作	命令	说明
进入系统视图	**system-view**	-
进入要关联接口的接口视图	**interface** *interface-type interface-number*	-
将当前接口与VPN实例关联	**ip binding vpn-instance** *vpn-instance-name*	必选 缺省情况下，接口不关联任何VPN实例

说明

- 配置三层聚合接口与 VPN 实例关联时，还需要同时配置其所有成员端口与相同的 VPN 实例关联，否则会影响流量的正常转发。
- 执行 **ip binding vpn-instance** 命令将删除接口上已经配置的 IPv6 地址，因此需要重新配置接口的 IPv6 地址。

3. 配置VPN实例的路由相关属性

MCE 设备对 VPN 路由的发布控制过程如下：

- 当从 VPN 站点内学习到的一条 VPN 路由引入 BGP 时，BGP 为它关联一个 VPN Target 扩展团体属性列表，通常这个列表是与该站点相关联的 VPN 实例的输出路由属性列表。
- VPN 实例根据 VPN Target 中 **import-extcommunity** 确定可被接受并引入此 VPN 实例的路由。
- VPN 实例根据 VPN Target 中的 **export-extcommunity** 对向外发布的路由进行 VPN Target 属性的修改。

表2-3 配置 VPN 实例的路由相关属性

操作	命令	说明
进入系统视图	**system-view**	-
进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	-
进入IPv6 VPN视图	**ipv6-family**	可选
配置VPN Target	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	必选
配置支持的最大路由数	**routing-table limit** *number* { *warn-threshold* | **simply-alert** }	可选
应用入方向路由策略	**import route-policy** *route-policy*	可选 缺省情况下，接收所有VPN Target属性匹配的路由
应用出方向路由策略	**export route-policy** *route-policy*	可选 缺省情况下，不对发布的路由进行过滤

说明

- VPN 实例视图下配置的路由相关属性既可以用于 IPv4 VPN，也可以用于 IPv6 VPN。
- 既可以在 VPN 实例视图下，也可以在 IPv6 VPN 视图下，配置 IPv6 VPN 的路由相关属性。如果同时在两个视图下配置了路由相关属性，则 IPv6 VPN 采用 IPv6 VPN 视图下配置的路由相关属性。
- 命令 **vpn-target** 最多可以配置 8 个 VPN Target；一个 VPN 实例最多可以配置 64 个 VPN Target。
- 可以配置一个 VPN 实例支持的最大路由数，以防止 PE 路由器的入接口有过多的路由。一个 PE 路由器最多可以支持的路由条数与具体产品有关。
- 为 VPN 实例配置路由策略之前必须已经创建了路由策略，否则无法对接收和发布的路由进行过滤。

2.3 配置IPv6 MCE的路由交换

IPv6 MCE 可以看作一种通过路由隔离实现业务隔离的组网方案。配置 IPv6 MCE 的关键为：

- 配置 IPv6 MCE 与站点之间的路由交换
- 配置 IPv6 MCE 与 PE 之间的路由交换

在 IPv6 MCE 组网方案中，路由计算时需要关闭 PE 上的路由环路检测功能，防止路由丢失；同时禁止各路由协议互操作功能，以节省系统资源。

2.3.1 配置准备

在配置 IPv6 MCE 之前，需完成以下任务：

- 创建 VPN 实例、在 IPv6 MCE 上将连接站点和 PE 的接口与 VPN 实例绑定
- 配置相关接口的链路层协议和网络层协议，保证路由可达

2.3.2 配置IPv6 MCE与站点之间的路由交换

1. 配置IPv6 MCE与站点之间使用IPv6 静态路由

IPv6 MCE 可以通过 IPv6 静态路由与 Site 连接。传统 CE 配置的 IPv6 静态路由对全局生效，无法解决多 VPN 间的地址重叠问题。以太网交换机提供的 IPv6 MCE 功能可以将 IPv6 静态路由与 VPN 实例相绑定，将各 IPv6 VPN 之间的 IPv6 静态路由进行隔离。

表2-4 配置 IPv6 MCE 与站点之间使用 IPv6 静态路由

操作	命令	说明
进入系统视图	**system-view**	-
为指定VPN实例配置IPv6静态路由	**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*]	二者必选其一 该配置在IPv6 MCE上进行，站点上的配

操作	命令	说明
	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*]	置方法与普通IPv6静态路由相同

2. 配置IPv6 MCE与站点之间使用RIPng

一个 RIPng 进程只能属于一个 IPv6 VPN 实例。如果在启动 RIPng 进程时不绑定到 IPv6 VPN 实例，则该进程属于公网进程。通过在 IPv6 MCE 上将 RIPng 进程与 IPv6 VPN 实例绑定，可以使不同 IPv6 VPN 内的私网路由通过不同的 RIPng 进程在 Site 和 IPv6 MCE 间进行交互，保证了私网路由的隔离和安全。

表2-5 配置 IPv6 MCE 与站点之间使用 RIPng

操作	命令	说明
进入系统视图	**system-view**	-
创建IPv6 MCE与站点间的RIPng实例，并进入RIPng视图	**ripng** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在IPv6 MCE上进行，站点上配置普通RIPng即可
引入由PE发布的远端站点的路由	**import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | **route-policy** *route-policy-name*] *	必选 缺省情况下，RIPng未引入其它路由
配置引入路由的缺省度量值	**default cost** *value*	可选 缺省情况下，引入路由的缺省度量值为0
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
在指定的网络接口上使能RIPng	**ripng** *process-id* **enable**	必选 缺省情况下，接口禁用RIPng

说明

有关 RIPng 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“RIPng”。

3. 配置IPv6 MCE与站点之间使用OSPFv3

一个 OSPFv3 进程只能属于一个 IPv6 VPN 实例。如果在启动 OSPFv3 进程时不绑定到 IPv6 VPN 实例，则该进程属于公网进程。

通过在 IPv6 MCE 上将 OSPFv3 进程与 IPv6 VPN 实例绑定，可以使不同 IPv6 VPN 内的私网路由通过不同的 OSPFv3 进程在 Site 和 IPv6 MCE 间进行交互，保证了私网路由的隔离和安全。

表2-6 配置 IPv6 MCE 与站点之间使用 OSPFv3

操作	命令	说明
进入系统视图	**system-view**	-

操作	命令	说明
创建IPv6 MCE与站点间的OSPFv3实例，并进入OSPFv3视图	**ospfv3** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在IPv6 MCE上进行，站点上配置普通OSPFv3即可
配置Router ID	**router-id** *router-id*	必选
引入由PE发布的远端站点的路由	**import-route** *protocol* [*process-id* | **allow-ibgp**] [**cost** *value* | **route-policy** *route-policy-name* | **type** *type*] *	必选 缺省情况下，没有引入其他协议的路由信息
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
在接口上使能OSPFv3	**ospfv3** *process-id* **area** *area-id* [**instance** *instance-id*]	必选 缺省情况下，接口上没有使能OSPFv3

说明

- 删除 VPN 实例后，相关的所有 OSPFv3 进程也将全部被删除。
- 有关 OSPFv3 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“OSPFv3”。

4. 配置IPv6 MCE与站点之间使用IPv6 IS-IS

一个 IPv6 IS-IS 进程只能属于一个 IPv6 VPN 实例。如果在启动 IPv6 IS-IS 进程时不绑定到 IPv6 VPN 实例，则该进程属于公网进程。

通过在 IPv6 MCE 上将 IPv6 IS-IS 进程与 IPv6 VPN 实例绑定，可以使不同 IPv6 VPN 内的私网路由通过不同的 IPv6 IS-IS 进程在 Site 和 IPv6 MCE 间进行交互，保证了私网路由的隔离和安全。

表2-7 配置 IPv6 MCE 与站点之间使用 IPv6 IS-IS

操作	命令	说明
进入系统视图	**system-view**	-
创建IPv6 MCE与站点间的IPv6 IS-IS实例，并进入IS-IS视图	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在IPv6 MCE上进行，站点上配置普通IPv6 IS-IS即可
配置网络实体名称	**network-entity** *net*	必选 缺省情况下，没有配置网络实体名称
使能IS-IS进程的IPv6能力	**ipv6 enable**	必选 缺省情况下，没有使能IS-IS路由进程的IPv6能力

操作	命令	说明
引入由PE发布的远端站点的路由	**ipv6 import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | [**level-1** | **level-1-2** | **level-2**] | **route-policy** *route-policy-name* | **tag** *tag*] *	可选 缺省情况下，IPv6 IS-IS不引入其它协议的路由信息 如果**ipv6 import-route**命令中不指定引入的级别，则默认为引入路由到Level-2路由表中
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
使能接口IS-IS路由进程的IPv6能力，并指定要关联的IS-IS进程号	**isis ipv6 enable** [*process-id*]	必选 缺省情况下，接口上没有使能IS-IS路由进程的IPv6能力

说明

有关 IPv6 IS-IS 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“IPv6 IS-IS”。

5. 配置IPv6 MCE与站点之间使用EBGP

IPv6 MCE 与站点间使用 EBGP 交换路由信息时，需要在 IPv6 MCE 上为每个 IPv6 VPN 实例配置 IPv6 BGP 对等体，并在站点上引入相应 IPv6 VPN 内的 IGP 路由信息。

IPv6 MCE 使用 EBGP 交换路由信息时，还可以通过 Filter-policy 对接收/发布的路由进行过滤。

(1) IPv6 MCE 上的配置

表2-8 配置 IPv6 MCE 与站点之间使用 EBGP

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入IPv6 BGP-VPN实例视图	**ipv6-family vpn-instance** *vpn-instance-name*	必选
配置IPv6 BGP对等体的AS号	**peer** *ipv6-address* **as-number** *as-number*	必选
引入由PE发布的远端站点的路由	**import-route** *protocol* [*process-id* [**med** *med-value* | **route-policy** *route-policy-name*] *]	必选 缺省情况下，BGP不引入且不通告其它协议的路由
配置对发布的路由信息进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ip-prefix-name* } **export** [**direct** | **isisv6** *process-id* | **ripng** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤

说明

配置 IPv6 BGP VPN 实例后，IPv6 VPN 实例中的 IPv6 BGP 路由交换与普通的 IPv6 BGP 相同。有关 IPv6 BGP 协议的配置，请参见“三层技术-IP 路由配置指导”中的“IPv6 BGP”。

(2) 站点上的配置

表2-9 站点上的配置

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入IPv6地址族视图	**ipv6-family**	-
将IPv6 MCE配置为EBGP对等体	**peer** *ipv6-address* **as-number** *as-number*	必选
引入VPN内的IGP路由	**import-route** *protocol* [*process-id* [**med** *med-value* | **route-policy** *route-policy-name*] *]	必选 缺省情况下，BGP不引入且不通告其它协议的路由 站点需要将自己所能到达的IPv6 VPN网段地址发布给接入的IPv6 MCE

2.3.3 配置IPv6 MCE与PE之间的路由交换

由于在 IPv6 MCE 设备上已经将站点内的私网路由信息与 IPv6 VPN 实例进行了绑定，因此，只需要在 IPv6 MCE 与 PE 之间将接口与 IPv6 VPN 实例进行绑定、进行简单的路由配置、并将 IPv6 MCE 上维护的站点内的 IPv6 VPN 路由引入到 IPv6 MCE-PE 间的路由协议中，便可以实现私网 VPN 路由信息的传播。

1. 配置IPv6 MCE与PE之间使用IPv6 静态路由

表2-10 配置 IPv6 MCE 与 PE 之间使用 IPv6 静态路由

操作	命令	说明
进入系统视图	**system-view**	-
为指定VPN实例配置IPv6静态路由	**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*]	二者必选其一
	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*]	

2. 配置IPv6 MCE与PE之间使用RIPng

表2-11 配置 IPv6 MCE 与 PE 之间使用 RIPng

操作	命令	说明
进入系统视图	**system-view**	-
创建IPv6 MCE与PE间的RIPng实例，并进入RIPng视图	**ripng** [*process-id*] **vpn-instance** *vpn-instance-name*	必选
引入站点内的VPN路由	**import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | **route-policy** *route-policy-name*] *	必选 缺省情况下，RIPng未引入其它路由
配置引入路由的缺省度量值	**default cost** *value*	可选 缺省情况下，引入路由的缺省度量值为0
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
在指定的网络接口上使能RIPng	**ripng** *process-id* **enable**	必选 缺省情况下，接口禁用RIPng

说明

有关 RIPng 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“RIPng”。

3. 配置IPv6 MCE与PE之间使用OSPFv3

表2-12 配置 IPv6 MCE 与 PE 之间使用 OSPFv3

操作	命令	说明
进入系统视图	**system-view**	-
创建IPv6 MCE与PE间的OSPFv3实例，并进入OSPFv3视图	**ospfv3** [*process-id*] **vpn-instance** *vpn-instance-name*	必选
配置Router ID	**router-id** *router-id*	必选
引入站点内的VPN路由	**import-route** *protocol* [*process-id* | **allow-ibgp**] [**cost** *value* | **route-policy** *route-policy-name* | **type** *type*] *	必选 缺省情况下，没有引入其他协议的路由信息
配置对发布的路由进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **export** [**bgp4+** | **direct** | **isisv6** *process-id* | **ospfv3** *process-id* | **ripng** *process-id* | **static**]	可选 缺省情况下，没有对发布的路由信息进行过滤
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
在接口上使能OSPFv3	**ospfv3** *process-id* **area** *area-id* [**instance** *instance-id*]	必选 缺省情况下，接口上没有使能OSPFv3

说明

有关 OSPFv3 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“OSPFv3”。

4. 配置IPv6 MCE与PE之间使用IPv6 IS-IS

表2-13 配置 IPv6 MCE 与 PE 之间使用 IPv6 IS-IS

操作	命令	说明
进入系统视图	**system-view**	-
创建 IPv6 MCE 与 PE 间的 IPv6 IS-IS实例，并进入IS-IS视图	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	必选
配置网络实体名称	**network-entity** *net*	必选 缺省情况下，没有配置网络实体名称
使能IS-IS进程的IPv6能力	**ipv6 enable**	必选 缺省情况下，没有使能IS-IS路由进程的IPv6能力
引入站点内的VPN路由	**ipv6 import-route** *protocol* [*process-id*] [**allow-ibgp**] [**cost** *cost* | [**level-1** | **level-1-2** | **level-2**] | **route-policy** *route-policy-name* | **tag** *tag*] *	可选 缺省情况下，IPv6 IS-IS不引入其它协议的路由信息 如果**ipv6 import-route**命令中不指定引入的级别，则默认为引入路由到Level-2路由表中
对发布的路由进行过滤	**ipv6 filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* | **route-policy** *route-policy-name* } **export** [*protocol* [*process-id*]]	可选 缺省情况下，IPv6 IS-IS不对发布的路由信息进行过滤
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
使能接口IS-IS路由进程的IPv6能力，并指定要关联的IS-IS进程号	**isis ipv6 enable** [*process-id*]	必选 缺省情况下，接口上没有使能IS-IS路由进程的IPv6能力

说明

有关 IPv6 IS-IS 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“IPv6 IS-IS”。

5. 配置IPv6 MCE与PE之间使用EBGP

表2-14 配置 IPv6 MCE 与 PE 之间使用 EBGP

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入IPv6 BGP-VPN实例视图	**ipv6-family vpn-instance** *vpn-instance-name*	必选
将PE配置为EBGP对等体	**peer** *ipv6-address* **as-number** *as-number*	必选
引入站点内的VPN路由	**import-route** *protocol* [*process-id* [**med** *med-value* | **route-policy** *route-policy-name*] *]	必选 缺省情况下，BGP不引入且不通告其它协议的路由
配置对发布的路由信息进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ip-prefix-name* } **export** [**direct** | **isisv6** *process-id* | **ripng** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤

说明

配置 IPv6 BGP VPN 实例后，IPv6 VPN 实例中的 IPv6 BGP 路由交换与普通的 IPv6 BGP 相同。有关 IPv6 BGP 协议的配置，请参见“三层技术-IP 路由配置指导”中的“IPv6 BGP”。

2.4 IPv6 MCE显示和维护

2.4.1 复位BGP连接

当 BGP 配置变化后，可以通过软复位或复位 BGP 连接使新的配置生效。软复位需要 BGP 对等体具备路由刷新能力（支持 ROUTE-REFRESH 消息）。

请在用户视图下进行下列操作。

表2-15 复位 BGP 连接

操作	命令
软复位指定VPN实例的IPv6 BGP连接	**refresh bgp ipv6 vpn-instance** *vpn-instance-name* { *ipv6-address* | **all** | **external** } { **export** | **import** }
复位指定VPN实例的IPv6 BGP连接	**reset bgp ipv6 vpn-instance** *vpn-instance-name* { *as-number* | *ipv6-address* | **all** | **external** }

2.4.2 显示IPv6 MCE的运行状态

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 MCE 的运行情况，通过查看显示信息验证配置的效果。

表2-16 显示 IPv6 MCE 的运行状态

操作	命令
显示指定VPN实例信息	**display ip vpn-instance** [**instance-name** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例的IPv6 FIB信息	**display ipv6 fib vpn-instance** *vpn-instance-name* [**acl6** *acl6-number* | **ipv6-prefix** *ipv6-prefix-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例中与指定目的IPv6地址匹配的IPv6 FIB信息	**display ipv6 fib vpn-instance** *vpn-instance-name ipv6-address* [*prefix-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例中PE与CE之间建立的IPv6 BGP对等体的信息	**display bgp vpnv6 vpn-instance** *vpn-instance-name* **peer** [*ipv6-address* **verbose** | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例的BGP VPNv6路由信息	**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*]

说明

有关查看路由表的命令介绍请参见“三层技术-IP 路由命令参考”中的“IP 路由基础”。

2.5 IPv6 MCE典型配置举例

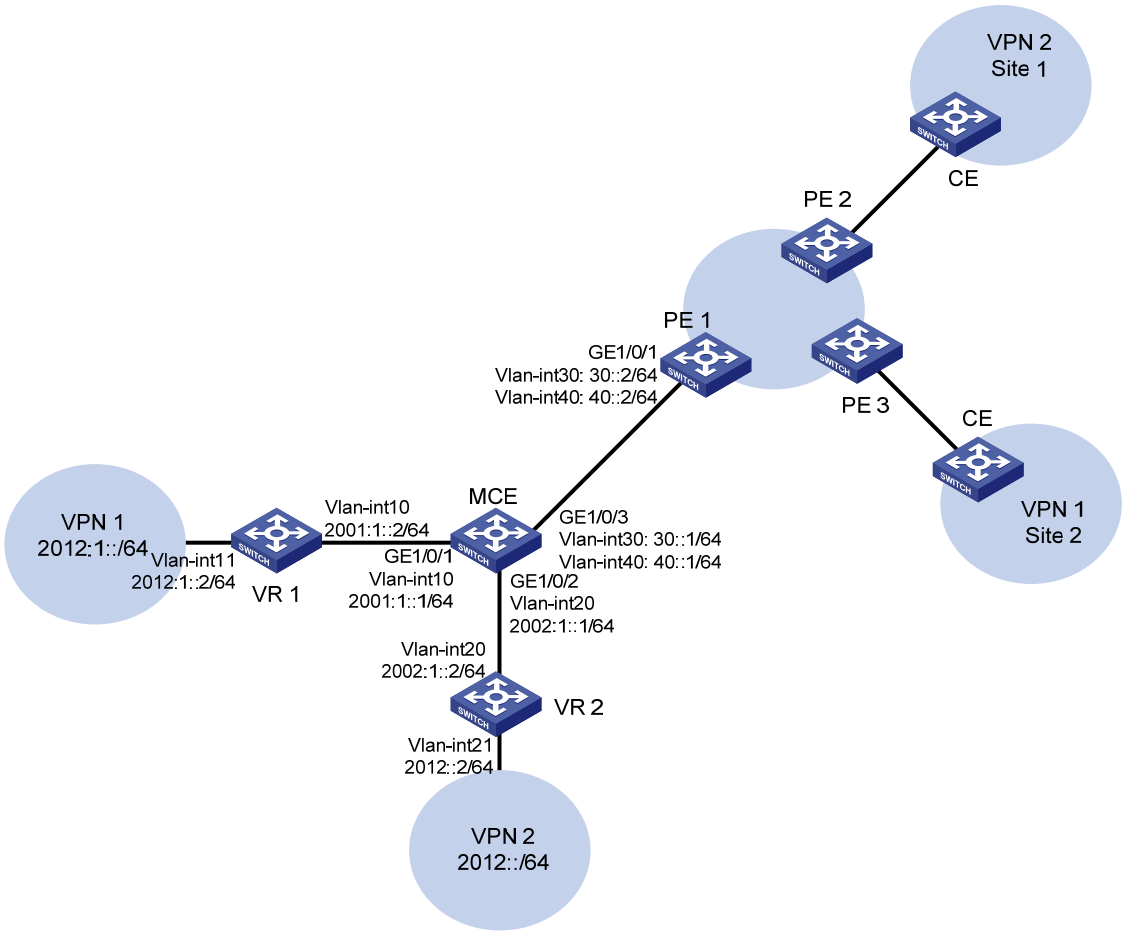
2.5.1 配置IPv6 MCE典型配置举例

1. 组网需求

- IPv6 MCE 设备通过 Vlan-interface10 接口（IPv6 地址为 2001:1::1/64）连接到 VPN 1，地址范围为 2012:1::/64；通过 Vlan-interface20 接口（IPv6 地址为 2002:1::1/64）连接到 VPN 2，VPN 2 内运行 RIPng 路由协议，地址范围为 2012::/64。
- 要求 IPv6 MCE 设备能够将 VPN 之间的路由隔离，并通过 OSPFv3 将各 VPN 的路由发布到 PE1 设备。

2. 组网图

图2-1 配置 IPv6 MCE 典型配置举例一组网示意图



3. 配置步骤

为区分设备，假设 IPv6 MCE 系统名为“MCE”，VPN 1 和 VPN 2 的边缘设备分别名为“VR1”和“VR2”，PE 1 系统名为“PE1”。

(1) 在 MCE 和 PE 1 上配置 VPN 实例

在 MCE 设备上配置 VPN 实例，名称分别为 VPN1 和 VPN2，RD 分别取值为 10:1 和 20:1，VPN Target 取值与 RD 取相同数值，Export 和 Import 均取此值。

```
<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
```

创建 VLAN10，将端口 GigabitEthernet1/0/1 加入 VLAN 10，并创建 Vlan-interface10 接口。

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

配置 Vlan-interface10 接口与 VPN 实例 vpn1 绑定，并配置接口的 IPv6 地址。

```
[MCE] interface vlan-interface 10
[MCE-Vlan-interface10] ip binding vpn-instance vpn1
[MCE-Vlan-interface10] ipv6 address 2001:1::1 64
[MCE-Vlan-interface10] quit
```

使用类似步骤配置 VLAN20，将端口 GigabitEthernet1/0/2 加入 VLAN 20，配置接口与 VPN 实例 vpn2 绑定，并配置接口的 IPv6 地址。

```
[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
```

在 PE 1 上配置 VPN 实例，名称分别为 VPN1 和 VPN2，RD 分别取值为 30:1 和 40:1，VPN Target 取值分别为 10:1 和 20:1，Export 和 Import 均取此值。

```
<PE1> system-view
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 30:1
[PE1-vpn-instance-vpn1] vpn-target 10:1
[PE1-vpn-instance-vpn1] quit
[PE1] ip vpn-instance vpn2
[PE1-vpn-instance-vpn2] route-distinguisher 40:1
[PE1-vpn-instance-vpn2] vpn-target 20:1
[PE1-vpn-instance-vpn2] quit
```

(2) MCE 与 Site 间路由配置

MCE 与 VPN 1 直接相连，且 VPN 1 内未使用路由协议，因此可以使用 IPv6 静态路由进行配置。

配置 VR1 与 MCE 连接的接口地址为 2001:1::2/64，连接 VPN 1 接口的地址为 2012:1::2/64。向 VLAN 中增加端口和接口 IP 地址的配置过程省略。

在 VR1 上配置缺省路由，指定出方向报文的下一跳地址为 2001:1::1。

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

在 MCE 上指定 IPv6 静态路由，去往 2012:1::/64 网段的报文，下一跳地址为 2001:1::2，并将此路由与 VPN 1 实例绑定。

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

VPN 2 内运行 RIPng，在 MCE 上配置 RIPng 进程 20，并与 VPN 实例 vpn2 绑定，以便将 VPN 2 内的路由学习到 VPN 实例 vpn2 的路由表中。

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

配置 RIPng 发布 2002:1::/64 网段路由。

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

在 VR 2 上，配置与 MCE 连接的接口地址为 2002:1::2/64，连接 VPN 2 接口的地址为 2012::2/64（配置过程略）。

在 VR 2 上配置 RIPng 发布 2012::/64 和 2002:1::/64 网段路由。

```
<VR2> system-view
[VR2] ripng 20
```

```
[VR2-ripng-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
```

在 MCE 上查看 VPN 实例 vpn1 和 vpn2 的路由信息。

```
[MCE] display ipv6 routing-table vpn-instance vpn1
 Routing Table : vpn1
         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 : vpn2
         Destinations : 5        Routes : 5

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  : Vlan20                                     Cost      : 0

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

Destination: 2012::/64                                  Protocol  : RIPng
NextHop    : FE80::20F:E2FF:FE3E:9CA2                   Preference: 100
Interface  : Vlan20                                     Cost      : 1
```

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

可以看到，MCE 已经通过 RIPng 学习到了 VPN 2 内的私网路由，并与 VPN 1 内的路由信息分别维护在两个路由表内，有效进行了隔离。

(3) MCE 与 PE 间路由配置

MCE 使用 GigabitEthernet1/0/3 端口连接到 PE 1 的 GigabitEthernet1/0/1 端口，需要配置这两个端口为 Trunk 端口，并允许 VLAN 30 和 VLAN 40 的报文携带 Tag 通过。

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

配置 PE 1 的 GigabitEthernet1/0/1 端口。

```
[PE1] interface gigabitethernet 1/0/1
[PE1-GigabitEthernet1/0/1] port link-type trunk
[PE1-GigabitEthernet1/0/1] port trunk permit vlan 30 40
[PE1-GigabitEthernet1/0/1] quit
```

在 MCE 上创建 VLAN 30 和接口 Vlan-interface30，配置接口与 VPN 实例 vpn1 绑定，并配置接口的 IPv6 地址。

```
[MCE] vlan 30
[MCE-vlan30] quit
[MCE] interface vlan-interface 30
[MCE-Vlan-interface30] ip binding vpn-instance vpn1
[MCE-Vlan-interface30] ipv6 address 30::1 64
[MCE-Vlan-interface30] quit
```

在 MCE 上创建 VLAN 40 和接口 Vlan-interface40，配置接口与 VPN 实例 vpn2 绑定，并配置接口的 IPv6 地址。

```
[MCE] vlan 40
[MCE-vlan40] quit
[MCE] interface vlan-interface 40
[MCE-Vlan-interface40] ip binding vpn-instance vpn2
[MCE-Vlan-interface40] ipv6 address 40::1 64
[MCE-Vlan-interface40] quit
```

在 PE 1 上创建 VLAN 30 和接口 Vlan-interface30，配置接口与 VPN 实例 vpn1 绑定，并配置接口的 IPv6 地址。

```
[PE1] vlan 30
[PE1-vlan30] quit
[PE1] interface vlan-interface 30
[PE1-Vlan-interface30] ip binding vpn-instance vpn1
[PE1-Vlan-interface30] ipv6 address 30::2 64
[PE1-Vlan-interface30] quit
```

在 PE 1 上创建 VLAN 40 和接口 Vlan-interface40，配置接口与 VPN 实例 vpn2 绑定，并配置接口的 IPv6 地址。

```
[PE1] vlan 40
[PE1-vlan40] quit
[PE1] interface vlan-interface 40
```

```
[PE1-Vlan-interface40] ip binding vpn-instance vpn2
[PE1-Vlan-interface40] ipv6 address 40::2 64
[PE1-Vlan-interface40] quit
```

配置 MCE 和 PE 1 的 Loopback0 接口，用于指定 MCE 和 PE 1 的 Router ID，地址分别为 101.101.10.1 和 100.100.10.1。配置步骤这里省略。

配置 MCE 启动 OSPFv3 进程 10，绑定到 VPN 实例 vpn1，并引入 VPN 1 的 IPv6 静态路由。

```
[MCE] ospfv3 10 vpn-instance vpn1
[MCE-ospf-10] router-id 101.101.10.1
[MCE-ospf-10] import-route static
[MCE-ospf-10] quit
```

在 Vlan-interface30 中使能 OSPFv3。

```
[MCE] interface vlan-interface 30
[MCE-Vlan-interface30] ospfv3 10 area 0.0.0.0
[MCE-Vlan-interface30] quit
```

配置 PE 1 启动 OSPFv3 进程 10，绑定到 VPN 实例 vpn1。

```
[PE1] ospfv3 10 vpn-instance vpn1
[PE1-ospf-10] router-id 100.100.10.1
[PE1-ospf-10] quit
```

在 Vlan-interface30 中使能 OSPFv3。

```
[PE1] interface vlan-interface 30
[PE1-Vlan-interface30] ospfv3 10 area 0.0.0.0
[PE1-Vlan-interface30] quit
```

显示 PE 1 上的 VPN 1 路由信息。

```
[PE1] display ipv6 routing-table vpn-instance vpn1
Routing Table : vpn1
         Destinations : 5        Routes : 5

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

Destination: 30::/64                                    Protocol  : Direct
NextHop    : 30::2                                      Preference: 0
Interface  : Vlan30                                     Cost      : 0

Destination: 30::2/128                                  Protocol  : Direct
NextHop    : ::1                                        Preference: 0
Interface  : InLoop0                                    Cost      : 0

Destination: 2012:1::/64                                Protocol  : OSPFv3
NextHop    : FE80::202:FF:FE02:2                        Preference: 150
Interface  : Vlan30                                     Cost      : 1

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

可以看到，PE 1 通过 OSPFv3 学习到了 VPN 1 内的私网路由。

MCE 与 PE 1 间配置 OSPFv3 进程 20，引入 VPN 实例 vpn2 的路由信息的过程与上面介绍的配置基本一致，不同的是在 MCE 的 OSPFv3 中配置引入的是 RIPng 进程 10 的路由，这里不再赘述。

通过下面的显示信息可以看出，PE 1 通过 OSPFv3 学习到了 VPN 2 内的私网路由。

```
[PE1] display ipv6 routing-table vpn-instance vpn2
Routing Table : vpn2
         Destinations : 5        Routes : 5

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

Destination: 40::/64                                    Protocol  : Direct
NextHop    : 40::2                                      Preference: 0
Interface  : Vlan40                                     Cost      : 0

Destination: 40::2/128                                  Protocol  : Direct
NextHop    : ::1                                        Preference: 0
Interface  : InLoop0                                    Cost      : 0

Destination: 2012::/64                                  Protocol  : OSPFv3
NextHop    : FE80::200:FF:FE0F:5                        Preference: 150
Interface  : Vlan40                                     Cost      : 1

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

至此，通过配置，已经将两个 VPN 实例内的路由信息完整地传播到 PE 1 中，配置完成。

目 录

1 MPLS基本配置 ····· 1-1
1.1 MPLS简介 ····· 1-1
1.1.1 MPLS基本概念 ····· 1-1
1.1.2 MPLS网络结构 ····· 1-3
1.1.3 LSP建立与标签的发布和管理 ····· 1-3
1.1.4 MPLS数据转发过程 ····· 1-6
1.1.5 LDP ····· 1-8
1.1.6 协议规范 ····· 1-10
1.2 MPLS配置任务简介 ····· 1-10
1.3 使能MPLS功能 ····· 1-11
1.3.1 配置准备 ····· 1-11
1.3.2 使能MPLS功能 ····· 1-11
1.4 建立静态LSP ····· 1-12
1.4.1 配置准备 ····· 1-12
1.4.2 建立静态LSP ····· 1-12
1.5 利用LDP建立动态LSP ····· 1-13
1.5.1 配置MPLS LDP能力 ····· 1-13
1.5.2 配置LDP本地会话的参数 ····· 1-13
1.5.3 配置LDP远端会话的参数 ····· 1-14
1.5.4 配置PHP特性 ····· 1-16
1.5.5 配置LSP触发策略 ····· 1-16
1.5.6 配置LDP标签分配控制方式 ····· 1-17
1.5.7 配置LDP环路检测 ····· 1-17
1.5.8 配置LDP MD5 认证 ····· 1-18
1.5.9 配置LDP标签过滤 ····· 1-19
1.5.10 配置发送的LDP报文的DSCP优先级 ····· 1-20
1.6 维护LDP会话 ····· 1-21
1.6.1 配置MPLS LDP与BFD联动 ····· 1-21
1.6.2 重启LDP会话 ····· 1-21
1.7 管理及优化MPLS转发功能 ····· 1-22
1.7.1 配置Ingress节点对TTL的处理方式 ····· 1-22
1.7.2 配置MPLS的TTL超时消息发送功能 ····· 1-23
1.7.3 配置LDP GR ····· 1-24

1.8 配置MPLS统计功能 ····· 1-26
1.8.1 配置LSP统计数据输出时间间隔 ····· 1-26
1.9 检测LSP ····· 1-27
1.9.1 MPLS LSP Ping功能 ····· 1-27
1.9.2 MPLS LSP Tracert功能 ····· 1-28
1.9.3 配置BFD检测LSP功能 ····· 1-28
1.9.4 配置周期性LSP Tracert功能 ····· 1-29
1.10 MPLS的Trap功能 ····· 1-30
1.11 MPLS显示和维护 ····· 1-31
1.11.1 显示MPLS运行状态 ····· 1-31
1.11.2 显示MPLS LDP运行状态 ····· 1-31
1.11.3 清除MPLS相关信息 ····· 1-32
1.12 MPLS配置举例 ····· 1-33
1.12.1 配置静态建立LSP ····· 1-33
1.12.2 配置利用LDP动态建立LSP ····· 1-35
1.12.3 配置利用BFD检测LSP ····· 1-40

1 MPLS基本配置

说明

本系列交换机中 S5500-28SC-HI 和 S5500-52SC-HI 不支持 MPLS 功能。

1.1 MPLS简介

MPLS（Multiprotocol Label Switching，多协议标签交换）是一种新兴的 IP 骨干网技术。MPLS 在无连接的 IP 网络上引入面向连接的标签交换概念，将第三层路由技术和第二层交换技术相结合，充分发挥了 IP 路由的灵活性和二层交换的简捷性。

MPLS 广泛应用于大规模网络中，它具有以下优点：

- 在 MPLS 网络中，设备根据短而定长的标签转发报文，省去了通过软件查找 IP 路由的繁琐过程，为数据在骨干网络中的传输提供了一种高速高效的方式。
- MPLS 位于链路层和网络层之间，它可以建立在各种链路层协议（如 PPP、ATM、帧中继、以太网等）之上，为各种网络层（IPv4、IPv6）提供面向连接的服务，兼容现有各种主流网络技术。
- 支持多层标签和面向连接的特点，使得 MPLS 在 VPN、流量工程、QoS 等方面得到广泛应用。
- 具有良好的扩展性，在 MPLS 网络基础上可以为客户提供各种服务。

1.1.1 MPLS基本概念

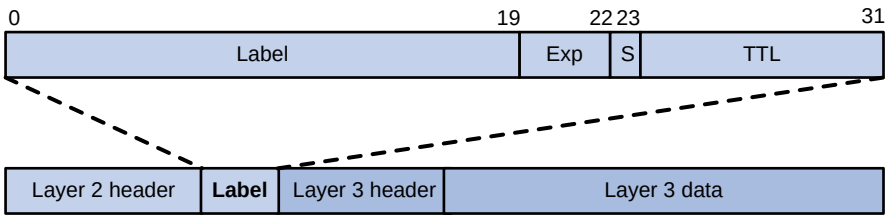
1. 转发等价类

FEC（Forwarding Equivalence Class，转发等价类）是 MPLS 中的一个重要概念。MPLS 是一种分类转发技术，它将具有相同特征（目的地相同或具有相同服务等级等）的报文归为一类，称为 FEC。属于相同 FEC 的报文在 MPLS 网络中将获得完全相同的处理。目前设备只支持根据报文的网络层目的地址划分 FEC。

2. 标签

标签是一个长度固定、只具有本地意义的标识符，用于唯一标识一个报文所属的 FEC。一个标签只能代表一个 FEC。

图1-1 标签的封装结构



如 图 1-1 所示，标签封装在链路层报头和网络层报头之间的一个垫层中。标签长度为 4 个字节，由以下四个字段组成：

- Label：标签值，长度为 20bits，用来标识一个 FEC。
- Exp：3bits，保留，协议中没有明确规定，通常用作服务等级。
- S：1bit，MPLS 支持多重标签。值为 1 时表示为最底层标签。
- TTL：8bits，和 IP 报文中的 TTL 意义相同，可以用来防止环路。

说明

Exp 称为 MPLS 报文服务等级，可影响报文的优先调度。有关报文调度的详细信息，请参见“ACL 和 QoS 配置指导”中的“QoS”。

---

3. 标签交换路由器

LSR（Label Switching Router，标签交换路由器）是具有标签分发能力和标签交换能力的设备，是 MPLS 网络中的基本元素。

4. 标签边缘路由器

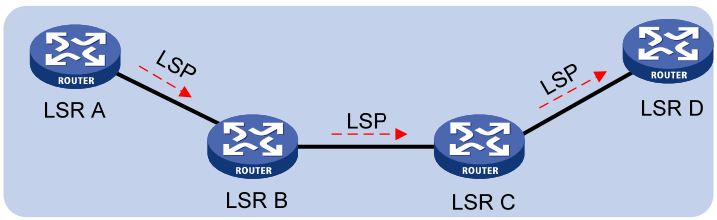
位于 MPLS 网络边缘、连接其他网络的 LSR 称为 LER（Label Edge Router，标签边缘路由器）。

5. 标签交换路径

属于同一个 FEC 的报文在 MPLS 网络中经过的路径称为 LSP（Label Switched Path，标签交换路径）。

LSP是从MPLS网络的入口到出口的一条单向路径。在一条LSP上，沿数据传送的方向，相邻的LSR分别称为上游LSR和下游LSR。如 图 1-2 所示，LSR B为LSR A的下游LSR，相应的，LSR A为LSR B的上游LSR。

图1-2 标签交换路径



6. 标签转发表

与 IP 网络中的 FIB（Forwarding Information Base，转发信息表）类似，在 MPLS 网络中，报文通过查找标签转发表确定转发路径。

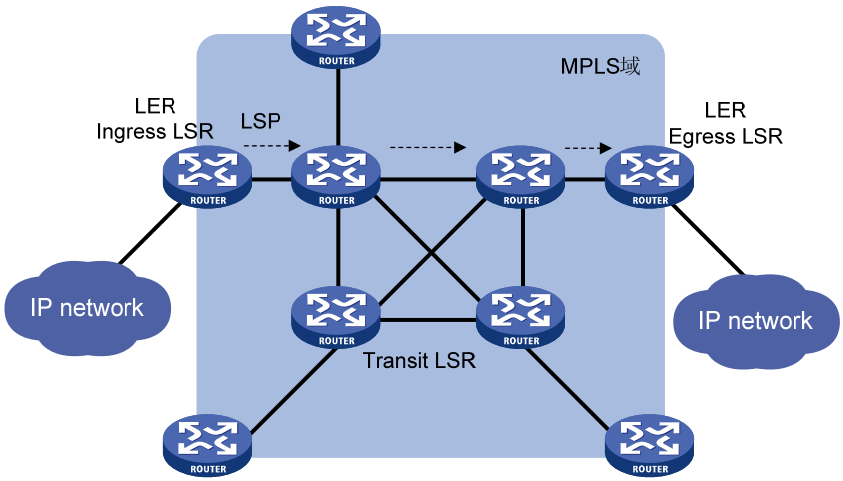
7. 控制平面和转发平面

MPLS 节点由两部分组成：

- 控制平面（Control Plane）：负责标签的分配、路由的选择、标签转发表的建立、标签交换路径的建立、拆除等工作。
- 转发平面（Forwarding Plane）：依据标签转发表对收到的分组进行转发。

1.1.2 MPLS网络结构

图1-3 MPLS 网络结构



如 图 1-3 所示，MPLS网络的基本构成单元是LSR，由LSR构成的网络称为MPLS域。MPLS网络包括以下几个组成部分：

- 入节点 Ingress：报文的入口 LER，负责为进入 MPLS 域的报文添加标签。
- 中间节点 Transit：MPLS 域内部的 LSR，根据标签沿着由一系列 LSR 构成的 LSP 将报文传送给出口 LER。
- 出节点 Egress：报文的出口 LER，负责剥离报文中的标签，并转发给目的网络。

Transit 根据报文上附加的标签进行 MPLS 转发，Ingress 和 Egress 负责 MPLS 与 IP 技术的转换。

1.1.3 LSP建立与标签的发布和管理

1. LSP建立

LSP 的建立过程实际就是将 FEC 和标签进行绑定，并将这种绑定通告相邻 LSR，以便在 LSR 上建立标签转发表的过程。LSP 既可以通过手工配置的方式静态建立，也可以利用标签分发协议动态建立。

(1) 手工配置的方式静态建立 LSP

建立静态 LSP 需要用户在报文转发路径中的各个 LSR 上手工配置为 FEC 分配的标签。建立静态 LSP 消耗的资源比较少，但静态建立的 LSP 不能根据网络拓扑变化动态调整。因此，静态 LSP 适用于拓扑结构简单并且稳定的小型网络。

(2) 利用标签发布协议动态建立 LSP

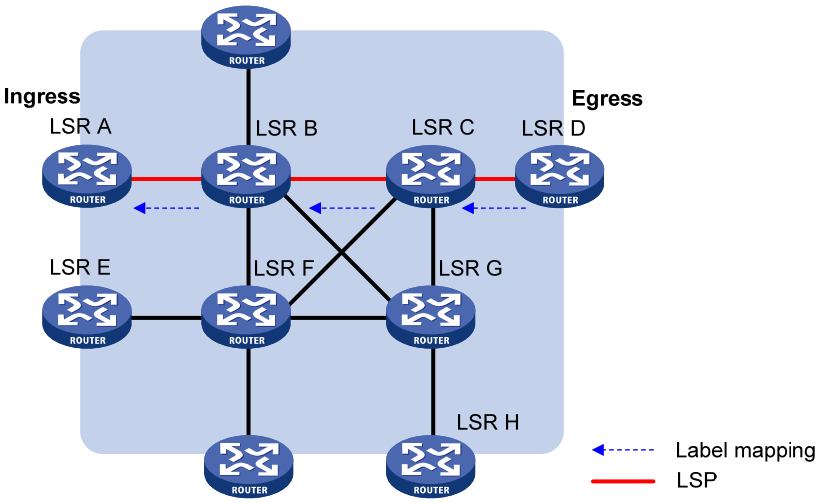
标签发布协议是 MPLS 的信令协议，负责划分 FEC、发布标签、建立维护 LSP 等。标签发布协议的种类较多，有专为标签发布而制定的协议，如 LDP（Label Distribution Protocol，标签分发协议），也有扩展后支持标签发布的协议，如 BGP、RSVP-TE。本文只介绍 LDP 协议。

说明

为了区分，本文中“标签发布协议”表示广义上所有用于标签发布的协议的总称；“LDP”表示RFC 5036规定的标签发布协议。LDP的详细介绍请参见“1.1.5 LDP”。

利用标签发布协议动态建立LSP的过程如图 1-4所示。下游LSR根据目的地址划分FEC，为特定FEC分配标签，并将标签和FEC的绑定关系通告给上游LSR；上游LSR根据该绑定关系建立标签转发表项。报文传输路径上的所有LSR都为该FEC建立对应的标签转发表项后，就成功地建立了用于转发属于该FEC报文的LSP。

图1-4 动态 LSP 建立过程



说明

如果在 LSR 上存在等价路由，则 MPLS 会根据等价路由建立等价 LSP，MPLS 报文在这些等价 LSP 之间进行负载分担。

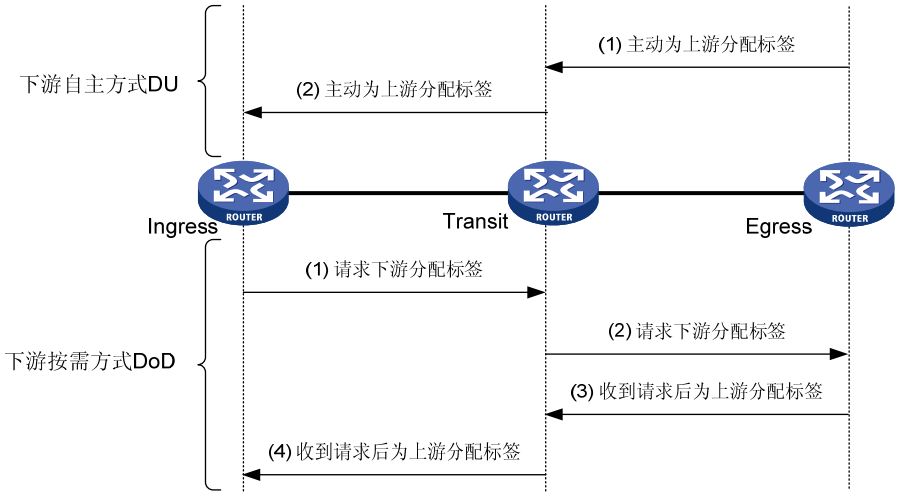
2. 标签的发布和管理

标签发布就是将为 FEC 分配的标签通告给其他 LSR。根据标签发布条件、标签发布顺序的不同，LSR 通告标签的方式分为 DU（Downstream Unsolicited，下游自主方式）和 DoD（Downstream On Demand，下游按需方式）、独立标签控制方式（Independent）和有序标签控制方式（Ordered）几种。

标签管理，即标签保持方式，是指 LSR 对收到的、但目前暂时用不到的 FEC 和标签绑定的处理方式，分为自由标签保持方式（Liberal）和保守标签保持方式（Conservative）两种。

(1) 标签发布方式（Label Advertisement Mode）

图1-5 标签发布方式



如 图 1-5 所示，标签发布方式分为：

- DU：对于一个特定的 FEC，下游 LSR 自动为该 FEC 分配标签，并主动将标签分发给上游 LSR。
- DoD：对于一个特定的 FEC，上游 LSR 请求下游 LSR 为该 FEC 分配标签，下游 LSR 收到请求后，为该 FEC 分配标签并向上游 LSR 通告该标签。

说明

- 目前，设备只支持 DU 标签发布方式。
- 具有标签分发邻接关系的上游 LSR 和下游 LSR 之间必须使用相同的标签发布方式，否则 LSP 无法正常建立。

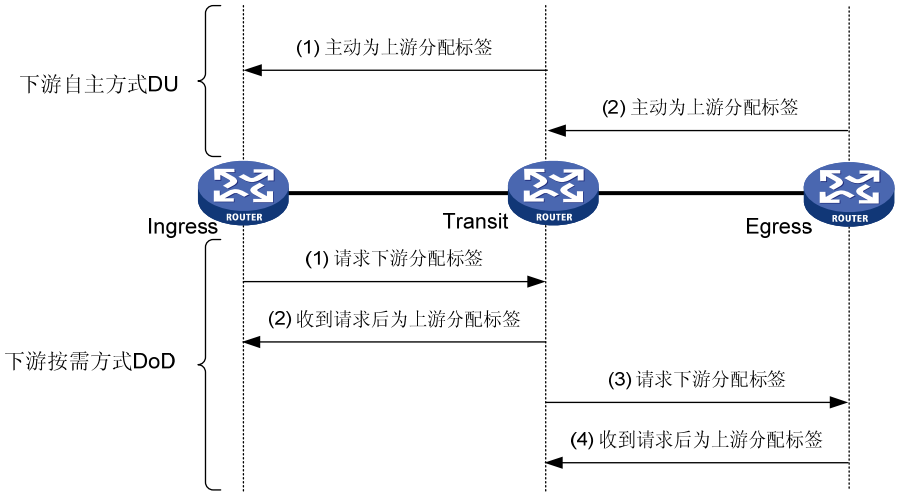
---

(2) 标签分配控制方式（Label Distribution Control Mode）

标签分配控制方式分为：

- 独立标签控制方式：LSR可以在任意时间向与它连接的LSR通告标签映射。使用这种方式时，LSR可能会在收到下游LSR的标签之前就向上游通告了标签。如 图 1-6 所示，如果标签发布方式是DU，则即使没有获得下游的标签，也会直接为上游分配标签；如果标签发布方式是DoD，则接收到标签请求的LSR直接为它的上游LSR分配标签，不必等待来自它的下游的标签。

图1-6 独立标签控制方式



- 有序标签控制方式：LSR只有收到它的下游LSR为某个FEC分配的标签，或该LSR是此FEC的出口节点时，才会向它的上游LSR通告此FEC的标签映射。图 1-5 中的标签发布过程采用了有序标签控制方式：如果标签发布模式为DU，则LSR只有收到下游LSR分配的标签后，才会向自己的上游LSR分配标签；如果标签发布模式为DoD，则下游LSR（Transit）收到上游LSR（Ingress）的标签请求后，继续向它的下游LSR（Egress）发送标签请求，Transit收到Egress分配的标签后，才会为Ingress分配标签。

(3) 标签保持方式（Label Retention Mode）

LSR 接收到标签映射后，保留标签的方式分为：

- 自由标签保持方式：对于从邻居 LSR 收到的标签映射，无论邻居 LSR 是不是指定 FEC 的下一跳都保留。这种方式的优点是 LSR 能够迅速适应网络拓扑变化，但是浪费标签，所有不能生成 LSP 的标签都需要保留。
- 保守标签保持方式：对于从邻居 LSR 收到的标签映射，只有当邻居 LSR 是指定 FEC 的下一跳时才保留。这种方式的优点是节省标签，但是对拓扑变化的响应较慢。

说明

目前只支持自由标签保持方式。

1.1.4 MPLS数据转发过程

1. 标签转发表构成

标签转发表由以下三部分构成：

- NHLFE（Next Hop Label Forwarding Entry，下一跳标签转发项）：描述对标签执行的操作，用于指导 MPLS 报文的转发。
- FTN（FEC to NHLFE map，FEC 到 NHLFE 表项的映射）：用于在 Ingress 节点将 FEC 映射到 NHLFE 表项。LSR 接收到不带标签的报文后，查找对应的 FIB 表项。如果 FIB 表项的

Token 值不是 Invalid，则该报文需要进行 MPLS 转发。LSR 根据 Token 值找到对应的 NHLFE 表项，以便确定需要执行的标签操作。

- ILM（Incoming Label Map，入标签映射）：用于将入标签映射到 NHLFE 表项。LSR 接收到带有标签的报文后，查找对应的 ILM 表项。如果 ILM 表项的 Token 值非空，则找到 Token 值对应的 NHLFE 表项，以便确定需要执行的标签操作。

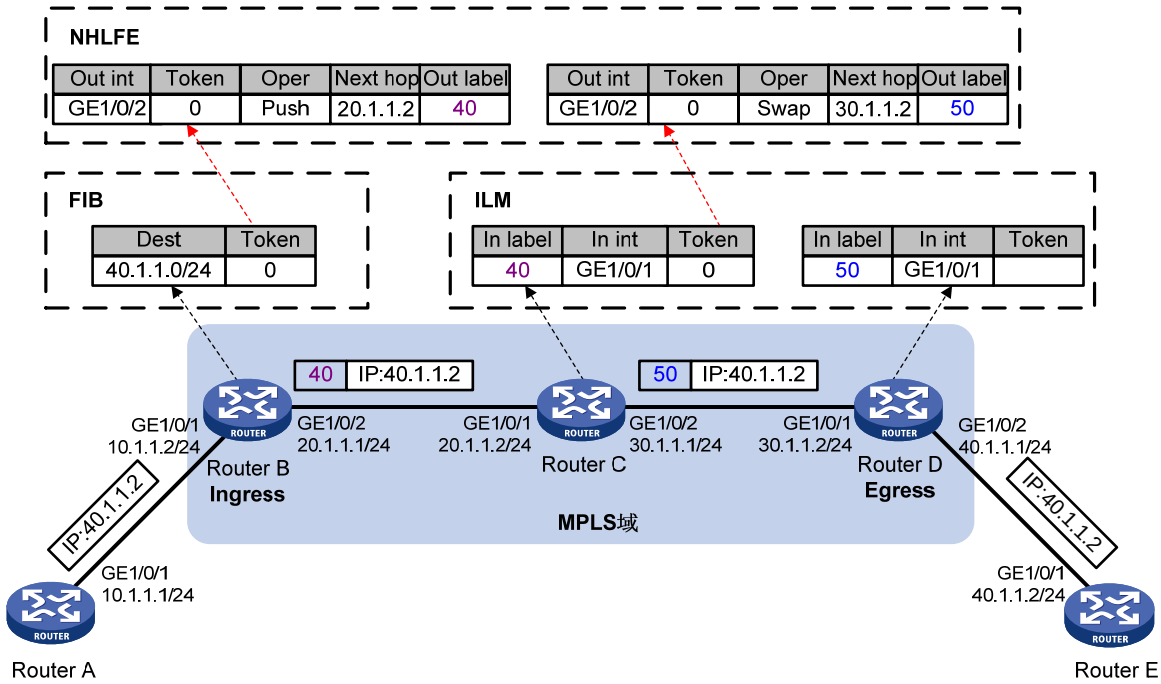
说明

FTN、ILM 通过 Token 与 NHLFE 关联。

---

2. MPLS数据转发

图1-7 MPLS 转发过程示意图



如 图 1-7 所示，MPLS网络中报文的转发过程为：

(1) Ingress（Router B）接收到不带标签的报文，根据目的地址判断该报文所属的 FEC，查找 FIB 表，获取 Token 值。Token 值不是 Invalid，则找到 Token 值对应的 NHLFE 表项。根据 NHLFE 表项为报文添加标签（40），并从相应的出接口（GigabitEthernet1/0/2）将带有标签的报文转发给下一跳 LSR（Router C）。

(2) Router C 接收到带有标签的报文，根据报文上的标签（40）查找 ILM 表项，获取 Token 值。Token 值非空，则找到 Token 值对应的 NHLFE 表项。根据 NHLFE 表项，用新的标签（50）替换原有标签，并从相应的出接口将带有标签的报文转发给下一跳 LSR（Router D）。

(3) Egress（Router D）接收到带有标签的报文，根据报文上的标签（50）查找 ILM 表项，获取 Token 值。Token 值为空，则删除报文中的标签。如果 ILM 表项中记录了出接口，则通过该出接口转发报文；否则，根据 IP 报头转发报文。

3. 倒数第二跳弹出

MPLS 网络中，Egress 节点接收到带有标签的报文后，查找标签转发表，弹出报文中的标签后，再进行下一层的标签转发或 IP 转发。Egress 节点转发报文之前要查找两次转发表：两次标签转发表，或一次标签转发表一次路由转发表。

为了减轻 Egress 节点的负担，提高 MPLS 网络对报文的处理能力，可以利用 PHP（Penultimate Hop Popping，倒数第二跳弹出）功能，在倒数第二跳节点处将标签弹出，Egress 节点只需查找一次转发表。

PHP 在 Egress 节点上配置。支持 PHP 的 Egress 节点分配给倒数第二跳节点的标签有以下两种：

- 标签值为 0 表示 IPv4 显示空标签（Explicit-null）。Egress 为 FEC 分配 IPv4 显式空标签，并通告给上游 LSR 后，上游 LSR 用这个值替代栈顶原来的标签，并将报文转发给 Egress。Egress 收到标签值为 0 的报文时，不会查找标签转发表，直接弹出标签栈，进行 IPv4 转发。
- 标签值 3 表示隐式空标签（Implicit-null），这个值不会出现在标签栈中。当一个 LSR 发现下游 LSR 通告的标签为隐式空标签时，它并不用这个值替代栈顶原来的标签，而是直接弹出标签，并将报文转发给下游 LSR（即 Egress）。Egress 接收到报文后，直接进行下一层的转发处理。

4. MPLS组网环境中的队列调度

当网络中报文流量过大时，易造成网络拥塞。队列调度是在网络发生拥塞时指定一个资源的调度策略，以决定报文转发的处理次序。

在 MPLS 组网环境中，如果需要使用报文优先级进行优先级映射并对报文进行队列调度，需将使能 MPLS 功能的设备上转发报文的端口配置为信任 802.1p 优先级或 DSCP 优先级。有关队列调度和优先级映射的详细信息，请参见“ACL 和 QoS 配置指导”中的“QoS”。

在 MPLS L2VPN、VPLS 和 MPLS L3VPN 组网环境中，还需在 CE 设备上转发报文的端口配置为信任 802.1p 优先级或 DSCP 优先级，有关 CE 设备的详细介绍请参见“MPLS 配置指导”中的“MPLS L3VPN”。

1.1.5 LDP

LDP 是标签发布协议的一种，用来动态建立 LSP。通过 LDP，LSR 可以把网络层的路由信息映射到数据链路层的交换路径上。

1. LDP基本概念

- LDP 会话

LDP 会话建立在 TCP 连接之上，用于在 LSR 之间交换标签映射、标签释放、差错通知等消息。

- LDP 对等体

LDP 对等体是指相互之间存在 LDP 会话，并通过 LDP 会话交换标签－FEC 映射关系的两个 LSR。

2. LDP消息类型

LDP 协议主要使用四类消息：

- 发现（Discovery）消息：用于通告和维护网络中 LSR 的存在；
- 会话（Session）消息：用于建立、维护和终止 LDP 对等体之间的会话；
- 通告（Advertisement）消息：用于创建、改变和删除“标签—FEC”映射关系；
- 通知（Notification）消息：用于提供建议性的消息和差错通知。

为保证 LDP 消息的可靠发送，除了发现消息使用 UDP 传输外，LDP 的会话消息、通告消息和通知消息都使用 TCP 传输。

3. LDP工作过程

LDP 主要包括以下四个阶段：

(1) 发现阶段

所有希望建立 LDP 会话的 LSR 都周期性地发送 Hello 消息，通告自己的存在。通过 Hello 消息，LSR 可以自动发现它的 LDP 对等体。

LDP 对等体发现机制分为两种：

- 基本发现机制：用于发现本地的 LDP 对等体，即通过链路层直接相连的 LSR，并与其建立 Link hello 邻接关系。这种方式下，LSR 周期性地向“子网内所有路由器”的组播地址 224.0.0.2 发送 LDP 链路 Hello 消息，以便链路层直接相连的 LSR 发现此 LDP 对等体。
- 扩展发现机制：用于发现远端的 LDP 对等体，即不通过链路层直接相连的 LSR，并与其建立 Targeted hello 邻接关系。这种方式下，LSR 周期性地向指定的 IP 地址发送 LDP 目标 Hello 消息，以便指定 IP 地址对应的 LSR 发现此 LDP 对等体。

两个 LSR 为基本发现机制和扩展发现机制配置的传输地址（用来建立 TCP 连接的源 IP 地址）相同时，这两个 LSR 之间可以同时建立 Link hello 邻接关系和 Targeted hello 邻接关系，并且 Link hello 邻接关系和 Targeted hello 邻接关系关联到同一个会话。在 LDP 对等体之间存在直连（只有一跳）和非直连（多于一跳）多条路径的组网环境中，同时建立 Link hello 邻接关系和 Targeted hello 邻接关系可以实现利用扩展发现机制来保护与对等体的会话。当直连链路出现故障时，Link hello 邻接关系将被删除。如果此时非直连链路正常工作，则 Targeted hello 邻接关系依然存在，因此，LDP 会话不会被删除，基于该会话的 FEC—标签绑定等信息也不会删除。直连链路恢复后，不需要重新建立 LDP 会话、重新学习 FEC—标签绑定等信息，从而加快了 LDP 收敛速度。

两个 LSR 为基本发现机制和扩展发现机制配置的传输地址不同时，如果在这两个 LSR 之间已经建立了一种邻接关系，则无法再建立另一种邻接关系。

(2) 会话建立与维护

发现 LDP 对等体后，LSR 开始建立会话。这一过程又可分为两步：

- 建立传输层连接，即在 LSR 之间建立 TCP 连接；
- 对 LSR 之间的会话进行初始化，协商会话中涉及的各种参数，如 LDP 版本、标签发布方式、Keepalive 定时器值等。

会话建立后，LDP 对等体之间通过不断地发送 Hello 消息和 Keepalive 消息来维护这个会话。

(3) LSP 建立与维护

LDP 通过发送标签请求和标签映射消息，在 LDP 对等体之间通告 FEC 和标签的绑定关系，从而建立 LSP。

LSP的建立过程，请参见“1.1.3 LSP建立与标签的发布和管理”。

(4) 会话撤销

在以下情况下，LSR 将撤销 LDP 会话：

- LSR 通过周期性发送 Hello 消息表明自己希望与邻居 LSR 继续维持这种邻接关系。如果 Hello 保持定时器超时仍没有收到新的 Hello 消息，则删除 Hello 邻接关系。一个 LDP 会话上可能存在多个 Hello 邻接关系。当 LDP 会话上的最后一个 Hello 邻接关系被删除后，LSR 将发送通知消息，结束该 LDP 会话。

- LSR 通过 LDP 会话上传送的 LDP PDU（LDP PDU 中携带一个或多个 LDP 消息）来判断 LDP 会话的连通性。如果在会话保持定时器（Keepalive 定时器）超时前，LDP 对等体之间没有需要交互的信息，LSR 将发送 Keepalive 消息给 LDP 对等体，以便维持 LDP 会话。如果会话保持定时器超时，没有收到任何 LDP PDU，LSR 将关闭 TCP 连接，结束 LDP 会话。
- LSR 还可以发送 Shutdown 消息，通知它的 LDP 对等体结束 LDP 会话。因此，LSR 收到 LDP 对等体发送的 Shutdown 消息后，将结束与该 LDP 对等体的会话。

1.1.6 协议规范

与 MPLS 相关的协议规范有：

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

1.2 MPLS配置任务简介

表1-1 MPLS 配置任务简介

配置任务		说明		详细配置
使能MPLS功能		必选		1.3
建立静态LSP		必选	两种建立 LSP 的方式，必选其一	1.4
利用 LDP 建立动态 LSP	配置MPLS LDP能力	必选		1.5.1
	配置LDP本地会话的参数	可选		1.5.2
	配置LDP远端会话的参数	可选		1.5.3
	配置PHP特性	可选		1.5.4
	配置LSP触发策略	可选		1.5.5
	配置LDP标签分配控制方式	可选		1.5.6
	配置LDP环路检测	可选		1.5.7
	配置LDP MD5认证	可选		1.5.8
	配置LDP标签过滤	可选		1.5.9
	配置发送的LDP报文的DSCP优先级	可选		1.5.10
维护LDP会话	配置MPLS LDP与BFD联动	可选		1.6.1
	重启LDP会话	可选		1.6.2
管理及优化MPLS转发功能	配置Ingress节点对TTL的处理方式	可选		1.7.1
	配置MPLS的TTL超时消息发送功能	可选		1.7.2
	配置LDP GR	可选		1.7.3
配置MPLS统计功能	使能MPLS统计功能	可选		1.8.1
检测LSP	MPLS LSP Ping功能	可选		1.9.1

配置任务		说明	详细配置
	MPLS LSP Tracert功能	可选	1.9.2
	配置BFD检测LSP功能	可选	1.9.3
	配置周期性LSP Tracert功能	可选	1.9.4
MPLS的Trap功能		可选	1.10

说明

目前支持 MPLS 能力的接口类型为三层以太网接口、三层聚合接口和 VLAN 接口（Vlan-interface）。三层以太网接口是指工作模式被配置成三层模式的以太网端口，有关以太网端口工作模式切换的操作，请参见“二层技术-以太网交换配置指导”中的“以太网端口配置”部分。

1.3 使能MPLS功能

MPLS 域中参与 MPLS 转发的路由器上，必须先使能 MPLS 功能，才能进行 MPLS 其它特性的配置。

1.3.1 配置准备

在使能 MPLS 功能之前，需要完成以下任务：

- 配置链路层协议，保证链路层通信正常。
- 配置接口的网络层地址，使各相邻节点网络层可达。
- 配置单播静态路由或 IGP 协议，保证各 LSR 在网络层互通。

1.3.2 使能MPLS功能

表1-2 使能 MPLS 功能

操作	命令	说明
进入系统视图	**system-view**	-
配置本节点的LSR ID	**mpls lsr-id** *lsr-id*	必选 缺省情况下，未配置LSR ID
使能本节点的MPLS能力，并进入MPLS视图	**mpls**	必选 缺省情况下，未使能本节点的MPLS能力
退回系统视图	**quit**	-
进入需要转发MPLS报文的接口视图	**interface** *interface-type interface-number*	-

操作	命令	说明
使能接口的MPLS能力	**mpls**	必选 缺省情况下，未使能接口的MPLS能力

说明

LSR ID 使用 IP 地址格式，在 MPLS 域内唯一。推荐使用 Loopback 接口的 IP 地址作为 LSR ID。

1.4 建立静态LSP

建立静态 LSP 需要遵循的原则是：上游 LSR 出标签的值就是下游 LSR 入标签的值。

1.4.1 配置准备

在配置静态 LSP 之前，需完成以下任务：

- 确定静态 LSP 的入节点、中间节点和出节点。
- 在各节点上使能 MPLS 功能。
- 在入节点上建立静态 LSP 时，需确保该节点上存在 FEC 目的地址对应的路由。中间节点和出节点上不需要存在 FEC 目的地址对应的路由。

1.4.2 建立静态LSP

表1-3 建立静态 LSP

操作	命令	说明
进入系统视图	**system-view**	-
为Ingress节点配置静态LSP	**static-lsp ingress** *lsp-name* **destination** *dest-addr mask* | *mask-length* } { **nexthop** *next-hop-addr* | **out-label** *out-label*	必选
为Transit节点配置静态LSP	**static-lsp transit** *lsp-name* **incoming-interface** *interface-type interface-number* **in-label** *in-label* **nexthop** *next-hop-addr* **out-label** *out-label*	必选
为Egress节点配置静态LSP	**static-lsp egress** *lsp-name* **incoming-interface** *interface-type interface-number* **in-label** *in-label*	必选

说明

- 在 Ingress 节点上配置静态 LSP 时，指定的下一跳必须与路由表中最优下一跳保持一致。如果是通过静态路由配置路由信息，则静态路由必须与静态 LSP 指定的下一跳一致。
- 配置 Ingress 和 Transit 时，本地设备上接口的公网地址不能作为下一跳地址。
- 有关 IP 静态路由的介绍请参见“三层技术-IP 路由配置指导”中的“静态路由”。

1.5 利用LDP建立动态LSP

1.5.1 配置MPLS LDP能力

表1-4 配置 MPLS LDP 能力

操作	命令	说明
进入系统视图	**system-view**	-
使能本节点的LDP能力，并进入MPLS-LDP视图	**mpls ldp**	必选 缺省情况下，未使能LDP能力
配置LDP的LSR ID	**lsr-id** *lsr-id*	可选 缺省情况下，LDP的LSR ID与MPLS LSR ID相同
退回系统视图	**quit**	-
进入建立LDP会话的接口视图	**interface** *interface-type interface-number*	-
使能接口的LDP能力	**mpls ldp**	必选 缺省情况下，在接口上未使能LDP能力

说明

- 取消接口的 LDP 能力将会导致接口下的所有 LDP 会话中断，基于这些会话的所有 LSP 也将被删除。
- 通常情况下 LDP 使用缺省的 MPLS LSR ID 即可，在某些使用 VPN 实例（例如 MPLS L3VPN）的组网方案中，如果 VPN 与公网地址空间重叠，则需要为 LDP 另外配置 LSR ID，以保证 TCP 连接能够正常建立。

1.5.2 配置LDP本地会话的参数

本地 LDP 对等体之间建立的 LDP 会话，称为本地会话。建立 LDP 本地会话前：

- 需要先确认对端的 LDP 传输地址，并保证两端的 LDP 传输地址是相互可达的，这样才能建立 TCP 连接。
- 如果 LSR 之间的 LDP 会话数较多或 CPU 利用率较高时，需要对定时器进行调节确保 LDP 会话的稳定性。

表1-5 配置 LDP 本地会话的参数

操作	命令	说明
进入系统视图	**system-view**	-
进入建立LDP会话的接口视图	**interface** *interface-type interface-number*	-
配置链路Hello保持定时器的值	**mpls ldp timer hello-hold** *value*	可选 缺省情况下，链路Hello保持定时器的值为15秒
配置链路Keepalive保持定时器的值	**mpls ldp timer keepalive-hold** *value*	可选 缺省情况下，链路Keepalive保持定时器的值为45秒
配置LDP传输地址	**mpls ldp transport-address** { *ip-address* | **interface** }	可选 缺省情况下，传输地址是本LSR的LSR ID

说明

选择指定 IP 地址作为 LDP 传输地址时，配置的 IP 地址应为设备接口的 IP 地址，否则 LDP 会话将无法建立。

1.5.3 配置LDP远端会话的参数

远端 LDP 对等体之间建立的 LDP 会话，称为远端会话。LDP 远端会话主要应用于 Martini 方式的 MPLS L2VPN 和 Martini 方式的 VPLS，详细介绍请参见“MPLS 配置指导”的“MPLS L2VPN”和“VPLS”。

建立远端 LDP 会话前：

- 需要先确认对端的 LDP 传输地址，并保证两端的 LDP 传输地址是相互可达的，这样才能建立 TCP 连接。
- 如果 LSR 之间的 LDP 会话数较多或 CPU 利用率较高时，需要对定时器进行调节确保 LDP 会话的稳定性。

表1-6 配置 LDP 远端会话的参数

操作	命令	说明
进入系统视图	**system-view**	-

操作	命令	说明
创建远端对等体实体并进入MPLS-LDP远端对等体视图	**mpls ldp remote-peer** *remote-peer-name*	必选
指定LDP远端对等体的IP地址	**remote-ip** *ip-address*	必选
配置通过远端会话通告基于地址前缀的标签	**prefix-label advertise**	可选 缺省情况下，不会通过远端会话通告基于地址前缀的标签
配置目标Hello保持定时器的值	**mpls ldp timer hello-hold** *value*	可选 缺省情况下，目标Hello保持定时器的值为45秒
配置目标Keepalive保持定时器的值	**mpls ldp timer keepalive-hold** *value*	可选 缺省情况下，目标Keepalive保持定时器的值为45秒
配置LDP传输地址	**mpls ldp transport-address** *ip-address*	可选 缺省情况下，传输地址是本LSR的LSR ID

说明

- 配置的 LDP 传输地址应为设备接口的 IP 地址，否则 LDP 会话将无法建立。
- 即将配置的远端对等体的 IP 地址不能与已经存在的远端对等体 IP 地址重复。
- 如果对等体与指定的远端对等体之间已经存在本地邻接关系，则远程邻接关系将不能建立。如果已经存在远程邻接关系，又为远端对等体创建了本地邻接关系，则在本地对等体和远端对等体的 LDP 传输地址、Keepalive 保持时间配置一致时，将建立本地邻接关系，远端对等体将被删除；本地对等体和远端对等体的 LDP 传输地址或 Keepalive 保持时间配置不一致时，将无法建立本地邻接关系，继续保持远程邻接关系。即两个 LSR 之间只能存在一个远端会话或一个本地会话，并且本地会话的优先级高于远端会话。

说明

缺省情况下，LDP 不会通过远端会话通告基于地址前缀的标签映射消息，远端会话只用来为 L2VPN 传送消息。有关远端会话的应用请参见“MPLS 配置指导”中的“MPLS L2VPN”中的“配置 Martini 方式 MPLS L2VPN”部分。

1.5.4 配置PHP特性

表1-7 配置 PHP 特性

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
配置Egress节点向倒数第二跳分配的标签值	**label advertise** { **explicit-null** | **implicit-null** | **non-null** }	可选 缺省情况下，Egress节点向倒数第二跳分配隐式空标签（**implicit-null**）

- 选择哪种标签值需要根据倒数第二跳节点对 PHP 特性的支持情况而定。如果倒数第二跳节点支持 PHP 特性，则可以为其分配显式或隐式空标签；如果倒数第二跳节点不支持 PHP 特性，则为其正常分配标签。
- 设备作为倒数第二跳时，支持 PHP 特性。
- 会话建立后，需使用 **reset mpls ldp** 命令重启 LDP 会话，本命令的配置才能生效。

1.5.5 配置LSP触发策略

在 LSR 上通过配置 LSP 触发策略，可以限制只有通过策略过滤的路由信息才能触发 LSP 的建立，从而控制 LSP 的数量，避免 LSP 数量过多导致设备运行不稳定。

LSP 触发策略包括以下两种：

- 所有路由项都会触发 LDP 建立 LSP。
- 利用 IP 地址前缀列表对路由项进行过滤，被 IP 地址前缀列表拒绝的静态路由和 IGP 路由项不能触发建立 LSP。采用这种 LSP 触发策略时，需要创建 IP 地址前缀列表，创建方法请参见“三层技术-IP 路由配置指导”中的“路由策略”。

表1-8 配置 LSP 触发策略

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
配置LSP的触发策略	**lsp-trigger** [**vpn-instance** *vpn-instance-name*] { **all** | **ip-prefix** *prefix-name* }	可选 缺省情况下，只有32位掩码的主机路由能够触发LDP建立LSP

说明

- 建立 LSP 需要 LSR 上存在精确匹配的路由项。例如，要使用 32 位掩码的 Loopback 接口地址建立 LSP，则 LSR 上必须存在精确匹配的主机路由才能触发建立 LSP。
- 如果指定 **vpn-instance** *vpn-instance-name* 参数，则配置指定 VPN 的 LSP 触发策略；如果未指定 **vpn-instance** *vpn-instance-name* 参数，则配置公网路由的 LSP 触发策略。

1.5.6 配置LDP标签分配控制方式

使能标签重发布功能后，LSR 会周期性地查找尚未分配标签的 FEC，为其分配标签，并将标签和 FEC 绑定通告给其他的 LSR。同时，用户可以根据需要控制重发布标签的时间间隔。

表1-9 配置 LDP 标签分配控制方式

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS-LDP视图	**mpls ldp**	-
配置标签分配控制方式	**label-distribution** { **independent** | **ordered** }	可选 缺省情况下，标签分配控制方式为有序方式（**ordered**） 会话建立后，需重启LDP会话本命令的配置才能生效
使能DU模式下标签重发布功能	**du-readvertise**	可选 缺省情况下，DU模式下已使能标签重发布功能
配置DU模式下定期重发布标签的时间间隔	**du-readvertise timer** *value*	可选 缺省情况下，DU模式下定期重发布标签的时间间隔为30秒

1.5.7 配置LDP环路检测

在 MPLS 域中建立 LSP 也要防止产生环路，LDP 环路检测机制可以检测 LSP 环路的出现，并避免发生环路。

LDP 环路检测有两种方式：

(1) 最大跳数

在传递标签映射（或者标签请求）的消息中包含跳数信息，每经过一跳该值就加一。当该值达到规定的最大值时即认为出现环路，终止 LSP 的建立过程。

(2) 路径向量

在传递标签映射（或者标签请求）的消息中记录路径信息，每经过一跳，相应的设备就检查自己的 LSR ID 是否在此记录中。如果记录中没有自身的 LSR ID，就会将自身的 LSR ID 添加到该记录中；如果记录中已有本 LSR 的记录，则认为出现环路，终止 LSP 的建立过程。

采用路径向量方式进行环路检测时，也需要规定 LSP 路径的最大跳数，当路径的跳数达到配置的最大值时，也会认为出现环路，终止 LSP 的建立过程。

表1-10 配置 LDP 环路检测

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS-LDP视图	**mpls ldp**	-
开启环路检测功能	**loop-detect**	必选 缺省情况下，环路检测功能处于关闭状态
配置环路检测的最大跳数	**hops-count** *hop-number*	可选 缺省情况下，环路检测最大跳数为32
配置路径向量方式下LSP的最大跳数	**path-vectors** *pv-number*	可选 缺省情况下，路径向量方式下LSP的最大跳数为32

说明

- LDP 对等体上的环路检测配置必须一致，否则无法建立 LDP 会话。
- 如果对 MPLS 域进行环路检测，则必须在 MPLS 域内所有 LSR 上都开启环路检测功能。
- 环路检测功能需要在所有接口使能 LDP 之前进行配置。
- 所有环路检测的配置，只对修改后建立的 LSP 有效，修改环路检测配置不影响已经建立的 LSP。可以通过在用户视图下执行 **reset mpls ldp** 命令的方法，使环路检测配置对已经建立的 LSP 生效。
- 使能 LDP 环路检测功能，可能会导致 LSP 重新更新，产生冗余消息，消耗过多的系统资源，推荐使用路由协议的环路预防功能。

1.5.8 配置LDP MD5 认证

LDP 会话建立在 TCP 连接之上。为了提高 LDP 会话的安全性，可以配置在建立 LDP 会话使用的 TCP 连接时进行 MD5 认证，保证只有对端与本端配置了相同的认证密码时，才会与其建立 TCP 连接。

表1-11 配置 LDP MD5 认证

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS-LDP视图	**mpls ldp**	-

操作	命令	说明
使能LDP的MD5认证功能	**md5-password** { **cipher** | **plain** } *peer-lsr-id password*	必选 缺省情况下，未使能LDP的MD5认证功能

说明

要想在 LDP 对等体之间成功建立 LDP 会话，必须保证 LDP 对等体上的 LDP MD5 认证配置一致。

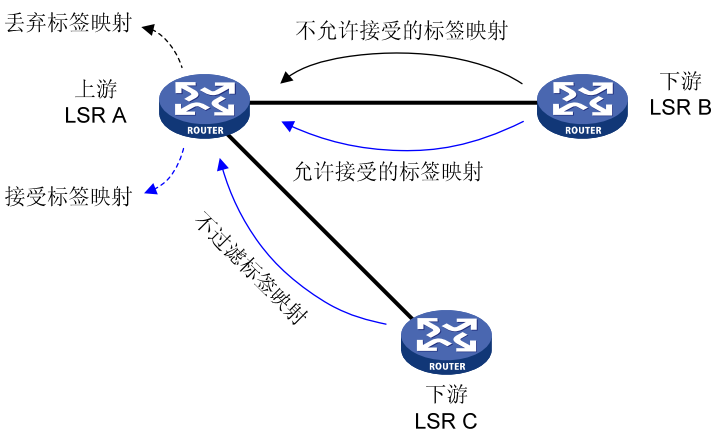
1.5.9 配置LDP标签过滤

LDP 标签过滤提供了标签接受控制和标签通告控制两种机制，用来实现对接收标签和通告标签的过滤。在复杂的 MPLS 网络环境中，通过 LDP 标签过滤可以规划动态建立的 LSP，并避免设备接收和通告大量的标签映射。

(1) 标签接受控制

标签接受控制用来实现对接收的标签映射进行过滤，即上游LSR对指定的下游LSR通告的标签映射进行过滤，只接受指定地址前缀的标签映射。如 图 1-8，上游设备LSR A对下游设备LSR B通告的标签进行过滤，只有FEC的目的地址通过指定地址前缀列表过滤后，才会接受该FEC对应的标签映射；对下游设备LSR C通告的标签不进行过滤。

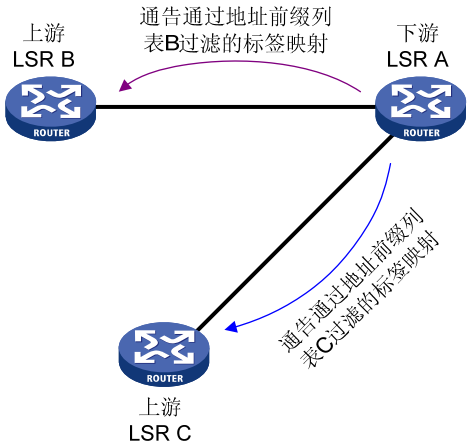
图1-8 标签接受控制示意图



(2) 标签通告控制

标签通告控制用来实现对通告的标签映射进行过滤，即下游LSR只将指定地址前缀的标签映射通告给指定的上游设备。如 图 1-9，下游设备LSR A将FEC目的地址通过地址前缀列表B过滤的标签映射通告给上游设备LSR B；将FEC目的地址通过地址前缀列表C过滤的标签映射通告给上游设备LSR C。

图1-9 标签通告控制示意图



1. 配置准备

在配置 LDP 标签过滤之前，需要先创建 IP 地址前缀列表，创建方法请参见“三层技术-IP 路由配置指导”中的“路由策略”。

2. 配置LDP标签过滤

表1-12 配置 LDP 标签过滤

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS-LDP视图	**mpls ldp**	-
配置标签接受控制策略	**accept-label peer** *peer-id* **ip-prefix** *ip-prefix-name*	可选 缺省情况下，接受LDP对等体通告的所有标签映射
配置标签通告控制策略	**advertise-label ip-prefix** *ip-prefix-name* [**peer** *peer-ip-prefix-name*]	必选 缺省情况下，不对通告的标签映射进行过滤

说明

在下游 LSR 上配置标签通告控制策略与在上游 LSR 上配置标签接受控制策略具有相同的效果，推荐使用前者，以减轻网络负担。

1.5.10 配置发送的LDP报文的DSCP优先级

在 IPv4 报文头中，包含一个 8bit 的 ToS 字段，用于标识 IP 报文的服务类型。RFC 2474 对这 8 个 bit 进行了定义，将前 6 个 bit 定义为 DSCP 优先级，最后 2 个 bit 作为保留位。在报文传输的过程中，DSCP 优先级可以被网络设备识别，并作为报文传输优先程度的参考。

用户可以对本设备发送的 LDP 报文的 DSCP 优先级进行配置。

表1-13 配置发送的 LDP 报文的 DSCP 优先级

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS-LDP视图	**mpls ldp**	-
配置发送的LDP报文的DSCP优先级	**dscp** *dscp-value*	必选 缺省情况下，发送的LDP报文的DSCP优先级为48

1.6 维护LDP会话

1.6.1 配置MPLS LDP与BFD联动

MPLS 本身无法快速检测到邻居（或链路）的故障。LDP 远端对等体之间的通信发生故障时，将导致 LDP 会话 down，MPLS 报文转发失败。MPLS LDP 与 BFD 联动功能能够快速检测到 LDP 远端对等体之间的通信故障，提升现有 MPLS 网络的性能。

说明

- 有关 BFD 的介绍及配置方法，请参见“可靠性配置指导”中的“BFD”。
- 一条 LSP 只能与一个 BFD 会话绑定。

表1-14 配置 MPLS LDP 与 BFD 联动

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS-LDP远端对等体视图	**mpls ldp remote-peer** *remote-peer-name*	-
为LDP远端对等体开启BFD链路检测功能	**remote-ip bfd**	必选 缺省情况下，LDP远端对等体的BFD链路检测功能处于关闭状态

说明

MPLS LDP 与 BFD 联动只能用来检测远端 LDP 对等体之间的通信是否存在故障，相关配置举例请参见“MPLS 配置指导”中的“VPLS”。

1.6.2 重启LDP会话

LDP 会话状态为 up 后，修改 LDP 会话的任何参数，都会引起 LDP 会话不能正常进行。此时，需要重启 LDP 会话，重新协商各种参数，建立 LDP 会话。

在用户视图下执行 **reset mpls ldp** 命令可以重启 LDP 会话。

表1-15 重启 LDP 会话

操作	命令
重启LDP会话	**reset mpls ldp** [**all** | [**vpn-instance** *vpn-instance-name*] [*fec mask* | **peer** *peer-id*]]

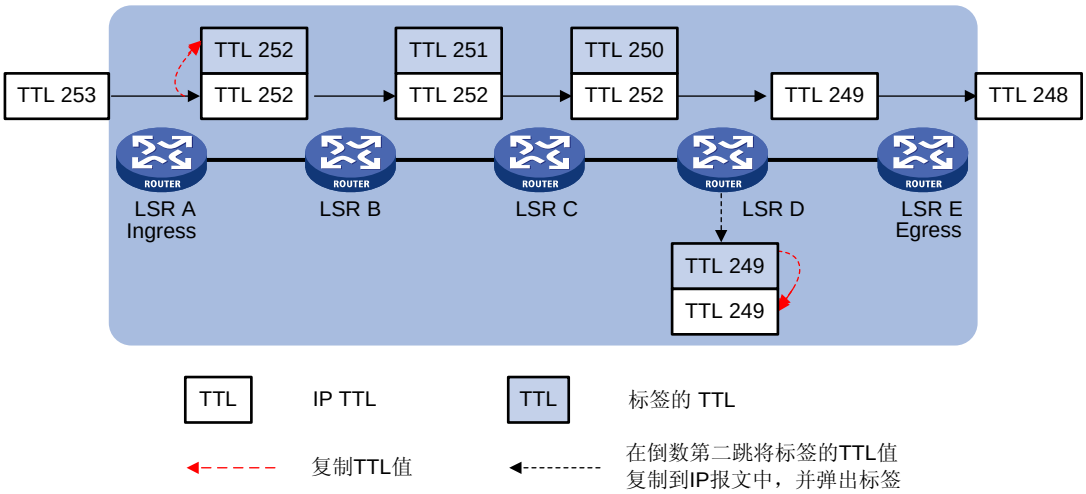
1.7 管理及优化MPLS转发功能

1.7.1 配置Ingress节点对TTL的处理方式

Ingress 节点对 TTL 域的处理方式分为以下两种：

- 使能 IP TTL 复制功能：Ingress 节点为报文压入标签时，将原 IP 报文中的 TTL 值复制到新增加的标签的 TTL 域。LSR 转发标签报文时，对栈顶标签的 TTL 值做减一操作。标签出栈时，再将栈顶标签的 TTL 值复制回 IP 报文。使用这种方式时，报文沿着 LSP 传输的过程中，TTL 逐跳递减，Tracert 的结果将反映报文实际经过的路径。

图1-10 使能 IP TTL 复制功能时的处理过程



- 未使能 IP TTL 复制功能：Ingress 节点为报文压入标签时，不会将原 IP 报文中的 TTL 值复制到新增加的标签的 TTL 域，标签的 TTL 取值为 255。LSR 转发标签报文时，对栈顶标签的 TTL 值做减一操作。标签出栈时，LSR 将 IP TTL 和标签 TTL 中的较小者，作为 IP TTL 的值。使用这种方式时，Tracert 的结果不包括 MPLS 骨干网络中的每一跳，就好像 Ingress 路由器与 Egress 路由器是直连的，从而隐藏 MPLS 骨干网络的结构。

图1-11 未使能 IP TTL 复制功能时的处理过程

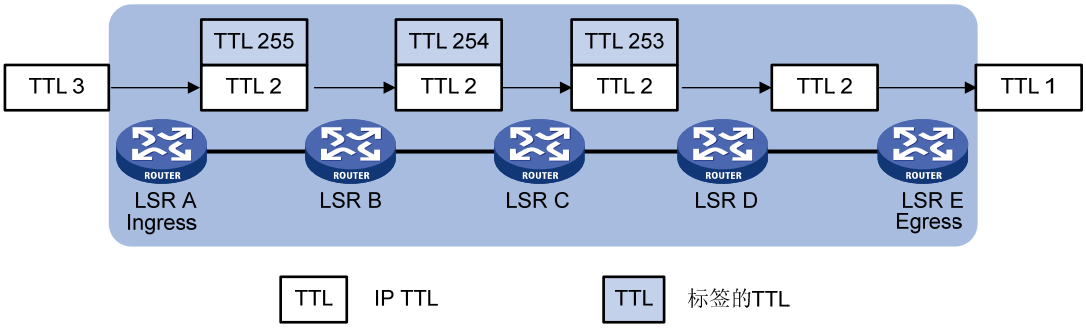


表1-16 配置 MPLS 的 IP TTL 复制功能

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
使能MPLS的IP TTL复制功能	**ttl propagate** { **public** | **vpn** }	可选 缺省情况下，只对公网报文使能IP TTL复制功能

注意

- 在 MPLS 域内部，MPLS 报文多层标签之间的 TTL 值总是互相复制，**ttl propagate** 命令只能决定是否将 IP TTL 复制到标签的 TTL 域，因此只有在 Ingress 上执行本命令才会生效。
- MPLS 的 IP TTL 复制功能对本地发送报文没有影响，本地发送报文都将进行 IP TTL 复制，从而保证本地管理员能够使用 Tracert 检测网络。
- 如果配置 **ttl propagate vpn** 命令使能对 VPN 报文的 IP TTL 复制功能，则建议在所有相关 PE（Provider Edge，运营商边缘）设备上都使能此功能，以保证不同的 PE 上执行 Tracert 得到的结果一致。PE 的详细介绍请参见“MPLS 配置指导”中的“MPLS L3VPN”。

1.7.2 配置MPLS的TTL超时消息发送功能

使能 MPLS 的 TTL 超时消息发送功能后，当 LSR 收到 TTL 为 1 的含有标签的 MPLS 报文时，LSR 会生成 ICMP 的 TTL 超时消息。

LSR 将 TTL 超时消息回应给报文发送者的方式有两种：

- 如果 LSR 上存在到达报文发送者的路由，则可以通过 IP 路由，直接向发送者回应 TTL 超时消息。
- 如果 LSR 上不存在到达报文发送者的路由，则需要沿着发送 MPLS 报文的 LSP 转发 TTL 超时消息，由 Egress 节点将该消息返回给发送者。

通常情况下，收到的 MPLS 报文只带一层标签时，LSR 采用第一种方式回应 TTL 超时消息；收到的 MPLS 报文包含多层标签时，LSR 采用第二种方式回应 TTL 超时消息。

但是，在 MPLS VPN 中，ASBR（Autonomous System Boundary Router，自治系统边界路由器）、HoVPN 组网应用中的 SPE（Superstratum PE or Sevice Provider-end PE，上层 PE 或运营商侧 PE）和嵌套 VPN 应用中的运营商骨干网 PE，接收到的承载 VPN 报文的 MPLS 报文可能只有一层标签，此时，这些设备上并不存在到达报文发送者的路由，无法采用第一种方法回应 TTL 超时消息。通过配置 **undo ttl expiration pop** 命令，可以保证只带一层标签的 MPLS 报文 TTL 超时时，使用 LSP 路径转发 TTL 超时消息，由 Egress 节点将该消息返回给发送者。

说明

有关 HoVPN 和嵌套 VPN 的详细介绍请参见“MPLS 配置指导”中的“MPLS L3VPN”。

表1-17 配置 MPLS 的 TTL 超时消息发送功能

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
使能MPLS的TTL超时消息发送功能	**ttl expiration enable**	可选 缺省情况下，MPLS的TTL超时消息发送功能处于开启状态
配置仅有一层标签的MPLS报文TTL超时时，沿本地IP路由返回ICMP报文	**ttl expiration pop**	可选 缺省情况下，对于一层标签的MPLS报文，TTL超时时沿本地IP路由返回ICMP差错报文 多层标签MPLS报文不受该命令控制，当MPLS报文的TTL超时时，始终沿LSP路径转发ICMP差错报文

1.7.3 配置LDP GR

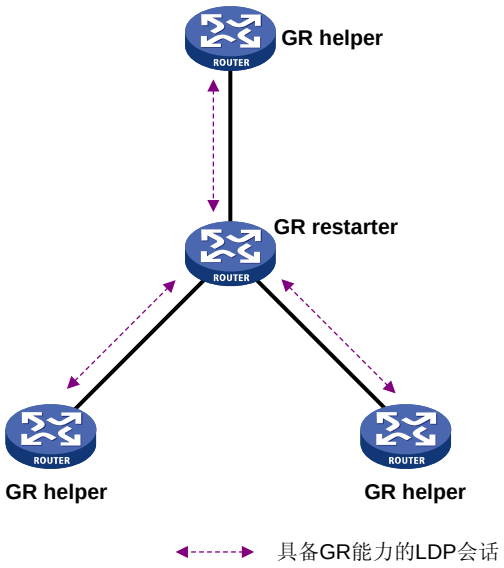
1. LDP GR简介

LDP GR（Graceful Restart，平滑重启）利用 MPLS 转发平面与控制平面分离的特点，在信令协议或控制平面出现异常时，保留标签转发表项，LSR 依然根据该表项转发报文，从而保证数据传输不会中断。

参与 LDP GR 过程的设备分为以下两种：

- GR restarter：GR 重启路由器，指由管理员手工或设备故障触发而重启协议的设备，它必须具备 GR 能力。
- GR helper：GR restarter 的邻居，与重启的 GR restarter 保持邻居关系，并协助其恢复重启前的转发状态，它也必须具备 GR 能力。

图1-12 LDP GR



如 图 1-12 所示，LDP对等体在建立LDP会话时协商GR能力，只有双方都支持LDP GR时，建立的LDP会话才支持LDP GR。

LDP GR 的工作过程如下：

(1) GR restarter 发生重启后，GR restarter 启动转发状态保持定时器，保留所有 MPLS 转发表项，并将其标记为 stale。

(2) GR helper 检测到与 GR restarter 之间的会话 down 后，将所有通过此会话学习到的 FEC-Label 绑定标记为 stale，并在 FT 重连时间内保留这些 FEC-Label 绑定。FT 重连时间的取值为对端 GR restarter 通告的重连时间和本地配置的 LDP 邻居存活定时器中的较小者。

(3) 如果在 FT 重连时间内，LDP 会话重建失败，则 Helper 删除这些标签为 stale 的 FEC-Label 绑定。

(4) 如果 LDP 会话建立成功，则 GR restarter 和 GR helper 在 LDP 恢复时间内，通过新建立的 LDP 会话交互标签映射，更新标签转发表，清除转发表项的 stale 标签。LDP 恢复时间的取值为本地配置的 LDP 恢复时间和对端 GR restarter 设备配置的 LDP 恢复时间中的较小者

(5) LDP 恢复定时器超时后，GR helper 删除标记仍为 stale 的 FEC－标签映射。

(6) MPLS 转发状态保持定时器超时后，GR restarter 删除标记仍为 stale 的标签转发表项。

2. 配置准备

在配置 LDP GR 之前，需完成以下任务：

作为 GR restarter 的设备上和 GR helper 的设备均配置 MPLS LDP 能力。

说明

设备既可以作为 GR restarter，又可以作为 GR helper，设备的角色由该设备在 LDP GR 过程中的作用决定。

---

3. 配置LDP GR

表1-18 配置 LDP GR

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS-LDP视图	**mpls ldp**	-
使能MPLS LDP协议的GR能力	**graceful-restart**	必选 缺省情况下，MPLS LDP协议的GR能力处于关闭状态
配置FT重连定时器的值	**graceful-restart timer reconnect** *timer*	可选 缺省情况下，FT重连定时器的值为300秒
配置LDP邻居存活定时器的值	**graceful-restart timer neighbor-liveness** *timer*	可选 缺省情况下，LDP邻居存活定时器的值为120秒
配置LDP恢复定时器的值	**graceful-restart timer recovery** *timer*	可选 缺省情况下，LDP恢复定时器的值为300秒

(2) 平滑重启 MPLS LDP

通过平滑重启 MPLS LDP 功能，可以测试 LDP GR 功能是否生效，即测试在 LDP 协议重启过程中报文转发路径是否改变，是否可以不间断地转发数据。

表1-19 平滑重启 MPLS LDP

操作	命令	说明
平滑重启MPLS LDP	**graceful-restart mpls ldp**	必选 请在用户视图下执行此命令

说明

graceful-restart mpls ldp 命令用来在不发生主备倒换的情况下测试 MPLS LDP GR 功能。其他情况下，请不要使用此命令。

1.8 配置MPLS统计功能

本配置任务用来开启 MPLS 统计功能，以便用户通过显示命令查看 MPLS 的统计信息。

1.8.1 配置LSP统计数据输出时间间隔

只有配置获取 LSP 统计数据的时间间隔后，用户才能通过显示命令查看 MPLS 的统计信息。

表1-20 配置 MPLS 统计功能

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
配置获取LSP统计数据的时间间隔	**statistics interval** *interval-time*	必选 缺省情况下，获取LSP统计数据的时间间隔为0秒，即不获取统计信息

1.9 检测LSP

在 MPLS 中，LSP 转发数据失败时，负责建立 LSP 的 MPLS 控制平面无法检测或不能及时发现这种错误，这会给网络维护带来困难。为了及时发现 LSP 错误，并定位失效节点，设备上提供了如下几种机制：

- MPLS LSP Ping 功能
- MPLS LSP Tracert 功能
- BFD 检测 LSP 功能
- 周期性 LSP Tracert 功能

1.9.1 MPLS LSP Ping功能

MPLS LSP Ping 功能用来对 LSP 的可达性进行检测。采取方法是：在 Ingress 节点为 MPLS Echo Request 报文压入待检测 FEC 对应的标签；经过 LSP 将该报文转发到 Egress 节点；Egress 节点处理该报文后，回应 MPLS Echo Reply 报文；如果 Ingress 节点接收到表示成功的 MPLS Echo Reply 报文，则说明该 LSP 可以用于数据转发；如果 Ingress 节点接收到带有错误码的 MPLS Echo Reply 报文，则说明该 LSP 存在故障。

说明

在 Ingress 节点和 Egress 节点间存在等价 LSP 的环境中，不能使用 MPLS LSP Ping 功能来检测路径可达性。

表1-21 MPLS LSP Ping 功能

操作	命令	说明
通过MPLS LSP Ping检测MPLS LSP的可达性	**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*]	必选 可在任意视图下执行本命令

1.9.2 MPLS LSP Tracert功能

MPLS LSP Tracert 用来对 LSP 的错误进行定位。MPLS LSP Tracert 通过沿着 LSP 连续发送 TTL 从 1 到某个值的 MPLS Echo Request 报文，让 LSP 路径上的每一跳收到该报文后，返回 TTL 超时的 MPLS Echo Reply 报文。这样，Ingress 节点可以收集到 LSP 路径上每一跳的信息，从而定位出故障节点。同时，MPLS LSP Tracert 还可用于收集整条 LSP 上每个节点的重要信息，如分配的标签等。

说明

在 Ingress 节点和 Egress 节点间存在等价 LSP 的环境中，不能使用 MPLS LSP Tracert 功能来定位路径中的节点。

1. 配置准备

- 需要在中间设备（源端与目的端之间的设备）上开启 ICMP 超时报文发送功能。
- 需要在目的端开启 ICMP 目的不可达报文发送功能。
- 如果在 Tracert 过程中需要显示 MPLS 信息，需要在源端和中间设备上使能 ICMP 携带扩展信息功能。

有关 ICMP 超时报文发送功能、ICMP 目的不可达报文发送功能、ICMP 携带扩展信息功能的详细介绍请参见“三层技术-IP 业务配置指导”中的“IP 性能优化”。

2. 配置MPLS LSP Tracert功能

表1-22 MPLS LSP Tracert 功能

操作	命令	说明
通过MPLS LSP Tracert对MPLS LSP的错误进行定位	**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*]	必选 可在任意视图下执行本命令

1.9.3 配置BFD检测LSP功能

BFD 检测 LSP 功能通过在 LSP 的 Ingress 节点和 Egress 节点之间建立 BFD 会话，利用 BFD 快速检测 LSP 的连通性。在 Ingress 节点为 BFD 控制报文压入 FEC 对应的标签，沿着 LSP 转发 BFD 控制报文，并根据收到的 Egress 节点的 BFD 控制报文来判断 LSP 的状态。当 BFD 检测到 LSP 故障后，还可以触发 LSP 进行流量切换。

可以通过两种方式建立检测 LSP 的 BFD 会话：

- 静态方式：如果执行 **bfd enable** 命令时通过 **discriminator** 参数指定了本地和远端的鉴别值，则根据指定的鉴别值建立 BFD 会话。该方式用来检测两台设备间从本地到远端和从远端到本地的一对 LSP 隧道。
- 动态方式：如果执行 **bfd enable** 命令时没有通过 **discriminator** 参数指定本地和远端的鉴别值，则自动运行 MPLS LSP Ping 来协商鉴别值，并根据协商好的鉴别值建立 BFD 会话。该方式用来检测两台设备间从本地到远端的一条单向 LSP 隧道。

1. 配置准备

- BFD 会话参数为 MPLS LSR-ID 对应的 Loopback 接口下配置的 BFD 参数，BFD 报文的源地址为 MPLS LSR-ID。因此，配置 BFD 检测 LSP 功能前，需要在 Loopback 接口下配置 IP 地址，并将 LSR-ID 配置为 Loopback 接口的 IP 地址，还可以根据需要在 Loopback 接口下配置 BFD 会话参数。BFD 的详细介绍，请参见“可靠性配置指导”中的“BFD”。
- 使用静态方式建立检测 LSP 的 BFD 会话前，需要确保已经建立从本地到远端和从远端到本地的两条 LSP。

2. 配置BFD检测LSP功能

表1-23 配置 BFD 检测 LSP 功能

操作	命令	说明
进入系统视图	**system-view**	-
使能LSP验证功能，并进入MPLS-LSPV视图	**mpls lspv**	必选 缺省情况下，未使能LSP验证功能
配置使用BFD检测指定FEC对应LSP的连通性	**bfd enable** *destination-address mask-length* [**nexthop** *nexthop-address* [**discriminator local** *local-id* **remote** *remote-id*]]	必选 缺省情况下，未使用BFD检测FEC对应LSP的连通性

说明

- 不能同时使用静态方式和动态方式建立检测同一个 LSP 的 BFD 会话。
- 使用静态方式建立检测 LSP 的 BFD 会话后，不允许修改 BFD 会话的鉴别值。
- 协议中规定：用于检测 LSP 的 BFD 会话中，Ingress 节点工作在主动（Active）模式，Egress 节点工作在被动(Passive)模式。在 Ingress 节点和 Egress 节点上执行 **bfd session init-mode** 命令不会改变节点的工作模式，即使 Ingress 节点和 Egress 节点均配置为被动模式，BFD 会话仍然可以正常建立。
- MPLS LDP 与 BFD 联动功能用来检测 LDP 远端对等体之间的 IP 连通性，而 BFD 检测 LSP 功能用来检测 LSP 的连通性。

1.9.4 配置周期性LSP Tracert功能

周期性 LSP Tracert 功能，即周期性地对 LSP 进行 Tracert 检测，该功能用来对 LSP 的错误进行定位，对转发平面和控制平面一致性进行检测，并通过日志记录检测结果。管理员可以通过查看日志信息，了解 LSP 是否出现故障。

如果同时配置了 BFD 检测 LSP 功能和周期性 LSP Tracert 功能，则周期性 LSP Tracert 检测到转发平面故障或转发平面与控制平面不一致时，会触发删除原有 BFD 会话，并基于控制平面重新建立 BFD 会话。

1. 配置准备

- 需要在中间设备（源端与目的端之间的设备）上开启 ICMP 超时报文发送功能。

- 需要在目的端开启 ICMP 目的不可达报文发送功能。
- 如果在 Tracert 过程中需要显示 MPLS 信息，需要在源端和中间设备上使能 ICMP 携带扩展信息功能。

有关 ICMP 超时报文发送功能、ICMP 目的不可达报文发送功能、ICMP 携带扩展信息功能的详细介绍请参见“三层技术-IP 业务配置指导”中的“IP 性能优化”。

2. 配置周期性LSP Tracert功能

表1-24 配置周期性 LSP Tracert 功能

操作	命令	说明
进入系统视图	**system-view**	-
使能LSP检测功能，并进入MPLS-LSPV视图	**mpls lspv**	必选 缺省情况下，未使能LSP检测功能
使能指定FEC的周期性LSP Tracert功能	**periodic-tracert** *destination-address mask-length* [**-a** *source-ip* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *wait-time* | **-t** *time-out* | **-u** *retry-attempt*] *	必选 缺省情况下，未使能指定FEC的周期性LSP Tracert功能

1.10 MPLS的Trap功能

开启 MPLS 模块的 Trap 功能后，该模块会生成级别为 notifications 的 Trap 报文，用于报告该模块的重要事件。生成的 Trap 报文将被发送到设备的信息中心，通过设置信息中心的参数，最终决定 Trap 报文的输出规则（即是否允许输出以及输出方向）。有关信息中心参数的配置请参见“网络管理和监控配置指导”中的“信息中心”。

表1-25 开启 MPLS 的 Trap 功能

操作	命令	说明
进入系统视图	**system-view**	-
开启MPLS的Trap功能	**snmp-agent trap enable mpls**	必选 缺省情况下，MPLS的Trap功能处于关闭状态

说明

snmp-agent trap enable mpls 命令的详细介绍请参见“网络管理和监控命令参考/SNMP”中的 **snmp-agent trap enable** 命令。

1.11 MPLS显示和维护

1.11.1 显示MPLS运行状态

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 MPLS 的运行情况，用户可以通过查看显示信息验证配置的效果。

表1-26 MPLS 运行状态显示

操作	命令
显示使能了MPLS能力接口的MPLS相关信息	**display mpls interface** [*interface-type interface-number*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示ILM表项信息	**display mpls ilm** [*label*] [**verbose**] [**slot** *slot-number*] [**include** *text* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]
显示MPLS标签的使用状态	**display mpls label** { *label-value1* [**to** *label-value2*] | **all** } [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示LSP信息	**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*]
显示LSP统计信息	**display mpls lsp statistics** [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示BFD对LSP隧道的检测信息	**display mpls lsp bfd** [**ipv4** *destination-address mask-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示NHLFE表的信息	**display mpls nhlfe** [*token*] [**verbose**] [**slot** *slot-number*] [**include** *text* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]
显示NHLFE表项的使用情况	**display mpls nhlfe reflist** *token* [**slot** *slot-number*] [**include** *text* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]
显示静态LSP信息	**display mpls static-lsp** [**lsp-name** *lsp-name*] [{ **exclude** | **include** } *dest-addr mask-length*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示路由的LSP相关信息	**display mpls route-state** [**vpn-instance** *vpn-instance-name*] [*dest-addr mask-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
根据LSP显示MPLS统计信息	**display mpls statistics lsp** { *index* | **all** | **name** *lsp-name* } [**|** { **begin** | **exclude** | **include** } *regular-expression*]
根据使能MPLS能力的接口显示MPLS统计信息	**display mpls statistics interface** { *interface-type interface-number* | **all** } [**|** { **begin** | **exclude** | **include** } *regular-expression*]
根据LSP的入标签显示LSP统计信息	**display mpls statistics lsp** [**in-label** *in-label*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]

1.11.2 显示MPLS LDP运行状态

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 MPLS LDP 的运行情况，用户可以通过查看显示信息验证配置的效果。

表1-27 显示 MPLS LDP 运行状态

操作	命令
显示LDP信息	**display mpls ldp** [**all** [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]]
显示指定FEC的标签通告信息	**display mpls ldp fec** [**vpn-instance** *vpn-instance-name*] *dest-addr mask-length* [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示使能了LDP能力接口的LDP相关信息	**display mpls ldp interface** [**all** [**verbose**] | [**vpn-instance** *vpn-instance-name*] [*interface-type interface-number* | **verbose**]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示对等体的信息	**display mpls ldp peer** [**all** [**verbose**] | [**vpn-instance** *vpn-instance-name*] [*peer-id* | **verbose**]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示远端对等体信息	**display mpls ldp remote-peer** [**remote-name** *remote-peer-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示对等体间会话信息	**display mpls ldp session** [**all** [**verbose**] | [**vpn-instance** *vpn-instance-name*] [*peer-id* | **verbose**]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示所有LDP会话的统计信息	**display mpls ldp session all statistics** [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示LDP创建的LSP相关信息	**display mpls ldp lsp** [**all** | [**vpn-instance** *vpn-instance-name*] [*dest-addr mask-length*]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]

说明

以上显示命令中的 **vpn-instance** *vpn-instance-name* 参数用来显示指定 LDP 实例的信息。关于 LDP 实例的详细介绍请参见“MPLS 配置指导”中的“MPLS L3VPN”。

1.11.3 清除MPLS相关信息

在完成上述配置后，在用户视图下执行 **reset** 命令可以清除 MPLS 的相关信息。

表1-28 清除 MPLS 相关信息

操作	命令
清除MPLS接口统计信息	**reset mpls statistics interface** { *interface-type interface-number* | **all** }
清除LSP统计信息	**reset mpls statistics lsp** { *index* | **all** | **name** *lsp-name* }

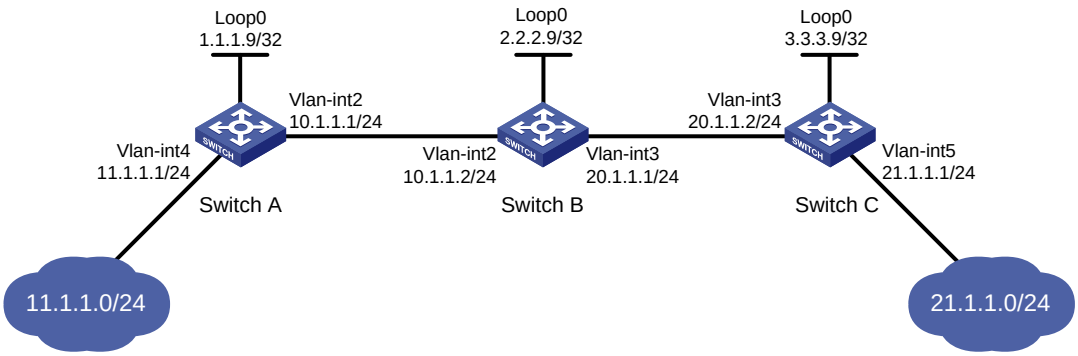
1.12 MPLS配置举例

1.12.1 配置静态建立LSP

1. 组网需求

- Switch A、Switch B 和 Switch C 均支持 MPLS。
- 在 Switch A 和 Switch C 之间建立静态 LSP，使 11.1.1.0/24 和 21.1.1.0/24 这两个网段中互访的报文能够通过 MPLS 进行传输。
- 检测 Switch A 和 Switch C 之间静态 LSP 的可达性。

图1-13 静态建立 LSP 组网图



2. 配置思路

- 手工指定静态 LSP 的标签时，需要遵循以下原则：一条 LSP 上，上游 LSR 出标签的值与下游 LSR 入标签的值相同。
- LSP 是一条单向路径，因此需要在数据传输的两个方向上分别配置一条静态 LSP。
- 配置静态 LSP 时，只要求在 Ingress 节点上存在到达 FEC 目的地址的路由，Transit 节点和 Egress 节点上不需要存在到达 FEC 目的地址的路由。因此，无需配置路由协议保证交换机之间路由可达，只需在 Ingress 节点上配置到达 FEC 目的地址的静态路由即可。

3. 配置步骤

(1) 配置各接口的 IP 地址

按照 图 1-13 配置各接口的IP地址和掩码，包括Loopback接口，具体配置过程略。

(2) 在 Ingress 上配置到达 FEC 目的地址的静态路由

在 Switch A 上配置到达 21.1.1.0/24 网段的静态路由。

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

在 Switch C 上配置到达 11.1.1.0/24 网段的静态路由。

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

(3) 使能 MPLS 功能

配置 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
```

配置 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
```

配置 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) 创建从 Switch A 到 Switch C 的静态 LSP

配置 Ingress Switch A。

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

配置 Transit Switch B

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

配置 Egress Switch C。

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

(5) 创建从 Switch C 到 Switch A 的静态 LSP

配置 Ingress Switch C。

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

配置 Transit Switch B。

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

配置 Egress Switch A。

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

(6) 检查配置结果

配置完成后，可以在各交换机上通过**display mpls static-lsp**命令查看静态 LSP 的信息。以 Switch A 的显示信息为例：

```
[SwitchA] display mpls static-lsp
total statics-lsp : 2
Name            FEC               I/O Label  I/O If                      State
AtoC            21.1.1.0/24       NULL/30    -/Vlan2                     Up
```

```
CtoA            -/-                70/NULL    Vlan2/-                     Up
```

在 Switch A 上检测 Switch A 到 Switch C 静态 LSP 的可达性。

```
[SwitchA] ping lsp -a 11.1.1.1 ipv4 21.1.1.0 24
 LSP Ping FEC: IPV4 PREFIX 21.1.1.0/24 : 100  data bytes, press CTRL_C to break
 Reply from 20.1.1.2: bytes=100 Sequence=1 time = 2 ms
 Reply from 20.1.1.2: bytes=100 Sequence=2 time = 2 ms
 Reply from 20.1.1.2: bytes=100 Sequence=3 time = 1 ms
 Reply from 20.1.1.2: bytes=100 Sequence=4 time = 2 ms
 Reply from 20.1.1.2: bytes=100 Sequence=5 time = 2 ms

 --- FEC: IPV4 PREFIX 21.1.1.0/24 ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 1/1/2 ms
```

在 Switch C 上检测 Switch C 到 Switch A 静态 LSP 的可达性。

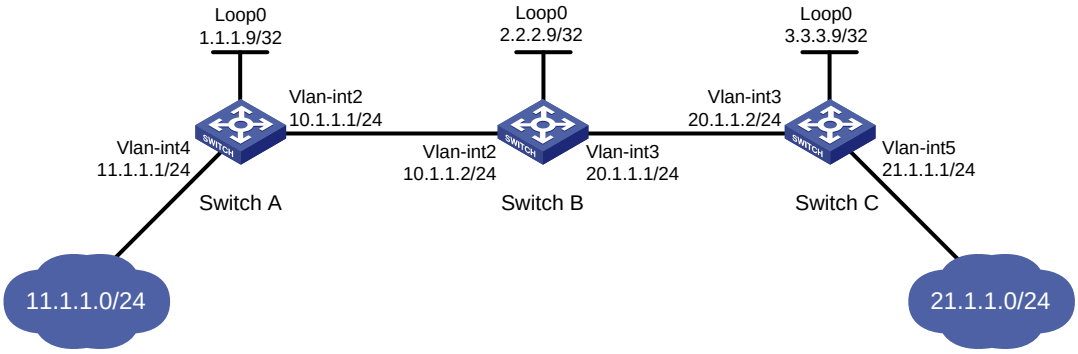
```
[SwitchC] ping lsp -a 21.1.1.1 ipv4 11.1.1.0 24
 LSP Ping FEC: IPV4 PREFIX 11.1.1.0/24 : 100  data bytes, press CTRL_C to break
 Reply from 10.1.1.1: bytes=100 Sequence=1 time = 3 ms
 Reply from 10.1.1.1: bytes=100 Sequence=2 time = 2 ms
 Reply from 10.1.1.1: bytes=100 Sequence=3 time = 2 ms
 Reply from 10.1.1.1: bytes=100 Sequence=4 time = 2 ms
 Reply from 10.1.1.1: bytes=100 Sequence=5 time = 2 ms

 --- FEC: IPV4 PREFIX 11.1.1.0/24 ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 2/2/3 ms
```

1.12.2 配置利用LDP动态建立LSP

1. 组网需求

- Switch A、Switch B 和 Switch C 均支持 MPLS。
- 在 Switch A 和 Switch C 之间使用 LDP 动态建立 LSP，使 11.1.1.0/24 和 21.1.1.0/24 这两个网段中互访的报文能够通过 MPLS 进行传输。
- 检测 Switch A 和 Switch C 之间 LSP 的可达性。

图1-14 配置动态建立 LSP 组网图



2. 配置思路

- 利用 LDP 动态建立 LSP 时，不需要手工为 LSP 指定标签，只要在 LSR 上启动 LDP 协议，即可动态分配标签并建立 LSP。
- LDP 根据路由信息动态分配标签，因此，利用 LDP 动态建立 LSP 时，需要配置路由协议，使得各交换机之间路由可达。本例中，采用的路由协议为 OSPF。

3. 配置步骤

(1) 配置各接口的 IP 地址

按照 图 1-14 配置各接口IP地址和掩码，包括VLAN接口和Loopback接口，具体配置过程略。

(2) 配置 OSPF，以保证各交换机之间路由可达

配置 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] network 11.1.1.0 0.0.0.255
[SwitchA-ospf-1-area-0.0.0.0] quit
[SwitchA-ospf-1] quit
```

配置 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
```

配置 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] network 21.1.1.0 0.0.0.255
[SwitchC-ospf-1-area-0.0.0.0] quit
[SwitchC-ospf-1] quit
```

配置完成后，在各交换机上执行 **display ip routing-table** 命令，可以看到相互之间都学到了到对方的主机路由。以 Switch A 为例：

```
[SwitchA] 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          OSPF   10   1            10.1.1.2        Vlan2
3.3.3.9/32          OSPF   10   2            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
11.1.1.0/24         Direct 0    0            11.1.1.1        Vlan4
11.1.1.1/32         Direct 0    0            127.0.0.1       InLoop0
20.1.1.0/24         OSPF   10   2            10.1.1.2        Vlan2
21.1.1.0/24         OSPF   10   3            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) 使能 MPLS 和 MPLS LDP 功能

配置 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 2
[SwitchA-Vlan-interface2] mpls
[SwitchA-Vlan-interface2] mpls ldp
[SwitchA-Vlan-interface2] quit
```

配置 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 2
[SwitchB-Vlan-interface2] mpls
[SwitchB-Vlan-interface2] mpls ldp
[SwitchB-Vlan-interface2] quit
```

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

配置 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 3
[SwitchC-Vlan-interface3] mpls
[SwitchC-Vlan-interface3] mpls ldp
[SwitchC-Vlan-interface3] quit
```

完成上述配置后，Switch A 和 Switch B、Switch B 和 Switch C 之间的本地 LDP 会话建立成功。在各交换机上执行 **display mpls ldp session** 命令，可以看到 LDP 会话的建立情况；执行 **display mpls ldp peer** 命令，可以看到 LDP 的对等体情况。以 Switch A 为例：

```
[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-interface2
 ------------------------------------------------------------------
```

(4) 配置 LSP 的触发建立策略为所有静态路由和 IGP 路由项都触发 LDP 建立 LSP

配置 Switch A。

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

配置 Switch B。

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

配置 Switch C。

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

(5) 检查配置结果

配置完成后，在各交换机上执行 **display mpls ldp lsp** 命令，可以看到 LDP LSP 的建立情况。以 Switch A 为例。

```
<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      -------/Vlan2
 3    3.3.3.9/32         NULL/1024    10.1.1.2      -------/Vlan2
 4    11.1.1.0/24        3/NULL       0.0.0.0       -------/Vlan4
 5    20.1.1.0/24        NULL/3       10.1.1.2      -------/Vlan2
 6    21.1.1.0/24        NULL/1027    10.1.1.2      -------/Vlan2
------------------------------------------------------------------------
 A '*' before an LSP means the LSP is not established
 A '*' before a Label means the USCB or DSCB is stale
```

在 Switch A 上检测到达 21.1.1.0/24 网段的 LDP LSP 的可达性。

```
[SwitchA] ping lsp ipv4 21.1.1.0 24
 LSP Ping FEC: IPV4 PREFIX 21.1.1.0/24 : 100  data bytes, press CTRL_C to break
 Reply from 20.1.1.2: bytes=100 Sequence=1 time = 3 ms
 Reply from 20.1.1.2: bytes=100 Sequence=2 time = 2 ms
 Reply from 20.1.1.2: bytes=100 Sequence=3 time = 1 ms
 Reply from 20.1.1.2: bytes=100 Sequence=4 time = 1 ms
 Reply from 20.1.1.2: bytes=100 Sequence=5 time = 3 ms

 --- FEC: IPV4 PREFIX 21.1.1.0/24 ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 1/2/3 ms
```

在 Switch C 上检测到达 11.1.1.0/24 网段的 LDP LSP 的可达性。

```
[SwitchC] ping lsp ipv4 11.1.1.0 24
 LSP Ping FEC: IPV4 PREFIX 11.1.1.0/24 : 100  data bytes, press CTRL_C to break
 Reply from 10.1.1.1: bytes=100 Sequence=1 time = 2 ms
 Reply from 10.1.1.1: bytes=100 Sequence=2 time = 2 ms
 Reply from 10.1.1.1: bytes=100 Sequence=3 time = 2 ms
 Reply from 10.1.1.1: bytes=100 Sequence=4 time = 3 ms
 Reply from 10.1.1.1: bytes=100 Sequence=5 time = 2 ms

 --- FEC: IPV4 PREFIX 11.1.1.0/24 ping statistics ---
 5 packet(s) transmitted
 5 packet(s) received
 0.00% packet loss
 round-trip min/avg/max = 2/2/3 ms
```

1.12.3 配置利用BFD检测LSP

1. 组网需求

在 图 1-14 的网络中，利用LDP建立 11.1.1.0/24 到 21.1.1.0/24、21.1.1.0/24 到 11.1.1.0/24 两条LSP后，使用BFD检测LSP隧道的连通性。

2. 组网图

参见 图 1-14。

3. 配置步骤

(1) 配置LDP会话，具体配置过程请参见“1.12.2 配置利用LDP动态建立LSP”

(2) 配置 BFD 检测 LSP

配置 Switch A。

```
<SwitchA> system-view
[SwitchA] mpls lspv
[SwitchA -mpls-lspv] bfd enable 21.1.1.0 24
[SwitchA -mpls-lspv] quit
```

配置 Switch C。

```
<SwitchC> system-view
[SwitchC] mpls lspv
[SwitchC-mpls-lspv] bfd enable 11.1.1.0 24
[SwitchC-mpls-lspv] quit
```

(3) 检查配置结果

配置完成后，在设备 SwitchA 和 SwitchC 上执行 **display mpls lsp bfd** 命令，可以看到 BFD 检测 LSP 的 BFD 会话的建立情况。以 Switch A 为例。

```
[SwitchA] display mpls lsp bfd

              MPLS BFD Session(s) Information
 ------------------------------------------------------------------------------
 FEC           : 11.1.1.0/24        Type         : LSP
 Local Discr   : 130                Remote Discr : 130
 Tunnel ID     : ---                NextHop      : ---
 Session State : Up                 Source IP    : 3.3.3.9
 Session Role  : Passive

 FEC           : 21.1.1.0/24        Type         : LSP
 Local Discr   : 129                Remote Discr : 129
 Tunnel ID     : 0x6040000          NextHop      : 10.1.1.2
 Session State : Up                 Source IP    : 1.1.1.9
 Session Role  : Active

 Total Session Num: 2
```

以上显示信息表示，Switch A 和 Switch C 之间建立了两个 BFD 会话，分别用来检测 11.1.1.0/24 到 21.1.1.0/24、21.1.1.0/24 到 11.1.1.0/24 两条 LSP。

```
[SwitchA] display bfd session verbose
```

```
Total session number: 2   Up session number: 2   Init mode: Active

IPv4 session working under Ctrl mode:

    Local Discr: 129                Remote Discr: 129
      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: 00:15:52              Auth mode: None
   Connect Type: Indirect              Board Num: 6
       Protocol: MFW/LSPV
      Diag Info: No Diagnostic

    Local Discr: 130                Remote Discr: 130
      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: 00:15:44              Auth mode: None
   Connect Type: Indirect              Board Num: 7
       Protocol: MFW/LSPV
      Diag Info: No Diagnostic
```

通过以上显示信息可以查看 BFD 会话的详细信息。

目 录

1 MPLS TE配置 ····· 1-1
1.1 MPLS TE简介 ····· 1-1
1.1.1 流量工程与MPLS TE ····· 1-1
1.1.2 MPLS TE的基本概念 ····· 1-2
1.1.3 MPLS TE的实现 ····· 1-2
1.1.4 CR-LSP ····· 1-3
1.1.5 RSVP-TE ····· 1-4
1.1.6 流量转发 ····· 1-8
1.1.7 CR-LSP备份 ····· 1-9
1.1.8 快速重路由 ····· 1-10
1.1.9 MPLS TE隧道的保护倒换 ····· 1-11
1.1.10 协议规范 ····· 1-12
1.2 MPLS TE配置任务简介 ····· 1-12
1.3 配置MPLS TE基本能力 ····· 1-13
1.3.1 配置准备 ····· 1-13
1.3.2 配置MPLS TE基本能力 ····· 1-13
1.4 使用静态CR-LSP配置MPLS TE隧道 ····· 1-14
1.4.1 配置准备 ····· 1-14
1.4.2 使用静态CR-LSP配置MPLS TE隧道 ····· 1-14
1.5 使用动态信令协议配置MPLS TE隧道 ····· 1-15
1.5.1 配置准备 ····· 1-15
1.5.2 使用动态信令协议配置MPLS TE隧道 ····· 1-16
1.6 配置RSVP-TE高级特性 ····· 1-20
1.6.1 配置准备 ····· 1-20
1.6.2 配置RSVP-TE高级特性 ····· 1-20
1.7 调整CR-LSP的建立 ····· 1-24
1.7.1 配置准备 ····· 1-24
1.7.2 调整CR-LSP的建立 ····· 1-24
1.8 调整MPLS TE隧道的建立 ····· 1-26
1.8.1 配置准备 ····· 1-26
1.8.2 调整MPLS TE隧道的建立 ····· 1-26
1.9 配置流量转发 ····· 1-28
1.9.1 配置准备 ····· 1-28

1.9.2 配置流量转发 ····· 1-28
1.10 配置影响流量转发的参数 ····· 1-30
1.10.1 配置准备 ····· 1-30
1.10.2 配置影响流量转发的参数 ····· 1-30
1.11 配置CR-LSP备份 ····· 1-31
1.11.1 配置准备 ····· 1-31
1.11.2 配置CR-LSP备份 ····· 1-31
1.12 配置MPLS TE快速重路由 ····· 1-32
1.12.1 配置准备 ····· 1-32
1.12.2 配置MPLS TE快速重路由 ····· 1-33
1.13 检测MPLS TE隧道 ····· 1-34
1.13.1 MPLS LSP Ping功能 ····· 1-35
1.13.2 MPLS LSP Tracert功能 ····· 1-35
1.13.3 配置BFD检测MPLS TE隧道功能 ····· 1-35
1.13.4 配置MPLS TE隧道的周期性LSP Tracert功能 ····· 1-37
1.14 配置保护倒换功能 ····· 1-38
1.14.1 配置准备 ····· 1-38
1.14.2 配置过程 ····· 1-38
1.15 MPLS TE显示和维护 ····· 1-38
1.15.1 MPLS TE显示和维护 ····· 1-38
1.16 MPLS TE典型组网举例 ····· 1-40
1.16.1 使用静态CR-LSP配置MPLS TE隧道示例 ····· 1-40
1.16.2 使用RSVP-TE配置MPLS TE隧道示例 ····· 1-45
1.16.3 配置RSVP-TE GR示例 ····· 1-51
1.16.4 配置MPLS RSVP-TE与BFD联动 ····· 1-53
1.16.5 配置CR-LSP备份示例 ····· 1-55
1.16.6 配置快速重路由示例 ····· 1-58
1.16.7 配置MPLS L3VPN中的MPLS TE示例 ····· 1-67
1.17 MPLS TE常见配置错误举例 ····· 1-75
1.17.1 不能产生TE LSA ····· 1-75

1 MPLS TE配置

说明

本系列交换机中 S5500-28SC-HI 和 S5500-52SC-HI 不支持 MPLS TE 功能。

1.1 MPLS TE简介

1.1.1 流量工程与MPLS TE

1. 流量工程

(1) 流量工程的作用

网络拥塞是影响骨干网络性能的主要问题。拥塞的原因可能是网络资源不足，也可能网络资源负载不均衡导致的局部拥塞。TE（Traffic Engineering，流量工程）解决的是由于负载不均衡导致的拥塞。

流量工程通过实时监控网络的流量和网络单元的负载，动态调整流量管理参数、路由参数和资源约束参数等，使网络运行状态迁移到理想状态，优化网络资源的使用，避免负载不均衡导致的拥塞。

总的来说，流量工程的性能指标包括两个方面：

- 面向业务的性能指标：增强业务的 QoS（Quality of Service，服务质量）性能，例如对分组丢失、时延、吞吐量以及 SLA（Service Level Agreement，服务等级协定）的影响。
- 面向资源的性能指标：优化资源利用。带宽是一种重要的资源，对带宽资源进行高效管理是流量工程的一项中心任务。

(2) 流量工程的解决方案

现有的 IGP 协议都是拓扑驱动的，只考虑网络的连接情况，不能灵活反映带宽和流量特性这类动态状况。

解决 IGP 上述缺点的方法之一是使用重叠模型（Overlay），如 IP over ATM、IP over FR 等。重叠模型在网络的物理拓扑结构之上提供了一个虚拟拓扑结构，从而扩展了网络设计的空间，为支持流量与资源控制提供了许多重要功能，可以实现多种流量工程策略。然而，由于协议之间往往存在很大差异，重叠模型在可扩展性方面存在不足。

为了在大型骨干网络中部署流量工程，必须采用一种可扩展性好、简单的解决方案。MPLS TE 就是为这一需求而提出的。

2. MPLS TE

MPLS 本身具有一些不同于 IGP 的特性，其中就有实现流量工程所需要的，例如：

- MPLS 支持显式 LSP 路由；
- LSP 较传统单个 IP 分组转发更便于管理和维护；
- 基于 MPLS 的流量工程的资源消耗较其它实现方式更低。

MPLS TE 结合了 MPLS 技术与流量工程，通过建立到达指定路径的 LSP 隧道进行资源预留，使网络流量绕开拥塞节点，达到平衡网络流量的目的。

在资源紧张的情况下，MPLS TE 能够抢占低优先级 LSP 隧道带宽资源，满足大带宽 LSP 或重要用户的需求。

同时，当 LSP 隧道故障或网络的某一节点发生拥塞时，MPLS TE 可以通过备份路径和 FRR（Fast ReRoute，快速重路由）提供保护。

使用 MPLS TE，网络管理员只需要建立一些 LSP 和旁路拥塞节点，就可以消除网络拥塞。随着 LSP 数量的增长，还可以使用专门的离线工具进行业务量分析。

1.1.2 MPLS TE的基本概念

1. LSP隧道

对于一条 LSP，一旦在 Ingress 节点给报文打上标签，流量的转发就完全由标签决定了。流量对 LSP 的中间节点是透明的，从这个意义上来说，一条 LSP 可以看作是一条 LSP 隧道。

2. MPLS TE隧道

在部署重路由（Reroute）或需要将流量通过多条路径传输时，可能需要用到多条 LSP 隧道。在 TE 中，这样的一组 LSP 隧道称为 TE 隧道（Traffic Engineered Tunnel）。

1.1.3 MPLS TE的实现

MPLS TE 主要实现两类功能：

- 静态 CR-LSP（Constraint-based Routed Label Switched Paths，基于约束路由的 LSP）的处理：创建和删除静态 CR-LSP。静态 CR-LSP 的带宽需要手工指定。
- 动态 CR-LSP 处理：包括对三种不同类型 CR-LSP 的处理：基本 CR-LSP、备份 CR-LSP 和快速重路由 CR-LSP。

静态 CR-LSP 的处理比较简单。对于动态 CR-LSP，MPLS TE 在实现上主要包括四个部分。

1. 发布含TE属性的信息

MPLS TE 需要了解每条链路的动态 TE 相关属性，这可以通过对现有的使用链路状态算法的 IGP 协议进行扩展来实现，比如 OSPF 协议和 IS-IS 协议的扩展。

扩展后的 OSPF 和 IS-IS 协议在链路连接状态中增加了链路带宽、着色等 TE 相关属性。

每台设备收集本区域或本级别所有设备每条链路的 TE 相关信息，生成 TEDB（TE DataBase，流量工程数据库）。

2. 计算路径

使用链路状态算法的路由协议通过 SPF（Shortest Path First，最短路径优先）算法计算出到达网络各个节点的最短路径。

MPLS TE 使用 CSPF（Constraint-based Shortest Path First，基于约束的最短路径优先）算法计算出到达某个节点的满足 TE 属性要求的最短路径。

CSPF 算法是从 SPF 算法衍生来的，CSPF 有两个输入条件：

- 需要建立的 LSP 的带宽、着色、建立/保持优先级、显式路径等约束条件，这些都在 LSP 的入口处配置；

- 流量工程数据库 TEDB。

CSPF 的计算过程就是针对 LSP 要求，先对 TEDB 中的链路进行剪切，把不满足 TE 属性要求的链路剪掉；再采用 SPF 算法，寻找一条到 LSP egress 节点的满足 TE 属性要求的最短路径。

3. 建立路径

支持建立 LSP 隧道的信令包括 CR-LDP 和 RSVP-TE。它们都能够携带 LSP 的带宽、部分显式路由、着色等约束参数，两者完成的功能是一样的。

从内部实现来看，CR-LDP 通过 TCP 建立 LSP，RSVP-TE 则通过 Raw IP 建立 LSP 连接。

RSVP 技术经历了多年的发展，其体系结构、协议规程与对各种业务的支持机制相对比较成熟；CR-LDP 则是新技术，在可扩展性方面优势明显。

目前设备仅支持使用 RSVP-TE 信令协议。

4. 转发报文

使用建立的隧道转发报文。

1.1.4 CR-LSP

基于一定约束条件建立的 LSP 称为 CR-LSP，与普通 LSP 不同，CR-LSP 的建立不仅依赖路由信息，还需要满足其他一些条件，比如指定的带宽、选定的路径或 QoS 参数。

建立和管理约束条件的机制称为 CR（Constraint-based Routing，基于约束的路由）。

下面对 CR 的主要内容进行简单介绍。

1. 严格显式路由与松散显式路由

- 如果约束信息是对沿途 LSR 的精确指定，建立的 LSP 称为严格的显式路由（Strict Explicit Route）；
- 如果约束信息是对选择下游 LSR 时的模糊限制，建立的 LSP 称为松散的显式路由（Loose Explicit Route）。

2. 流量参数

路径的流量参数有三个：峰值速率（peak rate）和承诺速率（committed rate），描述路径本身对带宽的约束；另外一个是服务粒度（service granularity）。

3. 抢占

如果在建立 CR-LSP 的过程中，无法找到满足所需带宽要求的路径，一种解决方法是拆除另外一条已经建立的路径，占用为它分配的带宽资源，这种处理方式称为抢占（Preemption）。

CR-LSP 使用两个优先级属性来决定是否可以进行抢占：建立优先级（Setup Priority）和保持优先级（Holding Priority）。建立优先级和保持优先级的取值范围都是 0～7，数值越小则优先级越高。

抢占由 RSVP-TE 的 Resv 消息发起。当新建一条路径 Path1 时，如果需要与已建立的路径 Path2 争夺资源，只有当 Path1 的建立优先级高于 Path2 的保持优先级时，Path1 才能抢占成功。

因此，为保证 CR-LSP 能够正确建立，建立优先级不能高于保持优先级，否则可能会导致 LSP 间无穷尽的互相抢占，造成振荡。

4. 路由固定（Route Pinning）

CR-LSP 创建成功后，不随路由变化而变化的特性叫做路由固定。

当某个网络未运行 IGP TE 时，网络管理员不能确定网络上的哪些地方可以获得带宽，这时需要选择具有所需带宽的松散 ER-hop（Explicit Route）来创建 CR-LSP，但这些 CR-LSP 将会随路由变化而变化。当路由变化时，比如出现了一个更好的下一跳，已建立的 CR-LSP 也将会随之改变。

如果不希望使用松散路由建立的 CR-LSP 随路由变化而改变，网络管理员可以在 CR-LSP 创建成功时把这些 CR-LSP 配置成永久性的，不随路由变化而变化。

5. 管理组和亲和属性

MPLS TE 隧道的亲和属性决定隧道使用的链路属性，亲和属性与链路管理组配合，确定隧道可以使用哪些链路。

6. 重优化

流量工程是系统规划网络资源使用的过程。根据用户需求可以配置流量工程，提供要求的 QoS。

服务提供商通常利用一定的机制去优化 CR-LSP，以优化网络资源使用。一种方法是人工配置，但是需要服务提供商进行测量和对 CR-LSP 微调。使用 MPLS TE 则能够动态优化 CR-LSP，从而节省人力。

动态优化 CR-LSP 即定期重计算 CR-LSP 穿越的路由。如果重计算的路由优于当前路由，则创建一条新的 CR-LSP，为之分配新路由，并将业务从旧的 CR-LSP 切换至新的 CR-LSP，删除旧 CR-LSP。

1.1.5 RSVP-TE

1. RSVP-TE概述

现在使用两种 QoS 体系：IntServ（Integrated Service，综合服务）模型和 DiffServ（Differentiated Service，区分业务）模型。

RSVP（Resource Reservation Protocol，资源预留协议）是为 IntServ（Integrated Service，综合业务模型）而设计的，用于在一条路径的各节点上进行资源预留。RSVP 工作在传输层，但不参与应用数据的传送，是一种 Internet 上的控制协议，类似于 ICMP。

简单来说，RSVP 具有以下几个主要特点：

- 单向；
- 面向接收者，由接收者发起对资源预留的请求，并维护资源预留信息；
- 使用“软状态”（soft state）机制维护资源预留信息。

RSVP 经扩展后可以支持 MPLS 标签的分发，并在传送标签绑定消息的同时携带资源预留信息，这种扩展后的 RSVP 称为 RSVP-TE，作为一种信令协议用于在 MPLS TE 中建立 LSP 隧道。

2. RSVP-TE基本概念

(1) 软状态

“软状态”是指在 RSVP-TE 中，通过消息的定时刷新来维持节点上的资源预留状态。

资源预留状态包括由 Path 消息创建的路径状态（path state）和由 Resv 消息创建的预留状态（reservation state）。这两种状态分别由 Path 消息和 Resv 消息定时刷新。对于某个状态，如果连续没有收到刷新消息，这个状态将被删除。

(2) 资源预留类型

使用 RSVP-TE 建立的 LSP 都具有某种资源预留类型（reservation style），在建立 RSVP 会话时，由接收者决定此会话使用哪种预留类型，从而决定可以使用哪些 LSP。

目前设备支持以下两种预留类型：

- FF（Fixed-Filter style）：固定过滤器类型。为每个发送者单独预留资源，不能与同一会话中其他发送者共享资源。
- SE（Shared-Explicit style）：共享显式类型。为同一个会话的发送者建立一个预留，可以共享资源。

说明

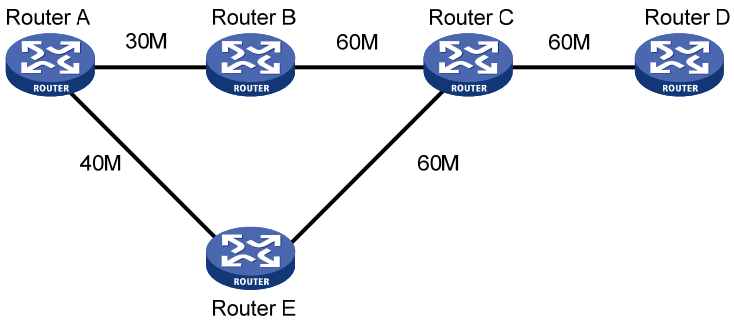
由于目前同一会话不能同时存在多条 LSP，SE 资源预留方式主要用于中断前建立（make-before-break）。

---

3. make-before-break

make-before-break 是指一种可以在尽可能不丢失数据，也不占用额外带宽的前提下改变 MPLS TE 隧道属性的机制。

图1-1 make-before-break 示意图



在 图 1-1 中，假设需要建立一条Router A到Router D的路径，保留 30M带宽，开始建立的路径是 Router A→Router B→Router C→Router D。

现在希望将带宽增大为 40M，Router A→Router B→Router C→Router D 路径不能满足要求。而如果选择 Router A→Router E→Router C→Router D，则 Router C→Router D 也存在带宽不够的问题。

采用 make-before-break 机制，新建立的路径在 Router C→Router D 可以共享原路径的带宽，新路径建立成功后，流量转到新路径上，之后拆除原路径，从而有效地避免了流量中断。

4. RSVP-TE消息类型

RSVP-TE 使用 RSVP 的消息类型，并进行了扩展。RSVP 使用以下消息类型：

- Path 消息：由发送者沿数据报文传输的方向向下游发送，在沿途所有节点上保存路径状态（path state）。
- Resv 消息：由接收者沿数据报文传输的方向逆向发送，在沿途所有节点上进行资源预留，并创建和维护预留状态（reservation state）。
- PathTear 消息：此消息产生后马上向下游发送，并立即删除沿途节点的路径状态和相关的预留状态。
- ResvTear 消息：此消息产生后马上向上游发送，并立即删除沿途节点的预留状态。

- PathErr 消息：如果在处理 Path 消息的过程中发生了错误，就会向上游发送 PathErr 消息，PathErr 消息不影响沿途节点的状态，只是把错误报告给发送者。
- ResvErr 消息：如果在处理 Resv 消息的过程中发生了错误，或者由于抢占导致预留被破坏，就会向下游节点发送 ResvErr 消息。
- ResvConf 消息：该消息发往接收者，用于对预留消息进行确认。
- Hello 消息：在两个直连的 RSVP 邻居之间建立和维持链路局部的邻居关系。

RSVP 的 TE 扩展主要是在其 Path 消息和 Resv 消息中增加新的对象，新增对象除了可以携带标签绑定信息外，还可以携带对 LSR 在沿途寻找路径时的限制信息，从而支持 CR-LSP 的功能，并支持 FRR。

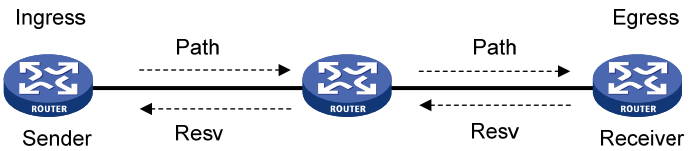
- Path 消息新增的对象包括：LABEL_REQUEST、EXPLICIT_ROUTE、RECORD_ROUTE 和 SESSION_ATTRIBUTE。
- Resv 消息新增的对象包括：LABEL 和 RECORD_ROUTE。

LABEL_REQUEST 对象包含在 Path 消息中，为 LSP 请求标签绑定，该对象也保存在路径状态块 PSB（Path State Block）中。接收到该对象的节点将分配的标签通过 Resv 消息中的 LABEL 对象通知上游节点，从而完成标签的发布和传递。

5. 建立LSP隧道

图 1-2 是使用RSVP建立LSP隧道的示意图。

图1-2 建立 LSP 隧道



使用 RSVP 建立 LSP 隧道的过程可以简单描述为：

(1) Ingress LSR 产生携带标签请求信息的 Path 消息，沿着通过 CSPF 计算出的路径逐跳发送给 Egress LSR；

(2) Egress LSR 收到 Path 消息后，产生携带预留信息和标签的 Resv 消息，沿着 Path 消息发送的相反路径逐跳返回 Ingress LSR，同时，Resv 消息在沿途的 LSR 上进行资源预留；

(3) 当 Ingress LSR 收到 Resv 消息时，LSP 建立成功。

采用 RSVP-TE 建立的 LSP 具有资源预留功能，沿途的 LSR 可以为该 LSP 分配一定的资源，使在此 LSP 上传送的业务得到保证。

6. RSVP刷新机制

RSVP 通过 Refresh 消息来维护路径和预留状态，Refresh 消息不仅用于在 RSVP 邻居节点进行状态同步，也用于恢复丢失的 RSVP 消息。

Refresh 消息并不是一种新的消息，它是以前发布过的消息的再次传送，Refresh 消息中携带的主要信息和传送时使用的路径都与它要刷新的消息完全一致。只有 Path 消息和 Resv 消息才可能是 Refresh 消息。

由于 Refresh 消息是定时发送的，当网络中的 RSVP 会话比较多时，Refresh 消息会加重网络负载；而对于时延敏感的应用，当消息丢失时，等待通过 Refresh 消息恢复的时间可能无法接受。简单地调整刷新间隔并不能同时解决这两类问题。

RFC 2961（RSVP Refresh Overhead Reduction Extensions）定义了几种新的扩展机制，用于解决 Refresh 消息带来的上述问题。

(1) Message_ID 扩展

RSVP 本身使用 Raw IP 发送消息，RFC 2961 中定义的 Message_ID 扩展机制增加了可以在 RSVP 消息中携带的对象，其中，Message_ID 和 Message_ID_ACK 对象用于 RSVP 消息确认，从而提高 RSVP 消息发送的可靠性。

在接口使能 Message_ID 机制后，可以配置重传功能，设定 RSVP 消息的重传参数。重传功能是指：节点发送了携带 Message_ID 对象的消息，且 Message_ID 对象的 ACK_Desired 标识（是否需要应答标识）置位后，如果在重传时间 Rf 内没有收到携带对应 Message_ID_ACK 对象的消息，则重传时间 Rf 超时后重传此消息，并将重传时间置为（1＋Delta）×Rf。节点持续按照上述方法重传此消息，直到节点在重传时间超时前接收到对应的应答消息，或消息传送次数达到允许的最大值。

(2) 摘要刷新扩展

摘要刷新 Srefresh（Summary Refresh）可以不传送标准的 Path 或 Resv 消息，而仍能实现对 RSVP 的状态刷新，从而可以减少网络上的 Refresh 消息流量，并加快节点对这类消息的处理速度。

摘要刷新扩展需要与 Message_ID 扩展配合使用。只有那些已经被包含 Message_ID 对象的 Path 和 Resv 消息发布过的状态才能使用摘要刷新扩展机制刷新。

7. PSB、RSB与BSB的超时

为建立 LSP，发送者在 Path 消息中携带 LABEL_REQUEST 对象，接收者收到带有 LABEL_REQUEST 对象的 Path 消息后，就会分配一个标签，并将标签放在 Resv 消息的 LABEL 对象中。

LABEL_REQUEST 对象保存在上游节点的 PSB（Path State Block，路径状态块）中，LABEL 对象则保存在下游节点的 RSB（Reservation State Block，预留状态块）中。当连续未收到刷新消息的次数超过 PSB 或 RSB 的超时倍数（当达到此数值时即为超时）时，PSB 或 RSB 中相应的状态将被删除。

假设有一个资源预留请求，在某些节点上没有通过准入控制，有时可能不希望立即删除这个请求的状态，但这个请求也不应该阻止其他请求使用它预留的资源。这种情况下，节点将进入阻塞状态（Blockade State），在下游节点生成 BSB（Blockade State Block，阻塞状态块）。当连续未收到刷新消息的次数超过阻塞状态超时倍数时，BSB 中相应的状态被删除。

8. RSVP-TE GR

RSVP-TE GR（Graceful Restart，平滑重启）功能是指在信令协议或控制平面出现异常时，保留软状态信息和转发表项信息，以保证数据转发不中断。

参与 RSVP-TE GR 过程的设备分为以下两种：

- GR Restarter：GR 重启路由器，指由管理员手工或设备故障触发而重启协议的设备，它必须具备 GR 能力。
- GR Helper：GR Restarter 的邻居，与重启的 GR Restarter 保持邻居关系，并协助其恢复重启前的转发状态，它也必须具备 GR 能力。

RSVP-TE GR 依赖于 RSVP-TE 的 Hello 扩展能力，通过扩展的 RSVP Hello 报文向邻居通告自己的 GR 能力和相关时间参数。设备和邻居如果都具备 RSVP GR 能力，那么在完成 GR 参数的交互后，就可以在检测到对方发生 GR 重启时，充当对方的 GR Helper，保证在 GR Restarter 重启的过程中，数据转发不会中断。

当 GR Restarter 发生重启时，GR Helper 连续丢失的 Hello 报文次数超过了配置的值，由此判定 GR Restarter 发生了重启。此时 GR Helper 会保留与该邻居相关的软状态信息，并保持向对方周期性发送 Hello 报文，直到重启定时器（Restart Timer）超时。

在重启定时器超时前，如果 GR Helper 邻居和 GR Restarter 重新建立了 Hello 会话协商，那么启动恢复定时器，并触发信令报文交互以恢复原有的软状态；否则，将删除与该邻居相关的所有 RSVP 软状态信息和转发表项。如果恢复定时器超时，则删除那些在 GR 恢复过程中没有恢复的软状态和表项信息。

说明

本系列交换机在使能 RSVP-TE GR 能力之后，可以同时充当 GR Restarter 和 GR Helper;

1.1.6 流量转发

当 MPLS TE 隧道建立之后，如果不配置流量沿隧道转发，缺省的情况下依然会沿 IP 路由转发。

配置流量沿隧道转发有如下三种方法：

1. 静态路由

使用静态路由转发流量，是最简便的方法，因为 Tunnel 的接口地址通常情况下不会发布到 IGP 中。这时候通过定义一条通过 Tunnel 接口到达目的网络地址的静态路由，就把流量引入到 MPLS TE 隧道上进行转发。

说明

有关静态路由的介绍请参见“三层技术-IP 路由配置指导”中的“静态路由”。

2. 策略路由

使用基于策略的路由（Policy-based routing，PBR），通过 Tunnel 接口的流量需要通过 ACL 定义策略，如果匹配该流量，将下一跳的接口指向 Tunnel，在流量的入接口应用策略路由，就把流量引入到 MPLS TE 隧道上进行转发。

说明

有关策略路由的介绍请参见“三层技术-IP 路由配置指导”中的“IP 单播策略路由”。

3. 自动路由发布

自动路由发布会将 Tunnel 的接口发布到 IGP 路由中，这样流量都会通过 MPLS TE 隧道转发。

自动路由发布包括两种：IGP Shortcut 与转发邻接。

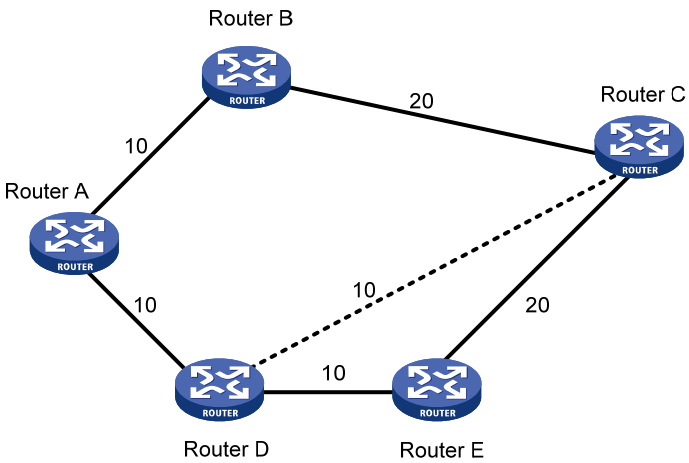
OSPF 和 IS-IS 支持 IGP Shortcut 和转发邻接特性，可以使用 TE Tunnel 作为出接口。在这种应用中，TE Tunnel 被看做点到点链路。

IGP Shortcut 特性也称为自动路由宣告（AutoRoute Announce），该特性将 TE Tunnel 看作直接与目的地址相连的逻辑接口，计算该 TE Tunnel 隧道入口设备的 IGP 路由。

IGP Shortcut 和转发邻接的区别在于：

- 在 IGP Shortcut 应用中，使能此特性的设备使用 TE Tunnel 作为出接口，但它不将这条路由发布给邻居设备，因此，其他设备不能使用此 TE Tunnel。
- 如果配置了转发邻接，则使能此特性的设备在使用 TE Tunnel 作为出接口的同时，也将这条 TE Tunnel 发布给邻居设备，因此，其他设备能够使用此 TE Tunnel。

图1-3 IGP Shortcut 与转发邻接示意图



在 图 1-3 中，Router D到Router C之间有一条TE Tunnel，IGP Shortcut只能使入节点Router D在计算IGP路由时利用这条隧道，Router A并不能利用这条隧道到达Router C。如果配置了转发邻接特性，则Router A也能够知道这条TE Tunnel的存在，从而可以利用该隧道将到Router C的流量转发到Router D上。

IGP Shortcut 和转发邻接包括 Tunnel 上的配置和 IGP 本身的配置两部分。

Tunnel 接口上的配置需要注意：

- Tunnel 接口的目的地址应该属于使能相应特性的区域内；
- Tunnel 接口的目的地址可通过区域内路由到达。

1.1.7 CR-LSP备份

CR-LSP 备份是一种端到端的路径保护（Path Protection，end-to-end protection），对整条 LSP 提供保护，而 FRR 则是一种局部保护措施，只能保护 LSP 中的某条链路和某个节点。并且，FRR 是一种快速响应的临时性保护措施，对于切换时间有严格要求，LSP 备份则没有时间要求。

同一条隧道下对主 LSP 进行路径备份的 LSP 称为备份路径。当 Ingress 感知到主 LSP 不可用时，将流量切换到备份路径上，当主 LSP 路径恢复后再将流量切换回来，以实现对主 LSP 路径的备份保护。

有两种备份方法：

- 热备份：创建主 CR-LSP 后随即创建备份 CR-LSP。主 CR-LSP 失效时，通过 MPLS TE 直接将业务切换至备份 CR-LSP。
- 普通备份：指主 CR-LSP 失效后创建备份 CR-LSP。

1.1.8 快速重路由

1. 快速重路由概述

快速重路由 FRR（Fast ReRoute），是 MPLS TE 中实现网络局部保护的技术。FRR 的切换速度可以达到 50ms，能够最大程度减少网络故障时数据的丢失。

对 LSP 配置 FRR 功能后，当 LSP 上的某条链路或某个节点失效时，流量会被切换到保护链路上，同时 LSP 头节点尝试建立新的 LSP。

2. 基本概念

下面介绍 FRR 中的几个概念：

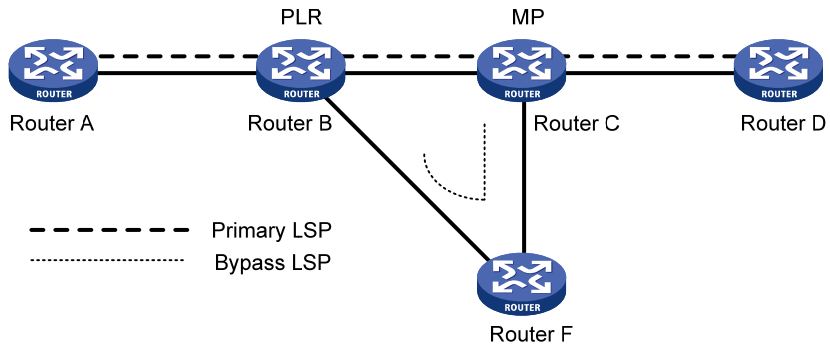
- 主 LSP：被保护的 LSP。
- Bypass LSP：旁路 LSP，保护主 LSP 的 LSP。
- PLR（Point of Local Repair）：本地修复节点。Bypass LSP 的头节点，必须在主 LSP 的路径上，并且不能是主 LSP 的尾节点。
- MP（Merge Point）：汇聚点。Bypass LSP 的尾节点，必须在主 LSP 的路径上，并且不能是主 LSP 的头节点。

3. 保护方式

根据保护的对象不同，FRR 分为两类：

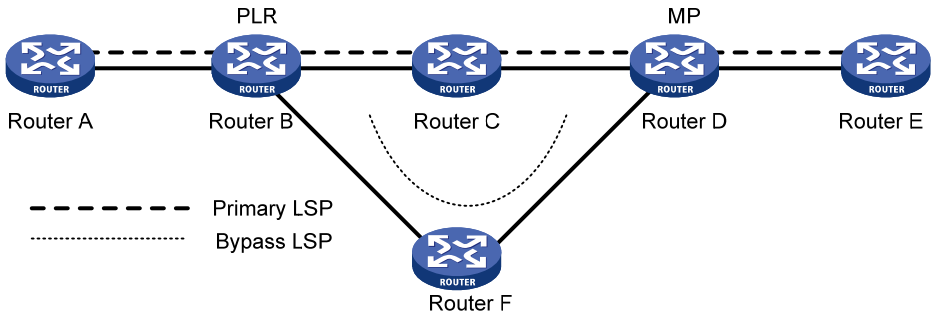
- 链路保护：PLR和MP之间有直接链路连接，主LSP经过这条链路。当这条链路失效时，流量可以切换到Bypass LSP上。如 图 1-4 所示，主LSP是Router A→Router B→Router C→Router D，Bypass LSP是Router B→Router F→Router C。

图1-4 FRR 链路保护示意图



- 节点保护：PLR和MP之间通过一台设备连接，主LSP经过这台设备。当这台设备失效时，流量可以切换到Bypass LSP上。如 图 1-5 所示，主LSP是Router A→Router B→Router C→Router D→Router E，Bypass LSP是Router B→Router F→Router D，Router C是被保护的设备。

图1-5 FRR 节点保护示意图



4. 部署快速重路由

在配置 Bypass LSP 时，应该规划好它所保护的链路或节点，并确保该 Bypass LSP 不会经过它所保护的链路或节点，否则不能真正起到保护作用。

另外，由于 Bypass 隧道需要预先建立，快速重路由会占用额外的带宽。在网络带宽余量不多的情况下，只能对关键的接口或链路进行快速重路由保护。

1.1.9 MPLS TE隧道的保护倒换

PS（Protection Switching，保护倒换）是为主 Tunnel 建立相应的保护 Tunnel（备用 Tunnel）。主 Tunnel 和保护 Tunnel 构成一个保护组。在主 Tunnel 发生缺陷时，数据流能迅速的倒换到保护 Tunnel，从而大大提高网络的可靠性。当主 Tunnel 恢复无缺陷时，还可以将数据流从保护 Tunnel 回切到主 Tunnel。

1. 保护倒换方式

设备目前实现的保护倒换为 1:1 保护倒换。在 Tunnel 的入节点和出节点之间提供主备两条 Tunnel。正常情况下，数据在主 Tunnel 上传输；当入节点通过检测机制发现主 Tunnel 发生缺陷，需要进行保护倒换时，将数据切换到保护 Tunnel 上继续传输。

2. 保护倒换触发方式

触发保护倒换的方式有如下几种：

(1) 外部倒换是指通过手工配置的命令触发的保护倒换。外部倒换的方式有以下几种：

- 清除倒换（Clear）：清除所有的外部倒换方式。
- 锁定倒换（Lockout of Protection）：数据流锁定在主 LSP 上传输。
- 强制倒换（Forced Switch）：强制数据流在备份 LSP 上传输。
- 手工倒换（Manual Switch）：手动将数据流从主/备 LSP 倒换到备/主 LSP 上传输。

(2) 信令倒换（Signal Fail）是指通过协议信令触发的保护倒换。如可通过 BFD 检测 MPLS-TE 隧道功能触发保护倒换，当 BFD 检测功能检测到隧道失效时，即触发启用信令倒换。

外部倒换方式及信令倒换方式优先级从高到低依次为：

- 清除倒换
- 锁定倒换
- 强制倒换
- 信令倒换

- 手工倒换

在实际操作中，只有在手工输入的外部倒换方式比当前启用的倒换方式优先级高的情况下，外部倒换才会生效。设备上已经配置了外部倒换方式时，若要将其修改为低优先级的外部倒换方式，则需要先配置清除倒换（Clear）方式，再配置低优先级的外部倒换方式。

3. 路径切换方式

保护组内主备 Tunnel 的路径切换方式为单向路径切换，即进行保护倒换时，只在头节点进行倒换操作，不会通知尾节点进行相应的倒换操作。

1.1.10 协议规范

与 MPLS TE 相关的协议规范有：

- 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
- ITU-T Recommendation Y.1720：Protection switching for MPLS networks

1.2 MPLS TE配置任务简介

表1-1 MPLS TE 配置任务简介

配置任务		说明	详细配置
配置MPLS TE基本能力		必选	1.3
配置MPLS TE隧道	使用静态CR-LSP配置MPLS TE隧道	二者必选其一	1.4
	使用动态信令协议配置MPLS TE隧道		1.5
配置RSVP-TE高级特性		可选	1.6
调整CR-LSP的建立		可选	1.7
调整MPLS TE隧道的建立		可选	1.8
配置流量转发	使用静态路由沿MPLS TE隧道转发流量	三者必选其一	1.9
	使用策略路由沿MPLS TE隧道转发流量		
	使用自动路由发布沿MPLS TE隧道转发流量		
配置影响流量转发的参数		可选	1.10
配置CR-LSP备份		可选	1.11
配置MPLS TE快速重路由		可选	1.12

配置任务	说明	详细配置
检测MPLS TE隧道	可选	1.13
配置保护倒换功能	可选	1.14

1.3 配置MPLS TE基本能力

MPLS TE 基本能力的配置任务包括各种 MPLS TE 特性配置中都涉及的基本配置，仅完成本节的配置并不能使用 MPLS TE 特性，还需要根据更具体的要求进行进一步的配置。

1.3.1 配置准备

在配置 MPLS TE 的基本能力之前，需要完成以下任务：

- 配置静态路由或 IGP 协议保证各 LSR 之间可达
- 配置 MPLS 基本能力

说明

有关 MPLS 基本能力的配置请参见“MPLS 配置指导”中的“MPLS 基本配置”。

1.3.2 配置MPLS TE基本能力

表1-2 配置 MPLS TE 基本能力

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
使能本节点的MPLS TE能力	**mpls te**	必选 缺省情况下，未使能本节点的MPLS TE能力
退回到系统视图	**quit**	-
进入MPLS TE链路的接口视图	**interface** *interface-type interface-number*	-
使能接口的MPLS TE能力	**mpls te**	必选 缺省情况下，未使能接口的MPLS TE能力
退回系统视图	**quit**	-
创建Tunnel接口，并进入Tunnel接口视图	**interface tunnel** *tunnel-number*	必选
配置隧道接口的IP地址	**ip address** *ip-address netmask*	可选

操作	命令	说明
配置隧道协议为MPLS TE	**tunnel-protocol mpls te**	必选
配置隧道的目的地址	**destination** *ip-address*	必选
配置隧道的Tunnel ID	**mpls te tunnel-id** *tunnel-id*	必选
提交隧道当前配置	**mpls te commit**	必选

说明

有关 Tunnel 接口的介绍和配置请参见“三层技术-IP 业务配置指导”中的“隧道”。

1.4 使用静态CR-LSP配置MPLS TE隧道

使用静态 CR-LSP 建立 MPLS TE 隧道的过程非常简单，不需要配置隧道的约束条件，也不涉及 IGP TE 扩展或 CSPF。只需要建立一条静态 CR-LSP 和使用静态信令的 TE 隧道，并将二者关联起来即可。

静态 CR-LSP 建立的 TE 隧道不能根据网络的变化动态调整，因此实际应用非常有限。

静态 CR-LSP 作为一种特殊的静态 LSP，其配置限制条件与静态 LSP 相同，使用相同的标签空间。

1.4.1 配置准备

在使用静态 CR-LSP 建立 MPLS TE 隧道之前，需完成以下任务：

- 配置静态路由或 IGP 协议保证各 LSR 之间可达
- 配置 MPLS 基本能力
- 配置 MPLS TE 基本能力

1.4.2 使用静态CR-LSP配置MPLS TE隧道

表1-3 使用静态 CR-LSP 配置 MPLS TE 隧道

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置隧道使用静态CR-LSP	**mpls te signal-protocol static**	必选
提交隧道当前配置	**mpls te commit**	必选
退回到系统视图	**quit**	-

操作		命令	说明
配置指定静态CR-LSP的节点时根据本节点在网络中的位置	为Ingress节点配置静态CR-LSP	**static-cr-lsp ingress** *tunnel-name* **destination** *dest-addr* **nexthop** *next-hop-addr* | **outgoing-interface** *interface-type interface-number*	必选 三者选择其一 参数*tunnel-name*是使用此静态CR-LSP的MPLS TE隧道接口的名称。例如，使用命令**interface tunnel** 2创建了一个Tunnel接口，该接口的名称为Tunnel2
	为Transit节点配置静态CR-LSP	**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*	
	为Egress节点配置静态CR-LSP	**static-cr-lsp egress** *tunnel-name* **incoming-interface** *interface-type interface-number* **in-label** *in-label-value*	

说明

- **static-cr-lsp ingress** 命令的 *tunnel-name* 参数必须与隧道接口名称完全相同，包括大小写。例如，使用命令 **interface tunnel** 2 创建了一个 Tunnel 接口，则其静态 CR-LSP 入节点中的 *tunnel-name* 参数应该写作“Tunnel2”，而不能写作“tunnel2”、“TUNNEL2”等。否则，隧道将不能正确建立。中间节点和出节点无此限制。
- 配置 Ingress 和 Transit 时，下一跳地址不能为本地的公网地址。

1.5 使用动态信令协议配置MPLS TE隧道

动态信令协议可以根据网络的动态变化调整 TE 隧道的路径，并能够应用备份、快速重路由等高级特性。

使用动态信令协议建立 MPLS TE 隧道的过程可以简要描述为以下几个部分：

- 配置链路的 MPLS TE 属性，并通过 IGP 的 TE 扩展发布，形成 TEDB；
- 配置隧道的约束条件；
- 使用 CSPF 算法，通过 TEDB 和隧道约束条件计算出符合要求的路径；
- 利用动态信令协议 RSVP-TE 建立路径。

说明

如果不配置 IGP 的 TE 扩展，网络中就不会有 TE LSA，不能形成 TEDB。这种情况下生成的 CR-LSP 是由 IGP 路由得到的，而不是 CSPF 计算出来的。

1.5.1 配置准备

在使用动态信令协议建立 MPLS TE 隧道之前，需完成以下任务：

- 配置静态路由或 IGP 协议保证各 LSR 之间可达
- 配置 MPLS 基本能力

- 配置 MPLS TE 基本能力

1.5.2 使用动态信令协议配置MPLS TE隧道

1. 使用动态信令协议配置MPLS TE隧道任务简介

表1-4 使用动态信令协议配置 MPLS TE 隧道过程

配置任务	说明	详细配置
配置CSPF	可选	2.
配置OSPF TE	配置CSPF后，二者必选其一 根据IGP协议的类型选择	3.
配置IS-IS TE		4.
配置MPLS TE显式路径	可选	5.
配置MPLS TE隧道的约束条件	可选	6.
使用RSVP-TE建立MPLS TE隧道	可选 缺省情况下，使用RSVP-TE建立MPLS TE隧道	7.

2. 配置CSPF

使能本节点的 CSPF 功能后，本节点将采用 CSPF 算法计算满足 TE 属性要求的最短路径。

表1-5 配置 CSPF

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
使能本节点的CSPF功能	**mpls te cspf**	必选 缺省情况下，未使能本节点的CSPF功能

3. 配置OSPF TE

当路由协议使用 OSPF 并使用动态信令协议配置 MPLS TE 隧道时，需要配置 OSPF TE 功能。OSPF TE 扩展使用 Opaque Type 10 LSA 携带链路的 TE 属性信息，因此，配置 OSPF TE 时必须先使能 OSPF 的 Opaque 能力。并且，只有当至少有一个邻居处于 FULL 状态时，才会产生 TE LSA。

表1-6 配置 OSPF TE

操作	命令	说明
进入系统视图	**system-view**	-
进入OSPF协议视图	**ospf** [*process-id*]	-

操作	命令	说明
使能OSPF的Opaque LSA能力	**opaque-capability enable**	必选 缺省情况下，未使能OSPF的Opaque LSA能力
进入OSPF的区域视图	**area** *area-id*	必选
在当前OSPF区域下使能MPLS TE能力	**mpls-te enable**	必选 缺省情况下，OSPF区域不支持MPLS TE能力
退回OSPF协议视图	**quit**	-

说明

- 有关 OSPF Opaque 能力的介绍请参见“三层技术-IP 路由配置指导”中的“OSPF”。
- 由于 MPLS TE 无法在 OSPF 虚连接上预留资源和分配标签，即 MPLS TE 无法通过 OSPF 虚连接建立 LSP 隧道，因此，配置 OSPF TE 时，OSPF 路由域内不能存在虚连接。

4. 配置IS-IS TE

当路由协议使用 IS-IS 并使用动态信令协议配置 MPLS TE 隧道时，需要配置 IS-IS TE 功能。当 OSPF TE 和 IS-IS TE 共存时，优选 OSPF 计算。

IS-IS TE 扩展使用 IS 可达性 TLV（类型为 22）的子 TLV 携带 TE 属性信息。因此，配置 IS-IS TE 时必须先使能 IS-IS 的 WideMetric 特性，可以配置 wide、compatible 或 wide-compatible。

注意

- RFC 3784 规定，在某些情况下 IS 可达性 TLV（22）的长度将会达到其最大长度 255 个字节。
- 为了使 IS-IS LSP 能顺利携带此类 TLV 并且在使能 IS-IS 的所有接口上进行正确泛洪，要求使能 IS-IS 的所有接口的 MTU 不小于 284 字节（其中包含 LSP 头长度 27 字节以及 TLV 头长度 2 字节）。如果 LSP 同时需要携带认证信息，则最小 MTU 需要根据报文结构另行推算。

综合考虑以上因素，在使用 TE 特性的情况下使能 IS-IS 的所有接口的 MTU 不要小于 512 字节，以确保 IS-IS LSP 能在网络上正确泛洪。

表1-7 配置 IS-IS TE

操作	命令	说明
进入系统视图	**system-view**	-
进入IS-IS协议视图	**isis** [*process-id*]	-
配置IS-IS的Wide Metric属性	**cost-style** { **narrow** | **wide** | **wide-compatible** | { **compatible** | **narrow-compatible** } [**relax-spf-limit**] }	必选 缺省情况下，IS-IS只收发采用Narrow方式表示路由权值的报文

操作	命令	说明
使能IS-IS TE功能	**traffic-eng** [**level-1** | **level-1-2** | **level-2**]	必选 缺省情况下，未使能IS-IS TE功能

说明

- 有关 IS-IS 的介绍请参见“三层技术-IP 路由配置指导”中的“IS-IS”。
- IS-IS TE 不支持发布从 IP 地址功能。在配置了多个 IP 地址的接口上使能 IS-IS TE 后，IS-IS TE 在其 LSP 的 IS 可达性 TLV（22）中只发布该接口的主 IP 地址。建议避免在配置了从 IP 地址的接口上配置 IS-IS TE 特性。

5. 配置MPLS TE显式路径

显式路径由一系列节点构成，一条显式路径上的两个相邻节点之间存在两种关系：

- 严格下一跳（**strict**）：两个节点必须直接相连；
- 松散下一跳（**loose**）：两个节点之间可以存在其他设备。

在向显式路径中增加或修改节点时，参数 **include** 表示建立的 LSP 必须经过指定节点；参数 **exclude** 表示建立的 LSP 不能经过指定节点。

表1-8 配置 MPLS TE 显式路径

操作	命令	说明
进入系统视图	**system-view**	-
创建隧道的显式路径，进入显式路径视图	**explicit-path** *path-name* [**disable** | **enable**]	必选
向显式路径中插入一个节点	**add hop** *ip-address1* [**include** [**loose** | **strict**] | **exclude**] { **after** | **before** } *ip-address2*	可选 缺省情况下，显式路径包含指定的节点（**include**），加入的节点是严格下一跳（**strict**）
在显式路径中指定下一跳IP地址	**next hop** *ip-address* [**include** [**loose** | **strict**] | **exclude**]	必选 缺省情况下，显式路径中指定的下一跳为严格（**strict**）下一跳
修改显式路径中的节点IP地址	**modify hop** *ip-address1 ip-address2* [[**include** [**loose** | **strict**]] | **exclude**]	可选 缺省情况下，显式路径包含指定的节点（**include**），修改的节点是严格下一跳（**strict**）
在显式路径中删除一个节点	**delete hop** *ip-address*	可选
显示显式路径节点信息	**list hop** [*ip-address*]	可选

说明

在不同的区域或自治系统之间建立 MPLS TE 隧道时必须使用松散显式路径，指定显式路径的下一跳为 ABR（Area Border Router，区域边界路由器）或 ASBR（Autonomous System Boundary Router，自治系统边界路由器），并保证隧道头节点与 ABR 之间或 ASBR 之间路由可达。

6. 配置MPLS TE隧道的约束条件

表1-9 配置 MPLS TE 隧道的约束条件

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置隧道应用的路径及路径的优先级	**mpls te path** { **dynamic** | **explicit-path** *path-name* } **preference** *value*	可选 缺省情况下，隧道应用动态自动计算的路径
提交隧道当前配置	**mpls te commit**	必选

7. 使用RSVP-TE建立MPLS TE隧道

表1-10 使用 RSVP-TE 建立 MPLS TE 隧道

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
使能本节点的RSVP-TE能力	**mpls rsvp-te**	必选 缺省情况下，未使能的本节点RSVP-TE能力
退回系统视图	**quit**	-
进入MPLS TE链路的接口视图	**interface** *interface-type interface-number*	-
使能接口的RSVP-TE能力	**mpls rsvp-te**	必选 缺省情况下，未使能接口的RSVP-TE能力
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置隧道使用RSVP-TE作为信令协议	**mpls te signal-protocol rsvp-te**	可选 缺省情况下，隧道使用RSVP-TE
提交隧道当前配置	**mpls te commit**	必选

注意

如果使用 RSVP-TE 作为 MPLS TE 隧道的信令协议，那么必须在隧道经过的路径上，使能接口的 MPLS TE 能力和 RSVP-TE 能力。

1.6 配置RSVP-TE高级特性

RSVP-TE 主要是在 Path 消息和 Resv 消息中增加了新的对象，以支持建立 CR-LSP。

RSVP-TE 提供了丰富的配置选项，能够满足用户对于可靠性、网络资源、以及一些 MPLS TE 高级特性的应用。

在实施本节介绍的配置任务之前，需要仔细了解各配置任务的目的和可能对网络造成的影响。

1.6.1 配置准备

在配置 RSVP-TE 高级特性之前，需完成以下任务：

- 配置 MPLS 基本能力
- 配置 MPLS TE 基本能力
- 使用 RSVP-TE 建立 MPLS TE 隧道

1.6.2 配置RSVP-TE高级特性

1. 配置RSVP资源预留风格

表1-11 配置 RSVP 资源预留风格

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置隧道的资源预留风格	**mpls te resv-style** { **ff** | **se** }	可选 缺省情况下，资源预留风格为**se**
提交隧道当前配置	**mpls te commit**	必选

说明

在目前的 MPLS TE 应用中，**se** 方式主要用于中断前建立（make-before-break），而 **ff** 方式则很少使用。

2. 配置RSVP的状态定时器

表1-12 配置 RSVP 的状态定时器

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
配置节点的路径/预留消息的刷新时间	**mpls rsvp-te timer refresh** *timevalue*	可选 缺省情况下，节点的路径/预留消息刷新周期为30秒
配置PSB和RSB的超时倍数	**mpls rsvp-te keep-multiplier** *number*	可选 缺省情况下，PSB和RSB的超时倍数为3
配置阻塞状态的超时倍数	**mpls rsvp-te blockade-multiplier** *number*	可选 缺省情况下，阻塞状态的超时倍数为4

3. 配置RSVP刷新机制

Message_ID 扩展机制（可靠性机制）用于 RSVP 消息确认，提高 RSVP 消息发送的可靠性。

在接口上使能可靠性功能后，可以使能重传功能。

摘要刷新 Srefresh（Summary Refresh）需要与 Message_ID 扩展配合使用。只有那些已经被包含 Message_ID 对象的 Path 和 Resv 消息发布过的状态才能使用摘要刷新扩展机制刷新。

表1-13 配置 RSVP 刷新机制

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE链路的接口视图	**interface** *interface-type interface-number*	-
使能RSVP可靠性功能	**mpls rsvp-te reliability**	可选 缺省情况下，禁止RSVP可靠性功能
使能重传功能	**mpls rsvp-te timer retransmission** { **increment-value** [*increment-value*] | **retransmit-value** [*retrans-timer-value*] } *	可选 缺省情况下，未使能重传功能
使能摘要刷新功能	**mpls rsvp-te srefresh**	可选 缺省情况下，未使能摘要刷新功能

4. 配置RSVP的Hello扩展

表1-14 配置 RSVP 的 Hello 扩展

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
使能本节点的RSVP的Hello扩展	**mpls rsvp-te hello**	必选 缺省情况下，未使能本节点的RSVP的Hello扩展
配置允许Hello消息丢失的最大次数	**mpls rsvp-te hello-lost** *times*	可选 缺省情况下，最多允许连续收不到3个Hello报文
配置Hello消息发送时间间隔	**mpls rsvp-te timer hello** *timevalue*	可选 缺省情况下，Hello消息的发送时间间隔为3秒
退回系统视图	**quit**	-
进入MPLS TE链路的接口视图	**interface** *interface-type interface-number*	-
使能接口的RSVP的Hello扩展	**mpls rsvp-te hello**	必选 缺省情况下，未使能接口RSVP的Hello扩展

说明

有关 RSVP 的 Hello 扩展机制用于检测 RSVP 邻居节点的可达性，在 RFC 3209 中有详细的介绍。

5. 配置RSVP-TE预留确认

表1-15 配置 RSVP-TE 预留确认

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS 视图	**mpls**	-
使能节点的预留确认机制	**mpls rsvp-te resvconfirm**	必选 缺省情况下，节点上不使能预留确认机制

说明

- 预留确认机制由接收者发起，接收者在发送Resv消息时，携带一个要求对预留进行确认的对象。
- 收到 ResvConf 消息并不表示成功建立了资源预留，只能说明在此 Resv 消息到达的最远上游节点处成功保留了资源，这些资源接下来仍可能被其它应用抢占。

6. 配置RSVP验证

为防止伪造的资源预留请求非法占用网络资源，RSVP 采用逐跳验证机制。

一条链路两端的接口需要配置相同的验证密钥，只有这样，它们才可以交换 RSVP 消息。

表1-16 配置 RSVP 验证

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE链路的接口视图	**interface** *interface-type interface-number*	-
使能RSVP验证功能	**mpls rsvp-te authentication** { **cipher** | **plain** } *auth-key*	必选 缺省情况下，禁止RSVP验证功能

说明

建议不要在同一个接口同时配置快速重路由功能和 RSVP 验证功能。

7. 配置发送的RSVP报文的DSCP优先级

用户可以对本设备发送的 RSVP 报文的 DSCP 优先级进行配置。

表1-17 配置发送的 RSVP 报文的 DSCP 优先级

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
配置发送的RSVP报文的DSCP优先级	**mpls rsvp-te dscp** *dscp-value*	必选 缺省情况下，发送的RSVP报文的DSCP优先级为48

8. 配置RSVP-TE GR

RSVP-TE GR 功能依赖于 RSVP-TE 的 Hello 扩展能力，因此在配置 RSVP-TE GR 功能之前必须使能 RSVP-TE 的 Hello 扩展能力。

在 GR 设备上进行以下配置。

表1-18 配置 RSVP-TE GR

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
使能本节点的RSVP的Hello扩展	**mpls rsvp-te hello**	必选 缺省情况下，未使能本节点的RSVP的Hello扩展
使能MPLS RSVP-TE协议的GR能力	**mpls rsvp-te graceful-restart**	必选 缺省情况下，MPLS RSVP-TE协议的GR能力处于关闭状态
配置RSVP-TE GR的重启定时器	**mpls rsvp-te timer graceful-restart restart** *restart-time*	可选 缺省情况下，RSVP-TE GR重启定时器的值为120秒
配置RSVP-TE GR的恢复定时器	**mpls rsvp-te timer graceful-restart recovery** *recovery-time*	可选 缺省情况下，RSVP-TE GR恢复定时器的值为300秒
进入MPLS TE链路的接口视图	**interface** *interface-type interface-number*	-
使能接口的RSVP的Hello扩展	**mpls rsvp-te hello**	必选 缺省情况下，未使能接口RSVP的Hello扩展

1.7 调整CR-LSP的建立

CSPF 使用 TEDB 和约束条件计算出符合要求的路径，并通过信令协议建立 CR-LSP。MPLS TE 提供多种方式影响 CSPF 的计算，从而调整 CR-LSP 的建立。

1.7.1 配置准备

本节的配置需要与 CSPF、动态信令协议（RSVP-TE）配合使用。

本节介绍的配置任务是 MPLS TE 中 CSPF 的一些专门配置，在实施本节的配置任务之前，需要明确理解这些配置对系统可能造成的影响。

1.7.2 调整CR-LSP的建立

1. 配置路由固定

如果使用路由固定特性，则不能同时使用 MPLS TE 重优化。

表1-19 配置路由固定

操作	命令	说明
进入系统视图	**system-view**	-

操作	命令	说明
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
使能路由固定功能	**mpls te route-pinning**	必选 缺省情况下，未使能路由固定功能
提交隧道当前配置	**mpls te commit**	必选

2. 配置管理组和亲和属性

MPLS TE 隧道的亲和属性决定隧道使用的链路属性，亲和属性与链路管理组配合，确定隧道可以使用哪些链路。管理组、亲和属性、掩码都是 32 位数值。

如果希望某条链路能够被隧道所用，则在所有掩码为 1 的位中，管理组中至少有 1 位与亲和属性中的相应位都为 1。对于掩码为 0 的位，则不对管理组的相应位进行检查。

举例说明：假设亲和属性为 0xFFFFFFFF，掩码为 0x0000FFFF，则可用链路的管理组属性高 16 位可以任意取 0 或 1，低 16 位则至少有 1 位为 1。

在隧道首节点配置隧道亲和属性后，亲和属性将通过信令协议（RSVP-TE）传递给各节点。

说明

不同设备制造商实现的管理组和亲和属性的关系可能有所不同，当在同一网络中使用不同设备制造商的设备时，需要事先了解各自的实现方式，正确配置链路的管理组和隧道的亲和属性，以便准确建立隧道。

表1-20 配置管理组和亲和属性

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE链路的接口视图	**interface** *interface-type interface-number*	-
配置链路的管理组	**mpls te link administrative group** *value*	可选 缺省情况下，链路管理组属性值为0x00000000
退回系统视图	**quit**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置隧道的亲和属性	**mpls te affinity property** *properties* [**mask** *mask-value*]	可选 缺省情况下，隧道的亲和属性为0x00000000，掩码为0x00000000
提交隧道当前配置	**mpls te commit**	必选

3. 配置CR-LSP重优化

动态优化 CR-LSP 即定期重计算 CR-LSP 穿越的路径。如果重计算的路径优于当前路径，则创建一条新的 CR-LSP，为之分配新路径，并将业务从旧的 CR-LSP 切换至新的 CR-LSP，删除旧 CR-LSP。

表1-21 配置 CR-LSP 重优化

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
使能隧道重优化功能	**mpls te reoptimization** [**frequency** *seconds*]	必选 缺省情况下，未使能重优化功能
提交隧道当前配置	**mpls te commit**	必选
退到用户视图	**return**	-
立即对所有使能了重优化功能的TE隧道进行重优化	**mpls te reoptimization**	可选

1.8 调整MPLS TE隧道的建立

在建立 MPLS TE 隧道的过程中，根据实际应用，可能还需要进行一些专门的配置。本节介绍这些特殊配置的使用。

1.8.1 配置准备

本节介绍的任务是 MPLS TE 中的一些专门配置，在实施本节的配置任务之前，需要明确理解这些配置对系统可能造成的影响。

本节介绍的配置需要与动态信令协议（如 RSVP-TE）配合使用。

1.8.2 调整MPLS TE隧道的建立

1. 配置环路检测

表1-22 配置环路检测

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置隧道建立时进行环路检测	**mpls te loop-detection**	必选 缺省情况下，不进行环路检测
提交隧道当前配置	**mpls te commit**	必选

2. 配置记录路由和标签

表1-23 配置记录路由和标签

操作		命令	说明
进入系统视图		**system-view**	-
进入MPLS TE隧道的Tunnel接口视图		**interface tunnel** *tunnel-number*	-
配置隧道支持路由记录或标签记录	建立隧道时记录路由	**mpls te record-route**	二者必选其一 缺省情况下，不进行路由记录和标签记录
	建立隧道时记录路由和标签	**mpls te record-route label**	
提交隧道当前配置		**mpls te commit**	必选

3. 配置隧道重建

配置隧道重建时间间隔和重建次数后，可以在隧道建立不成功，会每隔一定时间自动尝试重建，直到达到允许的最大重建次数。

表1-24 配置隧道重建

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置隧道重建的次数	**mpls te retry** *times*	可选 缺省情况下，隧道重建的次数为10次
配置隧道重建的间隔时间	**mpls te timer retry** *seconds*	可选 缺省情况下，重建隧道的时间间隔为2秒
提交隧道当前配置	**mpls te commit**	必选

4. 配置隧道的优先级

表1-25 配置隧道的优先级

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置隧道的优先级	**mpls te priority** *setup-priority* [*hold-priority*]	可选 缺省情况下，建立优先级和保持优先级的值都为7
提交隧道当前配置	**mpls te commit**	必选

1.9 配置流量转发

1.9.1 配置准备

在配置流量转发之前，需完成以下任务：

- 配置 MPLS 基本能力
- 配置 MPLS TE 基本能力
- 配置 MPLS TE 隧道

1.9.2 配置流量转发

1. 使用静态路由沿MPLS TE隧道转发流量

表1-26 使用静态路由沿 MPLS TE 隧道转发流量

操作	命令	说明
进入系统视图	**system-view**	-
使用静态路由沿MPLS TE隧道转发流量	**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*]	必选

说明

- *interface-type* 为 Tunnel 接口，并且必须设置 **preference** 值。
- 有关配置静态路由的命令请参见“三层技术-IP 路由命令参考”中的“静态路由”。

2. 使用自动路由发布沿MPLS TE隧道转发流量

配置自动路由发布前，需要首先在 MPLS TE 隧道的 Tunnel 接口使能 OSPF 或 ISIS 路由协议。

在 IGP Shortcut 和转发邻接特性中计算路径时，可以指定 TE 隧道使用的度量值。如果使用绝对度量（**absolute**），则 TE 隧道的度量值就是配置的值；如果使用相对度量（**relative**），则 TE 隧道的度量值是相应 IGP 路径度量值加上相对度量值。

(1) 配置 IGP Shortcut

表1-27 配置 IGP Shortcut

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-

操作	命令	说明
配置IGP在进行增强的SPF计算时，使用处于up状态的MPLS TE隧道	**mpls te igp shortcut** [**isis** | **ospf**]	必选 缺省情况下，IGP在进行增强的SPF计算时不使用MPLS TE隧道 如果配置IGP Shortcut时不指定IGP类型，则缺省为**ospf**和**isis**都支持
配置MPLS TE隧道的度量值	**mpls te igp metric** { **absolute** *value* | **relative** *value* }	可选 缺省情况下，TE隧道的度量等于其IGP度量
提交隧道当前配置	**mpls te commit**	必选
退回系统视图	**quit**	-
进入OSPF协议视图	**ospf** [*process-id*]	-
使能IGP Shortcut特性	**enable traffic-adjustment**	必选 缺省情况下，不使能IGP Shortcut特性

(2) 配置转发邻接

要想使转发邻接功能生效，需要创建双向 MPLS TE 隧道，并在隧道的两端同时配置转发邻接功能。

表1-28 配置转发邻接

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
将MPLS TE隧道作为链路发布到IGP网络	**mpls te igp advertise** [**hold-time** *value*]	必选 缺省情况下，MPLS TE隧道不被作为链路发布到IGP网络
配置MPLS TE隧道的度量值	**mpls te igp metric** { **absolute** | **relative** } *value*	可选 缺省情况下，TE隧道的度量等于其IGP度量
提交隧道当前配置	**mpls te commit**	必选
退回系统视图	**quit**	-
进入OSPF协议视图	**ospf** [*process-id*]	-
使能转发邻接特性	**enable traffic-adjustment advertise**	必选 缺省情况下，未使能转发邻接特性

说明

使用自动路由发布特性时，必须将 TE 隧道的目的地址指定为对端节点的 LSR ID，并且将 Tunnel 接口地址发布到 OSPF/ISIS 等 IGP 协议中。

1.10 配置影响流量转发的参数

在 MPLS TE 中，影响流量转发是指可能导致 IP 流量或 MPLS 流量经过的路径发生变化、限制 TE 隧道流量类型的相关配置。本节介绍 MPLS TE 中影响流量转发的几种主要措施。

1.10.1 配置准备

本节介绍的配置需要与 CSPF、动态信令协议（如 RSVP-TE）配合使用。

1.10.2 配置影响流量转发的参数

1. 配置失效链路定时器

一旦一条链路状态变为 down，失效链路定时器就会启动。如果 IGP 在定时器超时之前删除或修改此链路，IGP 将会把删除或修改情况通知 CSPF。CSPF 在 TEDB 中更新链路，并停止定时器。如果定时器超时时，IGP 还没有删除链路，链路状态将被更新为 up。

表1-29 配置失效链路定时器

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
配置失效链路定时器值	**mpls te cspf timer failed-link** *timer-interval*	可选 缺省情况下，失效链路定时器的值为10秒

2. 配置选路使用的度量

表1-30 配置选路使用的度量

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
配置未配置度量类型的隧道选路时使用的链路度量类型	**mpls te path metric-type { igp | te }**	可选 缺省情况下，选路时使用链路的TE度量
退回系统视图	**quit**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置当前隧道选路时使用的度量类型	**mpls te path metric-type { igp | te }**	可选 缺省情况下，没有指定选路时使用的链路度量类型，采用MPLS视图下配置的链路度量类型
提交隧道当前配置	**mpls te commit**	必选
退回系统视图	**quit**	-

操作	命令	说明
进入MPLS TE链路的接口视图	**interface** *interface-type interface-number*	-
配置链路的TE度量	**mpls te metric** *value*	可选 缺省情况下，链路使用其IGP度量作为TE的度量

说明

如果在隧道接口视图下没有配置 **mpls te path metric-type**，则使用 MPLS 视图下的配置。

3. 配置隧道转发的流量类型

表1-31 配置隧道转发的流量类型

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置TE隧道转发的流量类型	**mpls te vpn-binding** { **acl** *acl-number* | **vpn-instance** *vpn-instance-name* }	可选 缺省情况下，TE隧道不限制可以转发的流量类型
提交隧道当前配置	**mpls te commit**	必选

1.11 配置CR-LSP备份

CR-LSP 备份用于端到端的路径保护，对整条 LSP 提供保护。

1.11.1 配置准备

在配置 CR-LSP 备份之前，需完成以下任务：

- 配置 MPLS 基本能力
- 配置 MPLS TE 基本能力
- 配置 MPLS TE 隧道

1.11.2 配置CR-LSP备份

表1-32 配置 CR-LSP 备份

操作	命令	说明
进入隧道头节点的系统视图	**system-view**	-

操作	命令	说明
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
使能当前隧道的备份功能，并配置使用的备份模式	**mpls te backup { hot-standby | ordinary }**	必选 缺省情况下，隧道不进行备份
提交隧道当前配置	**mpls te commit**	必选

说明

CR-LSP 备份方式在隧道头节点进行配置，系统自动选择主 LSP 和备份 LSP 的路径，用户不需要分别配置主 LSP 和备份 LSP。

1.12 配置MPLS TE快速重路由

说明

建议不要在同一个接口同时配置快速重路由功能和 RSVP 验证功能。

FRR 是 MPLS TE 中的临时性局部保护技术。

由于 FRR 使用的 Bypass 隧道需要预先建立，占用额外的带宽，因此，在网络带宽余量不多的情况下，应该只对关键的接口或链路进行快速重路由保护。

用户可以指定哪种类型的 LSP 可以使用 Bypass 隧道，Bypass 隧道是否提供带宽保护以及带宽保护的总量。

Bypass 隧道的带宽用于保护主 LSP，用户在配置时应保证其带宽不小于被保护的所有 LSP 所需带宽之和，否则可能有主 LSP 不能绑定到 Bypass 隧道上。

Bypass 隧道一般不转发数据。如果 Bypass 隧道在保护主 LSP 的同时转发流量，需要为 Bypass 隧道提供足够的带宽。

Bypass 隧道不能同时做为 VPN 等业务的隧道使用。

1.12.1 配置准备

在配置 MPLS TE 快速重路由之前，需完成以下任务：

- 配置 IGP 协议保证各 LSR 之间可达
- 配置 MPLS 基本能力
- 配置 MPLS TE 基本能力
- 使用 RSVP-TE 建立 MPLS TE 隧道
- 建立主 LSP

1.12.2 配置MPLS TE快速重路由

1. 在主LSP的头节点使能快速重路由

表1-33 在主 LSP 的头节点使能快速重路由

操作	命令	说明
进入系统视图	**system-view**	-
进入主LSP的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
使能快速重路由功能	**mpls te fast-reroute**	必选 缺省情况下，未使能快速重路由功能
提交隧道配置	**mpls te commit**	必选

2. 在PLR上配置Bypass隧道

当一条隧道被指定保护一个接口以后，它所对应的 LSP 就成为 Bypass 隧道。Bypass 隧道的建立通过在 PLR 手工配置触发。它的配置与普通 LSP 相同，但不能配置快速重路由属性。也就是说，Bypass 隧道不能同时作为主 LSP，LSP 不能被嵌套保护。

为接口指定 Bypass 隧道，需要满足以下条件：

- Bypass 隧道必须是 up 状态的；
- 被保护的接口不是 Bypass 隧道的出接口。

设备最多可以为被保护的接口指定 3 条 Bypass 隧道，将使用最佳适配（best-fit）算法来决定使用哪一条 Bypass 隧道。

设备对使用 Bypass 隧道的链路进行限制，使这些隧道的带宽总量不超过指定的带宽。

表1-34 在 PLR 上配置 Bypass 隧道

操作	命令	说明
进入系统视图	**system-view**	-
进入Bypass隧道的接口视图	**interface tunnel** *tunnel-number*	-
配置Bypass隧道的目的地址	**destination** *ip-address*	必选 • 配置节点保护时，目的地址为 PLR 的下一跳路由器 LSR ID • 配置链路保护时，目的地址为 PLR 的下一跳设备的 LSR ID
提交隧道配置	**mpls te commit**	必选
退回系统视图	**quit**	-
进入被保护的LSP出接口的接口视图	**interface** *interface-type interface-number*	-
将Bypass隧道绑定到被保护的接口	**mpls te fast-reroute bypass-tunnel tunnel** *tunnel-number*	必选

3. 配置节点保护

如果使用 FRR 进行节点保护，则需要在 PLR 和被保护节点上进行本小节的配置；如果只是进行链路保护，则不必进行本小节的配置。

表1-35 配置节点保护

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
使能本节点的RSVP的Hello扩展	**mpls rsvp-te hello**	必选 缺省情况下，未使能本节点的RSVP的Hello扩展
退回系统视图	**quit**	-
进入PLR与被保护节点直连接口的接口视图	**interface** *interface-type interface-number*	-
使能接口的RSVP的Hello扩展	**mpls rsvp-te hello**	必选 缺省情况下，未使能接口的RSVP的Hello扩展

说明

对于链路失效引发的节点失效，不需要使用 RSVP 的 Hello 扩展机制来进行邻居检测。该配置主要用于在链路正常，但信令协议故障的特殊情况下检测节点失效。

4. 配置快速重路由扫描定时器

FRR 切换后，一旦被保护的 LSP 恢复正常或建立了新的 LSP，流量仍会切换回原 LSP 或新建立的 LSP。切换回来后，PLR 定时扫描，如果发现保护隧道，将刷新绑定关系。

表1-36 配置快速重路由扫描定时器

操作	命令	说明
进入PLR节点的系统视图	**system-view**	-
进入MPLS视图	**mpls**	-
配置快速重路由扫描间隔	**mpls te timer fast-reroute** [*seconds*]	可选 缺省情况下，扫描间隔为300秒

1.13 检测MPLS TE隧道

在 MPLS TE 网络中，MPLS TE 隧道出现故障，无法转发数据时，负责建立 MPLS TE 隧道的控制平面无法检测或不能快速发现这种错误，这会给网络维护带来困难。为了及时发现 MPLS TE 隧道错误，并定位失效节点，设备上提供了如下几种机制：

- MPLS LSP Ping 功能
- MPLS LSP Tracert 功能
- BFD 检测 MPLS TE 隧道功能
- MPLS TE 隧道的周期性 LSP Tracert 功能

1.13.1 MPLS LSP Ping功能

MPLS LSP Ping 功能可以用来对 MPLS TE 隧道的可达性进行检测。采取方法是：在头节点为 MPLS Echo Request 报文压入待检测的 MPLS TE 隧道对应的标签，使得 MPLS Echo Request 报文沿着 MPLS TE 隧道转发，头节点根据是否收到尾节点的应答报文，判断 MPLS TE 隧道是否可达。

表1-37 MPLS LSP Ping 功能

操作	命令	说明
通过MPLS LSP Ping检测MPLS TE隧道的可达性	**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*	必选 可在任意视图下执行本命令

1.13.2 MPLS LSP Tracert功能

MPLS LSP Tracert 用来对 MPLS TE 隧道的错误进行定位。MPLS LSP Tracert 通过连续发送 TTL 从 1 到某个值的 MPLS Echo Request 报文，让 MPLS TE 隧道路径上的每一跳收到该报文后，返回 TTL 超时的 MPLS Echo Reply 报文。这样，Ingress 节点可以收集到 MPLS TE 隧道上每一跳的信息，从而定位出故障节点。同时，MPLS LSP Tracert 还可用于收集整条 MPLS TE 隧道上每个节点的重要信息，如分配的标签等。

表1-38 MPLS LSP Tracert 功能

操作	命令	说明
通过MPLS LSP Tracert对MPLS TE隧道的错误进行定位	**tracert lsp** [**-a** *source-ip* | **-exp** *exp-value* | **-h** *ttl-value* | **-r** *reply-mode* |**-t** *time-out*] * **te** *interface-type interface-number*	必选 可在任意视图下执行本命令

1.13.3 配置BFD检测MPLS TE隧道功能

BFD 检测 MPLS TE 隧道功能通过在 MPLS TE 隧道头节点和尾节点之间建立 BFD 会话，利用 BFD 快速检测 MPLS TE 隧道的连通性。在头节点为 BFD 控制报文压入 MPLS TE 隧道的标签，沿着 MPLS TE 隧道转发 BFD 控制报文，并根据收到的尾节点的 BFD 控制报文来判断 MPLS TE 隧道的状态。当 BFD 检测到 MPLS TE 隧道故障后，还可以触发保护倒换进行流量的切换。

可以通过两种方式建立检测 MPLS TE 隧道的 BFD 会话：

- 静态方式：如果执行 **mpls te bfd enable** 命令时通过 **discriminator** 参数指定了本地和远端的鉴别值，则根据指定的鉴别值建立 BFD 会话。该方式用来检测两台设备间从本地到远端和从远端到本地的一对 MPLS TE 隧道。

- 动态方式：如果执行 **mpls te bfd enable** 命令时没有通过 **discriminator** 参数指定本地和远端的鉴别值，则自动运行 MPLS LSP Ping 来协商鉴别值，并根据协商好的鉴别值建立 BFD 会话。该方式用来检测两台设备间从本地到远端的一条单向 MPLS TE 隧道。

BFD 检测 MPLS TE 隧道功能还可以触发拆除故障的 RSVP-TE 隧道，并重新建立：通过 **mpls te failure-action teardown** 命令使能了 RSVP-TE 隧道故障后的拆除重建功能后，如果在 BFD 检测到 RSVP-TE 隧道故障后的一段时间内 RSVP 没有重新创建该 RSVP-TE 隧道，则将触发 MPLS TE 重新创建对应的 RSVP-TE 隧道。

1. 配置准备

- BFD 会话参数为 MPLS TE 隧道接口下配置的 BFD 参数，BFD 会话的源地址为 MPLS LSR-ID。因此，配置 BFD 检测 MPLS TE 隧道功能前，需要确保对端设备上存在 MPLS LSR-ID 对应的路由，还可以根据需要在隧道接口下配置 BFD 会话参数。BFD 的详细介绍，请参见“可靠性配置指导”中的“BFD”。
- 使用静态方式建立检测 MPLS TE 隧道的 BFD 会话前，需要确保已经建立从本地到远端和从远端到本地的两条 MPLS TE 隧道。

2. 配置BFD检测MPLS TE隧道功能

表1-39 配置 BFD 检测 MPLS TE 隧道功能

操作	命令	说明
进入系统视图	**system-view**	-
使能 LSP 检测功能，并进入 MPLS-LSPV视图	**mpls lspv**	必选 缺省情况下，未使能LSP检测功能
退回系统视图	**quit**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置使用BFD检测指定MPLS TE隧道的连通性	**mpls te bfd enable** [**discriminator local** *local-id* **remote** *remote-id*]	必选 缺省情况下，未使用BFD检测指定MPLS TE隧道的连通性
使能RSVP-TE隧道故障后的拆除重建功能	**mpls te failure-action teardown**	可选 缺省情况下，未使能RSVP-TE隧道故障后的拆除重建功能

说明

- **mpls lspv** 命令的详细介绍，请参见“MPLS 命令参考”中的“MPLS 基本配置”。
- 不能同时使用静态方式和动态方式建立检测同一个 MPLS TE 隧道的 BFD 会话。
- 使用静态方式建立检测 MPLS TE 隧道的 BFD 会话后，不能修改 BFD 会话的鉴别值。
- 如果在 MPLS TE 隧道上同时使用了 FRR 和 BFD 检测功能，则为了保证 FRR 切换或回切不会导致 BFD 会话 down，需要配置 BFD 会话的检测周期大于 FRR 的检测周期。
- 协议中规定：用于检测 MPLS TE 隧道的 BFD 会话中，头节点工作在主动（Active）模式，尾节点工作在被动（Passive）模式。在头节点和尾节点上执行 **bfd session init-mode** 命令不会改变节点的工作模式，即使头节点和尾节点均配置为被动模式，BFD 会话仍然可以正常建立。

1.13.4 配置MPLS TE隧道的周期性LSP Tracert功能

MPLS TE 隧道的周期性 LSP Tracert 功能，即周期性地对 MPLS TE 隧道进行 Tracert 检测，该功能用来对 MPLS TE 隧道的错误进行定位，对转发平面和控制平面一致性进行检测，并通过日志记录检测结果。管理员可以通过查看日志信息，了解 MPLS TE 隧道是否出现故障。

如果同时配置了 BFD 检测 MPLS TE 隧道功能和 MPLS TE 隧道的周期性 LSP Tracert 功能，则 MPLS TE 隧道的周期性 LSP Tracert 检测到转发平面故障或转发平面与控制平面不一致时，会触发删除原有 BFD 会话，并基于控制平面重新建立 BFD 会话。

MPLS TE 隧道的周期性 LSP Tracert 功能还可以触发拆除故障的 RSVP-TE 隧道，并重新建立：通过 **mpls te failure-action teardown** 命令使能了 RSVP-TE 隧道故障后的拆除重建功能后，如果在周期性 LSP Tracert 检测到 RSVP-TE 隧道故障后的一段时间内 RSVP 没有重新创建该 RSVP-TE 隧道，则将触发 MPLS TE 重新创建对应的 RSVP-TE 隧道。

表1-40 配置 MPLS TE 隧道的周期性 LSP Tracert 功能

操作	命令	说明
进入系统视图	**system-view**	-
启动LSP检测功能，并进入MPLS-LSPV视图	**mpls lspv**	必选 缺省情况下，未使能LSP检测功能
退回系统视图	**quit**	-
进入MPLS TE隧道的Tunnel接口视图	**interface tunnel** *tunnel-number*	-
使能指定MPLS TE隧道的周期性LSP Tracert检测功能	**mpls te periodic-tracert** [**-a** *source-ip* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *wait-time* | **-t** *time-out* | **-u** *retry-attempt*] *	必选 缺省情况下，未使能MPLS TE隧道的周期性LSP Tracert检测功能
使能RSVP-TE隧道故障后的拆除重建功能	**mpls te failure-action teardown**	可选 缺省情况下，未使能RSVP-TE隧道故障后的拆除重建功能

说明

mpls lspv 命令的详细介绍，请参见“MPLS 命令参考”中的“MPLS 基本配置”。

1.14 配置保护倒换功能

1.14.1 配置准备

在配置保护倒换功能之前，需完成以下任务：

- 配置 MPLS 基本功能
- 使能 MPLS TE 和创建 MPLS TE 隧道
- 配置 BFD 检测 MPLS TE 隧道功能

在配置 Tunnel 保护组之前，需准备以下数据：

- 保护组中主 Tunnel 的接口编号
- 保护组中保护 Tunnel ID

1.14.2 配置过程

表1-41 配置保护倒换功能

操作	命令	说明
进入系统视图	**system-view**	-
进入Tunnel接口视图	**interface tunnel** *tunnel-number*	-
配置主Tunnel的保护组	**mpls te protection tunnel** *tunnel-id* [**holdoff** *holdoff-time* | **mode** { **non-revertive** | **revertive** }] *	必选
配置外部倒换动作	**mpls te protect-switch** { **clear** | **force** | **lock** | **manual** { **protect-lsp** | **work-lsp** } }	可选 缺省情况下，没有配置外部倒换动作
提交隧道当前配置	**mpls te commit**	必选

1.15 MPLS TE显示和维护

1.15.1 MPLS TE显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 MPLS TE 的运行情况，通过查看显示信息验证配置的效果。

在用户视图下执行 **reset** 命令可以清除 MPLS TE 统计信息。

表1-42 MPLS TE 的显示和维护

操作	命令
显示显式路径信息	**display explicit-path** [*path-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示静态CR-LSP信息	**display mpls static-cr-lsp** [**lsp-name** *lsp-name*] [{ **include** | **exclude** } *ip-address prefix-length*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示RSVP-TE的配置信息	**display mpls rsvp-te** [**interface** [*interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]]
显示RSVP-TE隧道信息	**display mpls rsvp-te established** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示RSVP-TE邻居信息	**display mpls rsvp-te peer** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示RSVP-TE请求消息信息	**display mpls rsvp-te request** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示RSVP资源预留信息	**display mpls rsvp-te reservation** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示RSVP-TE PSB信息	**display mpls rsvp-te psb-content** *ingress-lsr-id lspid tunnel-id egress-lsr-id* [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示RSVP-TE RSB信息	**display mpls rsvp-te rsb-content** *ingress-lsr-id lspid tunnel-id egress-lsr-id nexthop-address* [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示RSVP-TE发送方消息信息	**display mpls rsvp-te sender** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示RSVP-TE的统计信息	**display mpls rsvp-te statistics** { **global** | **interface** [*interface-type interface-number*] } [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示满足指定条件的基于CSPF的TEDB信息	**display mpls te cspf tedb** { **all** | **area** *area-id* | **interface** *ip-address* | **network-lsa** | **node** [*mpls-lsr-id*] } [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示所有链路接纳的CR-LSP信息	**display mpls te link-administration admission-control** [**interface** *interface-type interface-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示MPLS TE隧道信息	**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*]
显示本地节点的MPLS TE隧道的路径属性信息	**display mpls te tunnel path** [**lsp-id** *lsr-id lsp-id* | **tunnel-name** *tunnel-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示MPLS TE隧道统计信息	**display mpls te tunnel statistics** [**|** { **begin** | **exclude** | **include** } *regular-expression*]

操作	命令
显示本地节点的MPLS TE隧道接口信息	**display mpls te tunnel-interface tunnel** *number* [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示与流量调节（IGP shortcut及转发邻接）相关的OSPF进程的隧道信息	**display ospf** [*process-id*] **traffic-adjustment** [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示OSPF TE信息	**display ospf** [*process-id*] **mpls-te** [**area** *area-id*] [**self-originated**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示IS-IS TE最近一次发布的TE信息	**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*]
显示IS-IS TE的链路信息	**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*]
显示IS-IS TE的网络信息	**display isis traffic-eng network** [**level-1** | **level-1-2** | **level-2**] [*process-id* | **vpn-instance** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示IS-IS TE的统计信息	**display isis traffic-eng statistics** [*process-id* | **vpn-instance** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示隧道信息	**display tunnel-info** { *tunnel-id* | **all** | **statistics** } [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示BFD对MPLS TE隧道的检测信息	**display mpls lsp bfd** [**te tunnel** *tunnel-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定Tunnel和其对应保护Tunnel信息	**display mpls te protection tunnel** { *tunnel-id* | **all** } [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
清除RSVP-TE的统计信息	**reset mpls rsvp-te statistics** { **global** | **interface** [*interface-type interface-number*]

1.16 MPLS TE典型组网举例

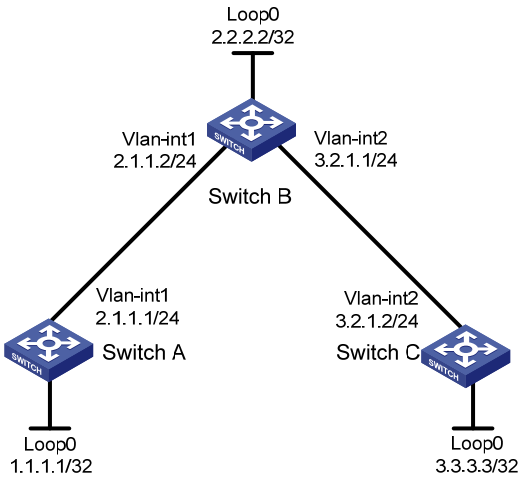
1.16.1 使用静态CR-LSP配置MPLS TE隧道示例

1. 组网需求

- 设备 Switch A、Switch B 和 Switch C 运行 IS-IS；
- 使用静态 CR-LSP 建立一条 Switch A 到 Switch C 的 TE 隧道。

2. 组网图

图1-6 静态 CR-LSP 配置组网图



3. 配置步骤

(1) 配置各接口的 IP 地址

按照 图 1-6 配置各接口的IP地址和掩码，具体配置过程略。

(2) 配置 IS-IS 协议发布 LSR ID 的主机路由

配置 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
```

配置 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
```

配置 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
```

配置完成后，在各设备上执行 **display ip routing-table** 命令，可以看到相互之间都学到了到对方的 LSR ID 的主机路由。以 Switch A 为例：

```
[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) 配置 MPLS TE 基本能力

配置 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
```

配置 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
```

配置 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) 配置 MPLS TE 隧道

在 Switch A 上配置 MPLS TE 隧道。

```
[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) 创建静态 CR-LSP

配置 Switch A 为静态 CR-LSP 的入口节点。

```
[SwitchA] static-cr-lsp ingress Tunnel0 destination 3.3.3.3 nexthop 2.1.1.2 out-label 20
```

配置 Switch B 为静态 CR-LSP 的中间节点。

```
[SwitchB] static-cr-lsp transit tunnel0 incoming-interface Vlan-interface1 in-label 20
nexthop 3.2.1.2 out-label 30
```

配置 Switch C 为静态 CR-LSP 的出节点。

```
[SwitchC] static-cr-lsp egress tunnel0 incoming-interface Vlan-interface2 in-label 30
```

(6) 配置完成后的检验

配置完成后，在 Switch A 上执行 **display interface tunnel** 命令，可以看到 Tunnel 接口的状态为 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
```

在各设备上执行 **display mpls te tunnel** 命令，可以看到 MPLS TE 隧道的建立情况。

```
[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
```

在各设备上执行 **display mpls lsp** 或 **display mpls static-cr-lsp** 命令，可以看到静态 CR-LSP 的建立情况。

```
[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       Vlan2/-
[SwitchA] display mpls static-cr-lsp
total statics-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
```

说明

使用静态 CR-LSP 建立 MPLS TE 隧道时，Transit 节点和 Egress 节点直接根据配置的入标签和出标签进行转发，因此，Switch B 和 Switch C 的相关显示信息中，FEC 等内容为空。

(7) 使用静态路由沿 MPLS TE 隧道转发流量

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

在 Switch A 上执行 **display ip routing-table** 命令，可以看到路由表中有以 Tunnel0 为出接口的静态路由信息。

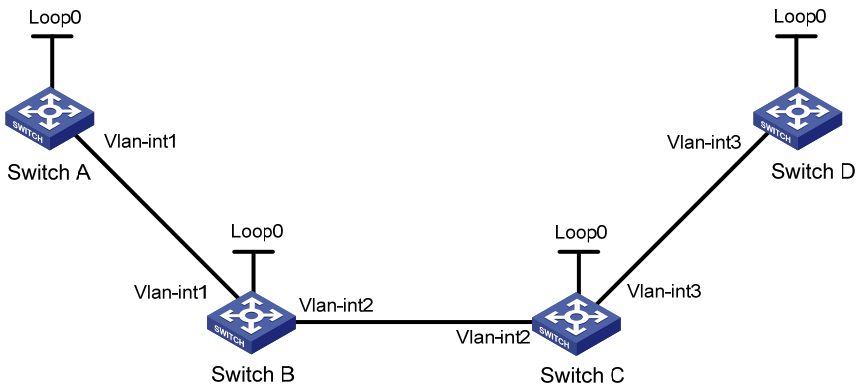
1.16.2 使用RSVP-TE配置MPLS TE隧道示例

1. 组网需求

- 设备 Switch A、Switch B、Switch C 和 Switch D 运行 IS-IS，都是 Level-2 设备；
- 使用 RSVP-TE 从 Switch A 到 Switch D 建立一条 TE 隧道，带宽为 2000kbit/s；
- 隧道沿途的链路最大带宽为 10000kbit/s。

2. 组网图

图1-7 RSVP-TE 配置 MPLS TE 隧道组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置各接口的 IP 地址

按照 图 1-7 配置各接口的IP地址和掩码，具体配置过程略。

(2) 配置 IS-IS 协议发布 LSR ID 的主机路由

配置 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-LoopBack0] isis enable 1
[SwitchA-LoopBack0] isis circuit-level level-2
[SwitchA-LoopBack0] quit
```

配置 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
```

配置 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
```

配置 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
```

配置完成后，在各设备上执行 **display ip routing-table** 命令，可以看到相互之间都学到了到对方的 LSR ID 的主机路由。以 Switch A 为例：

```
[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) 配置 MPLS TE 基本能力，并使能 RSVP-TE 和 CSPF

配置 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
```

配置 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-interface2] quit
```

配置 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
```

配置 Switch D。

```
[SwitchD] mpls lsr-id 4.4.4.9
[SwitchD] mpls
[SwitchD-mpls] mpls te
[SwitchD-mpls] mpls rsvp-te
[RouterD-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) 配置 IS-IS TE

配置 Switch A。

```
[SwitchA] isis 1
[SwitchA-isis-1] cost-style wide
[SwitchA-isis-1] traffic-eng level-2
[SwitchA-isis-1] quit
```

配置 Switch B。

```
[SwitchB] isis 1
[SwitchB-isis-1] cost-style wide
[SwitchB-isis-1] traffic-eng level-2
```

```
[SwitchB-isis-1] quit
```

配置 Switch C。

```
[SwitchC] isis 1
[SwitchC-isis-1] cost-style wide
[SwitchC-isis-1] traffic-eng level-2
[SwitchC-isis-1] quit
```

配置 SwitchD。

```
[SwitchD] isis 1
[SwitchD-isis-1] cost-style wide
[SwitchD-isis-1] traffic-eng level-2
[SwitchD-isis-1] quit
```

(5) 配置 MPLS TE 隧道

在 Switch A 上配置 MPLS TE 隧道。

```
[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 commit
[SwitchA-Tunnel1] quit
```

(6) 配置完成后的检验

配置完成后，在 Switch A 上执行 **display interface tunnel** 命令可以看到隧道接口状态为 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
```

在 Switch A 上执行 **display mpls te tunnel-interface** 命令可以看到隧道的详细信息。

```
[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          :  CT0               Tunnel BW    :  2000 kbps
    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          :  -
```

在 Switch A 上执行 **display mpls te cspf tedb all** 命令查看 TEDB 中的链路信息。

```
[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) 使用静态路由沿 MPLS TE 隧道转发流量

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

在 Switch A 上执行 **display ip routing-table** 命令，可以看到路由表中有以 Tunnel1 为出接口的静态路由信息。

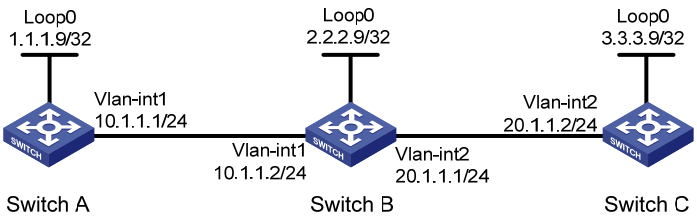
1.16.3 配置RSVP-TE GR示例

1. 组网需求

- 设备 Switch A、Switch B 和 Switch C 运行 IS-IS，都是 Level-2 设备；
- 使用 RSVP-TE 建立从 Switch A 到 Switch C 的 TE 隧道；
- 设备 Switch A、Switch B 和 Switch C 支持 RSVP 的 Hello 扩展能力；
- Switch A、Switch B 和 Switch C 为 RSVP-TE 邻居，通过使能 GR 能力，邻居之间可以在对方发生 GR Restart 的时候提供 GR Helper 支持。

2. 组网图

图1-8 RSVP-TE GR 配置组网图



3. 配置步骤

(1) 配置各接口的 IP 地址

按照 图 1-8 配置各接口的IP地址和掩码，具体配置过程略。

(2) 配置 IS-IS 协议发布 LSR ID 的主机路由

具体配置略。

(3) 配置 MPLS TE 基本能力，并使能 RSVP-TE 和节点的 RSVP 的 Hello 扩展

配置 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
```

配置 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
```

配置 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) 配置 IS-IS TE

具体配置略。

(5) 配置 MPLS TE 隧道

具体配置略。

(6) 配置 RSVP-TE GR

配置 Switch A

```
<SwitchA> system-view
[SwitchA] mpls
[SwitchA-mpls] mpls rsvp-te graceful-restart
```

配置 Switch B

```
<SwitchB> system-view
[SwitchB] mpls
[SwitchB-mpls] mpls rsvp-te graceful-restart
```

配置 Switch C

```
<SwitchC> system-view
[SwitchC] mpls
[SwitchC-mpls] mpls rsvp-te graceful-restart
```

(7) 检验配置效果

在 Switch A 和 Switch C 之间创建隧道且运行稳定后，通过命令可以看到邻居的 GR 状态已经处于 Ready 状态。

```
<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
```

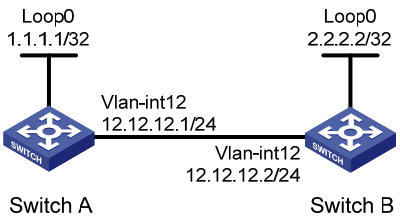
1.16.4 配置MPLS RSVP-TE与BFD联动

1. 组网需求

- Switch A、Switch B 物理直连，并且在双方直连口上使能 MPLS RSVP-TE BFD 应用，在设备上运行 OSPF，网络层相互可达。
- 当 Switch A 和 Switch B 之间的链路出现故障后，BFD 能够快速检测并通告 MPLS RSVP-TE 协议。

2. 组网图

图1-9 MPLS RSVP-TE 与 BFD 联动配置组网图



3. 配置步骤

(1) 配置 MPLS RSVP-TE 基本功能

配置 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
```

配置 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) 配置 OSPF 基本功能

配置 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
```

配置 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) 配置各交换机接口

配置 Switch A。

```
[SwitchA] interface vlan-interface 12
[SwitchA-Vlan-interface12] ip address 12.12.12.1 24
[SwitchA-Vlan-interface12] quit
```

配置 Switch B。

```
[SwitchB] interface vlan-interface 12
[SwitchB-Vlan-interface12] ip address 12.12.12.2 24
```

(4) 配置 MPLS TE 隧道

配置 Switch A 到 Switch B 的一条 RSVP-TE 隧道。

```
[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
```

(5) 检查配置结果

显示 Switch A 的 BFD 邻居详细信息。

```
<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
```

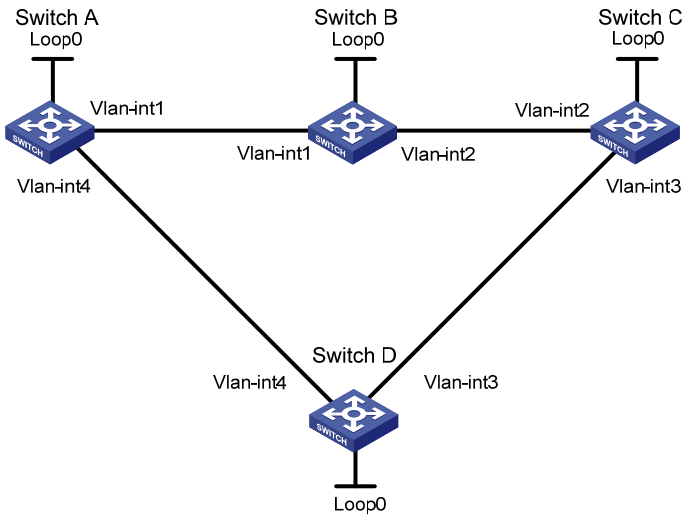
1.16.5 配置CR-LSP备份示例

1. 组网需求

要求建立一条 MPLS TE 隧道，使用 CR-LSP 的热备份。隧道头节点为 Switch A，目的节点为 Switch C。

2. 组网图

图1-10 CR-LSP 备份组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置各接口的 IP 地址

按照 图 1-10 配置各接口的IP地址和掩码，包括各Loopback接口，具体配置过程略。

(2) 配置 IGP 协议

在所有设备上配置 IS-IS 协议，发布 LSR ID 的主机路由，具体配置过程略。

配置完成后，在各设备上执行 **display ip routing-table** 命令，可以看到相互之间都学到了到对方 LSR ID 的主机路由。

(3) 配置 MPLS TE 基本能力，并使能 RSVP-TE 和 CSPF

配置 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
```

说明

Switch B、Switch C 和 Switch D 的配置与 Switch A 相似，此处不再赘述。

(4) 在 Switch A 上建立 MPLS TE Tunnel

配置主 LSP 的 MPLS TE 隧道。

```
[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
[SwitchA-Tunnel1] mpls te record route
```

使用热备份。

```
[SwitchA-Tunnel1] mpls te backup hot-standby
[SwitchA-Tunnel1] mpls te commit
[SwitchA-Tunnel1] quit
```

配置完成后，在 Switch A 上执行 **display interface tunnel** 命令，可以看到 Tunnel1 的状态为 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) 配置完成后的检验

在 Switch A 上执行 **display mpls te tunnel** 命令，可以看到存在两条 Tunnel，出接口分别是 Vlan-intface1 和 Vlan-intface4，即，在主 CR-LSP 创建后，备份 CR-LSP 也建立了。

```
[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
```

在 Switch A 上执行 **display mpls te tunnel path** 命令，可以看到这两条 Tunnel 使用的路径：

```
[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
```

对隧道目的地址进行 Tracert 操作：

```
[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
```

从以上显示可以判断出目前使用的 LSP 经过 SwitchB，没有使用 SwitchD。

对 SwitchB 的接口 Vlan-intface2 执行 **shutdown**，然后再对目的地址进行 Tracert 操作，可以看到报文经过了 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
```

在 Switch A 上执行 **display mpls te tunnel** 命令，可以看到只剩下一条经过 Switch D 的 Tunnel：

```
[SwitchA] display mpls te tunnel
LSP-Id            Destination     In/Out-If             Name
1.1.1.9:2054      3.3.3.9         -/Vlan4               Tunnel1
```

说明

普通 CR-LSP 备份与热备份在配置上的唯一区别在于隧道接口上的 **mpls te backup** 命令。对于本配置例，如果采用普通备份方式，则备份 CR-LSP 只有在主 LSP 变为 down 后才会创建，不会同时存在两条隧道。

(6) 使用静态路由沿 MPLS TE 隧道转发流量

```
[SwitchA] ip route-static 20.1.1.2 24 tunnel 1 preference 1
```

在 Switch A 上执行 **display ip routing-table** 命令，可以看到路由表中有以 Tunnel1 为出接口的静态路由信息。

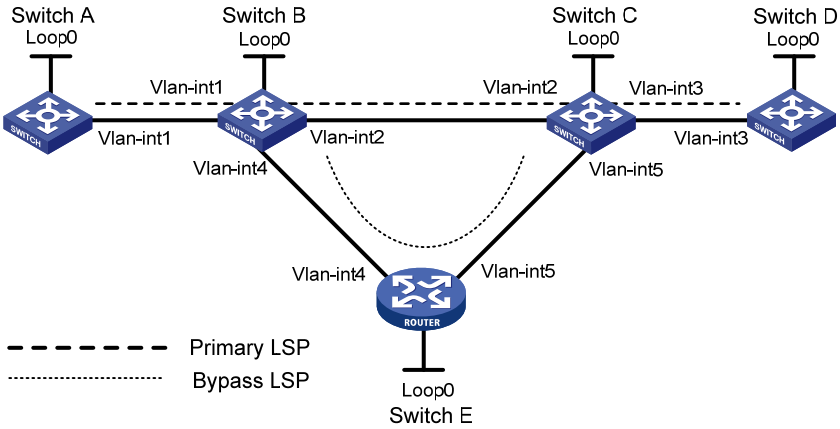
1.16.6 配置快速重路由示例

1. 组网需求

- 主 LSP 是 Switch A→Switch B→Switch C→Switch D，要求对 Switch B→Switch C 这段链路通过 FRR 进行链路保护；
- 建立一条 Bypass LSP，使用路径 Switch B→Switch E→Switch C（Switch B 是本地修复节点 PLR，Switch C 是汇聚点 MP）；
- 使用显式路径方式建立 MPLS TE 的主隧道和 Bypass 隧道。使用的信令协议是 RSVP-TE。

2. 组网图

图1-11 MPLS TE 快速重路由配置组网图



设备	接口	IP地址	设备	接口	IP地址
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			

3. 配置步骤

(1) 配置各接口的 IP 地址

按照 图 1-11 配置各接口的IP地址和掩码，包括各Loopback接口，具体配置过程略。

(2) 配置 IGP 协议

在所有设备上配置 IS-IS 协议，发布 LSR ID 的主机路由，具体配置过程略。

配置完成后，在各设备上执行 **display ip routing-table** 命令，应可以看到相互之间都学到了到对方 LSR ID 的主机路由。

以 Switch A 的显示为例：

```
<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) 配置 MPLS TE 基本能力，并使能 RSVP-TE 和 CSPF

配置 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
```

配置 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
```

说明

Switch C、Switch D、Switch E 的配置与 Switch A 和 Switch B 的配置相同，此处不再赘述。

(4) 在主 LSP 的头节点 Switch A 上建立 MPLS TE Tunnel

配置主 LSP 的显式路径。

```
[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
```

配置主 LSP 的 MPLS TE 隧道。

```
[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
```

使能 FRR。

```
[SwitchA-Tunnel4] mpls te fast-reroute
[SwitchA-Tunnel4] mpls te commit
[SwitchA-Tunnel4] quit
```

配置完成后，在 Switch A 上执行 **display interface tunnel** 命令，可以看到 Tunnel4 的状态为 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
```

在 Switch A 上执行 **display mpls te tunnel-interface** 命令，可以看到隧道接口的详细信息。

```
[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) 在作为 PLR 的 Switch B 上配置 Bypass Tunnel

配置 Bypass LSP 的显式路径。

```
[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
```

配置 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
```

将 Bypass Tunnel 绑定到被保护的接口。

```
[SwitchB] interface vlan-interface 2
[SwitchB-Vlan-interface2] mpls te fast-reroute bypass-tunnel tunnel 5
[SwitchB-Vlan-interface2] quit
```

配置完成后，在 Switch B 上执行 **display interface tunnel** 命令可以看到接口 Tunnel5 的状态为 up。在所有设备上执行 **display mpls lsp** 命令，可以看到 LSP 表项，并且，Switch B 和 Switch C 上有两条 LSP 经过。

```
[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
```

在所有设备上执行 **display mpls te tunnel** 命令，可以看到隧道的建立情况，Switch B 和 Switch C 上有两条隧道经过。

```
[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
```

在 Switch B 上执行 **display mpls lsp verbose** 命令，可以看到 Bypass 隧道绑定到出接口 Vlan-interface2，目前未被使用。

```
[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]

  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[---]
```

(6) 配置完成后的检验

使 PLR 上被保护的出接口失效。

```
[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
```

在 Switch A 上查看主 LSP 的状态，执行命令 **display interface tunnel** 4，可以看到 Tunnel 接口仍然处于 up 状态。

在 Switch A 上执行 **display mpls te tunnel-interface** 命令，可以看到隧道接口的详细信息。

```
[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          :  -
```

说明

如果在 FRR 切换后马上执行 **display mpls te tunnel-interface** 命令查看隧道接口的详细信息，会看到两条处于 up 状态的 CR-LSP。这是因为 FRR 采用 make-before-break 方式建立新的 LSP，旧的 LSP 在新 LSP 建立成功后过一段时间才被删除。

在 Switch B 上执行 **display mpls lsp verbose** 命令，可以看到 Bypass 隧道被使用。

```
[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]

  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[---]
```

在 PLR 上配置 FRR 的扫描定时器为 5 秒。

```
[SwitchB] mpls
[SwitchB-mpls] mpls te timer fast-reroute 5
[SwitchB-mpls] quit
```

使 PLR 上被保护的出接口重新生效。

```
[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
```

在 Switch A 上查看主 LSP 的状态，执行命令 **display interface tunnel** 4，可以看到 Tunnel 接口处于 up 状态。

等待约 5 秒钟后，在 Switch B 上执行 **display mpls lsp verbose** 命令，可以看到 Tunnel5 仍绑定到出接口 Vlan-interface2，未被使用。

(7) 使用静态路由沿 MPLS TE 隧道转发流量

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

在 Switch A 上执行 **display ip routing-table** 命令，可以看到路由表中有以 Tunnel4 为出接口的静态路由信息。

1.16.7 配置MPLS L3VPN中的MPLS TE示例

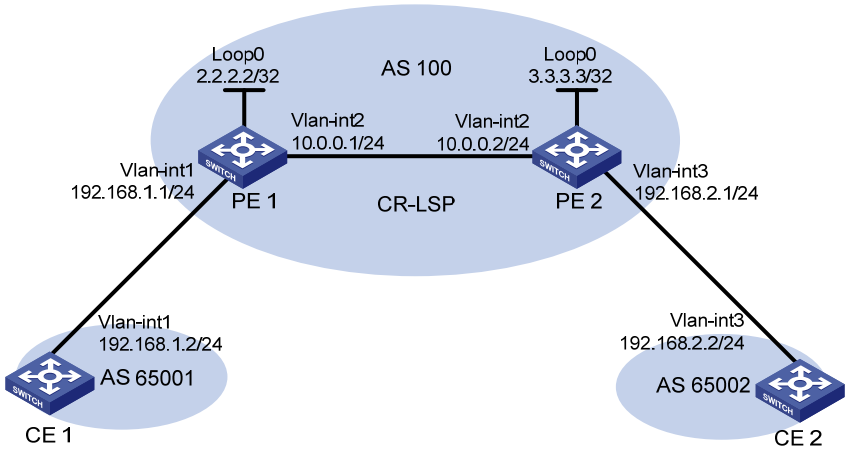
1. 组网需求

- CE 1 和 CE 2 都属于 VPN 1，分别通过 PE 1 和 PE 2 接入 MPLS 骨干网。MPLS 骨干网使用 OSPF 作为 IGP 协议；
- 配置 MPLS TE 隧道转发 PE 1 到 PE 2 的 VPN 流量；

- 为了使 TE 隧道能够转发 MPLS L3VPN 的流量，需要在建立 VPN 时配置隧道策略，使用 CR-LSP 作为 VPN 隧道。

2. 组网图

图1-12 在 VPN 中配置 MPLS TE 组网图



3. 配置步骤

(1) 配置 IGP 协议 OSPF，使 PE 1 和 PE 2 能学习到对方 LSR-ID 的路由。

配置 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
```

配置 PE2。

```
<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
```

上述配置完成后，PE 之间应能建立 OSPF 邻居，执行 **display ospf peer verbose** 命令可以看到邻居达到 FULL 状态。执行 **display ip routing-table** 命令可以看到 PE 之间学习到对方的 Loopback 路由。以 PE 1 为例：

```
[PE1] display ospf peer verbose
          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) 配置 MPLS TE 基本能力，并使能 RSVP-TE 和 CSPF

配置 PE 1。

```
[PE1] mpls lsr-id 2.2.2.2
[PE1] mpls
[PE1-mpls] lsp-trigger all
[PE1-mpls] mpls te
[PE1-mpls] mpls rsvp-te
[PE1-mpls] mpls te cspf
[PE1-mpls] quit
[PE1] interface vlan-interface 2
[PE1-Vlan-interface2] mpls
[PE1-Vlan-interface2] mpls te
[PE1-Vlan-interface2] mpls rsvp-te
[PE1-Vlan-interface2] quit
```

配置 PE2。

```
[PE2] mpls lsr-id 3.3.3.3
[PE2] mpls
[PE2-mpls] lsp-trigger all
[PE2-mpls] mpls te
[PE2-mpls] mpls rsvp-te
[PE2-mpls] mpls te cspf
```

```
[PE2-mpls] quit
[PE2] interface vlan-interface 2
[PE2-Vlan-interface2] mpls
[PE2-Vlan-interface2] mpls te
[PE2-Vlan-interface2] mpls rsvp-te
[PE2-Vlan-interface2] quit
```

(3) 使能 OSPF TE

配置 PE 1。

```
[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
```

配置 PE 2。

```
[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) 配置 MPLS TE 隧道

以 PE 1 为首端创建一条 TE 隧道，尾端为 PE 2，信令协议为 RSVP-TE。

```
[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 rsvp-te
[PE1-Tunnel1] mpls te commit
[PE1-Tunnel1] quit
```

上述配置完成后，在 PE 1 上执行 **display interface tunnel** 命令可以看到隧道接口状态为 up。

(5) 在 PE 设备上配置 VPN 实例，并绑定到连接 CE 的接口上

配置 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
```

在 PE 1 上配置 VPN 实例，使用 CR-LSP 建立 VPN，并将此实例绑定到连接 CE 1 的接口。

```
[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
```

配置 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
```

在 PE 2 上配置 VPN 实例，并将此实例绑定到连接 CE 2 的接口。

```
[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
```

上述配置完成后，在 PE 设备上执行 **display ip vpn-instance** 命令可以看到 VPN 实例的配置情况。

以 PE 1 为例：

```
[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
  Interfaces : Vlan-interface1
```

各 PE 能 ping 通自己接入的 CE。

以 PE 1 和 CE 1 为例：

```
[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
```

(6)　配置 BGP

配置 CE 1。

```
[CE1] bgp 65001
[CE1-bgp] peer 192.168.1.1 as-number 100
[CE1-bgp] quit
```

配置 PE 1：与 CE 1 建立 EBGP 对等体，与 PE 2 建立 IBGP 对等体。

```
[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 loopback 0
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 3.3.3.3 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

配置 CE 2。

```
[CE2] bgp 65002
[CE2-bgp] peer 192.168.2.1 as-number 100
[CE2-bgp] quit
```

配置 PE 2：与 CE 2 建立 EBGP 对等体，与 PE 1 建立 IBGP 对等体。

```
[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
```

上述配置完成后，在 PE 设备上执行 **display bgp peer** 命令和 **display bgp vpn-instance peer** 命令，可以看到 PE 之间、PE 与 CE 之间的 BGP 对等体关系都已建立，并达到 Established 状态。

以 PE 1 为例：

```
[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
```

CE 1 和 CE 2 之间能够互相 ping 通。

```
[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
```

(7) 配置完成后的检验

在 PE 1 上执行 **display mpls lsp verbose** 命令，可以确认 LspIndex 为 2050 的是使用 RSVP-TE 建立的 LSP，即 MPLS TE 隧道。

```
[PE1] display mpls lsp verbose
--------------------------------------------------------------------
                 LSP Information: RSVP 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[---]
--------------------------------------------------------------------
                 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
```

在 PE 1 上执行 **display interface tunnel** 命令可以看到流量通过 TE 隧道 CR-LSP 转发。

```
[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 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:  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
```

1.17 MPLS TE常见配置错误举例

1.17.1 不能产生TE LSA

1. 故障现象

配置 OSPF TE，无法产生描述 MPLS TE 信息的 TE LSA。

2. 分析

至少有一个 OSPF 邻居达到 FULL 状态时，才可能产生 TE LSA。

3. 处理过程

- 执行 **display current-configuration** 命令，检查是否在相关接口上配置了 MPLS TE；
- 执行 **debugging ospf mpls-te** 命令打开 OSPF TE 的调试开关，检查 OSPF 是否收到建立 TE LINK 的消息；
- 执行 **display ospf peer** 命令，检查 OSPF 邻居是否正常建立。

目 录

1 VPLS ····· 1-1
1.1 VPLS简介 ····· 1-1
1.1.1 VPLS的工作机制 ····· 1-1
1.1.2 VPLS报文封装 ····· 1-5
1.1.3 H-VPLS实现方式 ····· 1-5
1.2 VPLS配置任务简介 ····· 1-7
1.3 使能L2VPN和MPLS L2VPN ····· 1-8
1.4 配置LDP方式的VPLS ····· 1-8
1.4.1 配置准备 ····· 1-8
1.4.2 配置LDP方式的VPLS实例 ····· 1-8
1.5 配置BGP方式的VPLS ····· 1-10
1.5.1 配置准备 ····· 1-10
1.5.2 配置BGP扩展 ····· 1-10
1.5.3 配置BGP方式的VPLS实例 ····· 1-10
1.5.4 复位VPLS的BGP连接 ····· 1-11
1.6 绑定VPLS实例 ····· 1-11
1.6.1 配置服务实例与VPLS实例绑定 ····· 1-11
1.7 配置MAC地址学习功能 ····· 1-12
1.8 配置MAC地址迁移功能 ····· 1-12
1.9 配置VPLS实例的属性 ····· 1-13
1.10 检测PW ····· 1-13
1.11 VPLS显示和维护 ····· 1-14
1.12 VPLS典型配置举例 ····· 1-14
1.12.1 服务实例与VPLS实例绑定配置举例 ····· 1-14
1.12.2 H-VPLS备份链路配置举例 ····· 1-19
1.12.3 H-VPLS组网中通过BFD检测主链路故障配置举例 ····· 1-23
1.13 常见配置错误举例 ····· 1-28
1.13.1 PW状态不是up ····· 1-28

1 VPLS

说明

本系列交换机中 S5500-28SC-HI 和 S5500-52SC-HI 不支持 VPLS 功能。

1.1 VPLS简介

VPLS（Virtual Private LAN Service，虚拟专用局域网服务）是在公用网络中提供的一种点到多点的 L2VPN 业务。VPLS 使地域上隔离的用户站点能通过 MAN（Metropolitan Area Network，城域网）或 WAN（Wide Area Network，广域网）相连，并且使各个站点间的连接效果像在一个 LAN 中一样。

VPLS 也称 TLS（Transparent LAN Service，透明局域网服务）或 Virtual Private Switched Network Service（虚拟专有交换网络服务）。

VPLS 提供二层 VPN 服务。在 VPLS 中，用户是由多点网络连接起来，不同于传统 VPN 提供的 P2P（Point-to-Point，点到点）的连接服务。VPLS 实际上就是在 PE 上创建一系列的虚拟交换机租借给用户，虚拟交换机的组网和传统交换机完全相同，这样，用户就可以通过 MAN（Metropolitan Area Network，城域网）或 WAN（Wide Area Network，广域网）来实现自己的 LAN（Local Area Network，局域网）。

1.1.1 VPLS的工作机制

1. VPLS的基本概念

- CE（Custom Edge）

直接与服务提供商相连的用户边缘设备。

- PE（Provider Edge）

服务提供商网络上的边缘设备，与 CE 相连，主要负责 VPN 业务的接入。它完成报文从私网到公网隧道，并从公网隧道到私网的映射与转发。PE 可以细分为 UPE 和 NPE。

- UPE（User facing-Provider Edge）

靠近用户侧的 PE 设备，主要作为用户接入 VPN 的汇聚设备。

- NPE（Network Provider Edge）

网络核心 PE 设备，处于 VPLS 网络的核心域边缘，提供在核心网之间的 VPLS 透明传输服务。

- VSI（Virtual Switch Instance）

虚拟交换实例，下文中简称为 VPLS 实例。通过 VSI，可以将实际接入链路映射到各条虚链接上。

- PW（Pseudo Wire）

虚链路，在两个 VSI 之间的一条双向的虚拟连接，它由一对单向的 MPLS VC（Virtual Circuit，虚电路）构成。

- 服务实例（Service Instance）

服务实例在端口下创建，用来对通过该端口的报文进行识别和处理。PE 设备通过服务实例对从 CE 收到的流量进行分类匹配，从而将具备不同特征的流量通过不同的 PW 进行转发。服务实例提供了多种报文匹配规则，为报文接入 VPLS 实例提供了更加灵活的方式。

- AC（Attachment Circuit）

接入电路，指连接 CE 与 PE 的链路，对应的接口可以是实际的物理接口，也可以是虚拟接口。AC 上的所有用户报文一般都要求原封不动的转发到对端 Site（站点）去，包括用户的二、三层协议报文。在交换机上，端口上每一个服务实例对应一条 AC。

- QinQ（802.1Q in 802.1Q）

一种基于 802.1Q 封装的隧道协议，能够提供点到多点的 L2VPN 服务机制。它将用户私网 VLAN Tag 封装在公网 VLAN Tag 中，最终报文带着两层 Tag 穿越服务提供商的骨干网络，从而为用户提供一种较为简单的二层 VPN 隧道。

- Forwarders

转发器，PE 的一种。PE 收到 AC 上送的数据帧，由转发器选定转发报文使用的 PW，转发器事实上就是 VPLS 的转发表。

- Tunnel

隧道，用于承载 PW，一条隧道上可以承载多条 PW，一般情况下为 MPLS 隧道。隧道是一条本地 PE 与对端 PE 之间的直连通道，完成 PE 之间的数据透明传输。

- Encapsulation

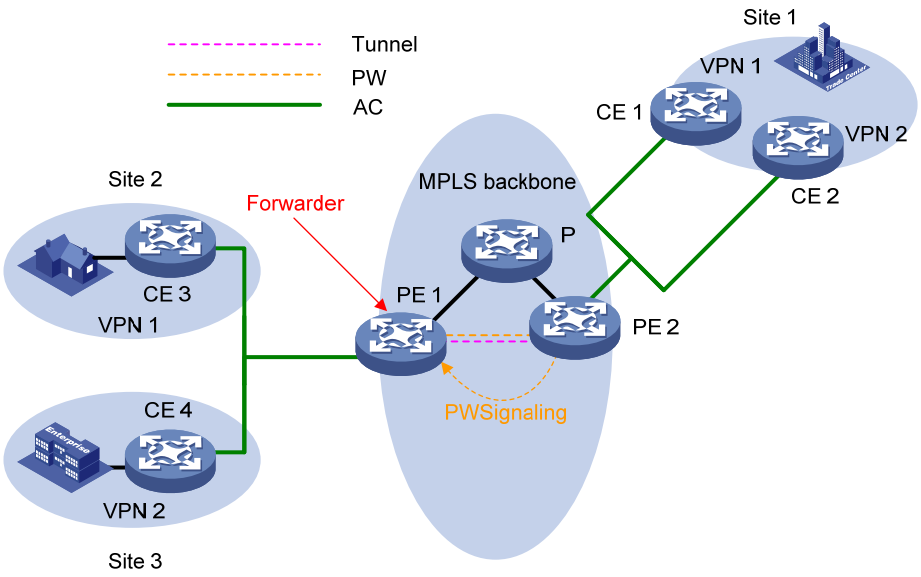
封装，PW 上传输的报文使用标准的 PW 封装格式和技术。PW 上的 VPLS 报文封装有两种模式：Ethernet 和 VLAN 模式。

- PW Signaling

PW 信令协议，VPLS 实现的基础，用于创建和维护 PW。PW 信令协议还可用于自动发现 VSI 的对端 PE 设备。目前，PW 信令协议主要有 LDP 和 BGP。

图 1-1 为VPLS典型组网示意图，图中简单显示出以上所涉及的各基本概念。

图1-1 VPLS 典型组网示意图



2. PW的创建

PW 是 VPLS 在公网上的通信隧道，它建立在 MPLS（包括 LSP 和 CR-LSP）隧道之上。创建 PW 需要：

(1) 在本端和对端 PE 之间建立 MPLS 隧道。

(2) 确定对端 PE 的地址。对于同一个 VSI 内的 PE 设备，可以通过手工配置来指定对端 PE 地址，也可以通过信令协议（如 BGP）自动发现对端 PE。

(3) 在两端的 PE 设备上分别为 PW 分配多路复用分离标记（VC 标签），并将分配的 VC 标签通告给对端 PE，建立单向的 VC。一对单向的 VC 建立成功后，它们组合起来形成双向的 PW。

根据 PW 建立方式的不同，VPLS 分为：

- LDP 方式的 VPLS：采用 LDP 作为信令协议为 PW 分发 VC 标签，该方式也称为 Martini 方式。
- BGP 方式的 VPLS：采用 BGP 扩展作为信令协议为 PW 分发 VC 标签，该方式也称为 Kompella 方式。

说明

有关 Martini 方式和 Kompella 方式的介绍请参见“MPLS 配置指导”中的“MPLS L2VPN”。

---

3. MAC地址学习与泛洪

VPLS 通过 MAC 地址学习来提供可达性。每个 PE 设备会维护一张 MAC 地址表。

(1) 源 MAC 地址学习

MAC 地址学习过程包含两部分：

- 与 PW 关联的远程 MAC 地址学习

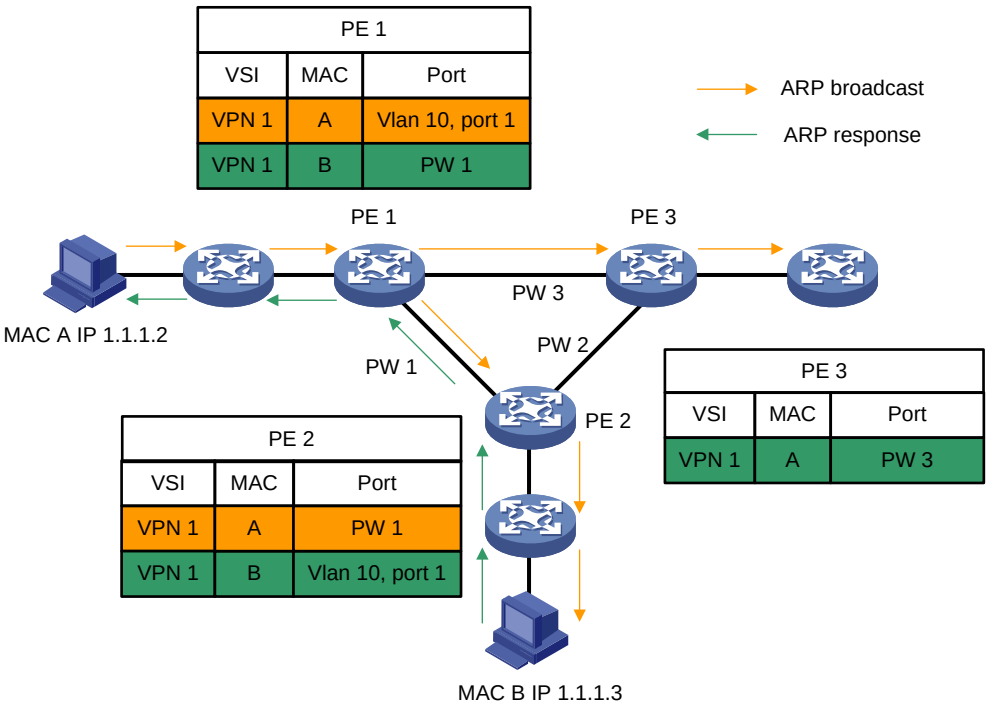
PW 是由一对单向的 VC LSP 组成（只有两个方向的 VC LSP 都 up 才被认为 PW 是 up 的）。当在入方向的 VC LSP 上学习到一个原来未知的 MAC 地址后，需要 PW 将此 MAC 地址与出方向的 VC LSP 形成映射关系。

- 与用户直接相连端口的本地 MAC 地址学习

对于 CE 上传送的报文，需要将报文中的源 MAC 地址学习到 VSI 的对应端口上。

PE的MAC地址学习与泛洪过程如 图 1-2 所示。

图1-2 PE 的 MAC 地址学习与泛洪过程



(2) MAC 地址回收

动态学习到的 MAC 地址必须有刷新和重学习的机制。在 VPLS 相关草案中提供一种动态学习的方法，即使用地址回收消息。地址回收消息中携带 MAC TLV，收到这个消息的设备根据 TLV 中指定的参数进行 MAC 地址的删除或者重新学习这些 MAC 地址。如果 TLV 中指定的 MAC 地址为 NULL，则删除此 VSI 下所有 MAC 地址，但不删除收到这个消息的 PW 上学习到的 MAC 地址。

在拓扑结构改变时为了能快速移除 MAC 地址，可以使用地址回收消息。地址回收消息分为两类：带有 MAC 地址列表的和不带 MAC 地址列表的。

如果在一条备份链路变为活动状态后，收到带有重学习 MAC 表项的通知消息，PE 将更新 VPLS 实例的 FIB 表中对应的 MAC 表项，并将此消息发送给其他相关的 LDP 会话直连的 PE。如果通知消息中包含空的 MAC 地址 TLV 列表，表示告知 PE 移除指定 VSI 中的所有 MAC 地址（从发送此消息的 PE 处学习到的 MAC 地址除外）。

(3) MAC 地址老化

PE 学习到的与 VC 标签相关但是不再使用的远程 MAC 地址需要有老化机制来移除。老化机制使用了 MAC 地址对应的老化定时器。在接收到报文并处理时，根据报文中的源 MAC 地址，如果这个源地址启动了相应的老化定时器，则 PE 重置该老化定时器。

4. VPLS的环路避免

为了避免环路，一般的二层网络都要求使能 STP（Spanning Tree Protocol，生成树协议）。但是对使用 VPLS 的用户来说，不会感知到 ISP 的网络，因此在私网侧使能 STP 的时候，不能把 ISP 的网络考虑进来。VPLS 中，使用全连接和水平分割转发来避免在 ISP 上使用 VPLS 私网侧的 STP 协议。

VPLS 环路避免的方法如下：

- PE 之间逻辑上全连接（PW 全连接），也就是每个 PE 必须为每一个 VPLS 转发实例创建一棵到该实例下的所有其他 PE 的树。
- 每个 PE 设备必须支持水平分割策略来避免环路，即 PE 不能在具有相同 VSI 的 PW 之间转发报文（由于在同一个 VSI 中每个 PE 直连），也就是说，从公网侧 PW 收到的数据包不再转发到其他 PW 上，只能转发到私网侧。

1.1.2 VPLS报文封装

1. AC上的报文封装

AC 上的报文封装方式由用户的 VSI 接入方式决定。用户接入方式可以分为两种：VLAN 接入和 Ethernet 接入。其含义如下：

- VLAN 接入：CE 发送给 PE 或 PE 发送给 CE 的以太网帧头带有一个 VLAN Tag，该 Tag 是一个服务提供商网络为了区分用户而要求用户压入的“服务定界符”。我们把这个作为服务定界符的 Tag 称为 P-Tag。
- Ethernet 接入：CE 发送给 PE 或 PE 发送给 CE 的以太网帧头中没有服务定界符，如果此时帧头中有 VLAN Tag，则说明它只是用户报文的内部 VLAN Tag，对于 PE 设备没有意义。这种用户内部 VLAN 的 Tag 称为 U-Tag。

至于用户的 VSI 接入方式，可以使用配置的方式来指定。

2. PW上的报文封装

PW 上的报文封装方式也可以分为两种：Ethernet 模式和 VLAN 模式。

- Ethernet 模式下，P-Tag 不在 PW 上传输：对于 CE 侧的报文，如果收到带有服务定界符的报文，则将其去除后再压入 PW 标签和隧道标签后转发；如果收到不带服务定界符的报文，则直接压入 PW 标签和隧道标签后转发。对于 PE 侧的下行报文，根据实际配置选择添加或不添加服务定界符后转发给 CE，但是它不允许重写或移除已经存在的任何 Tag。
- VLAN 模式下，PW 上传输的帧必须带 P-Tag：对于 CE 侧的报文，如果收到带有服务界定符的报文，保留 P-Tag，或者将 P-Tag 改写为对端 PE 期望的 VLAN Tag 或者空 Tag（Tag 值为 0），再压入 PW 标签和隧道标签后转发；如果收到不带服务界定符的报文，则添加一个对端 PE 期望的 VLAN Tag 或空 Tag 后，再压入 PW 标签和隧道标签后转发。对于 PE 侧的下行报文，根据实际配置选择重写、去除或保留服务界定符后转发给 CE。

根据协议规定，缺省情况下 PW 使用 VLAN 模式对报文进行封装。

1.1.3 H-VPLS实现方式

H-VPLS（Hierarchy of VPLS，分层 VPLS），延伸服务提供商的 VPLS 接入范围和降低成本。

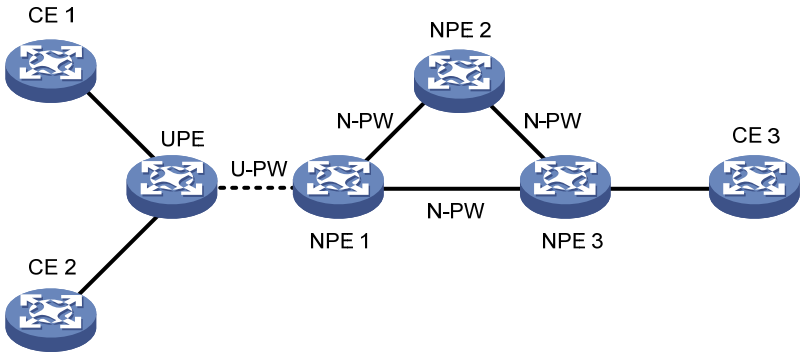
1. H-VPLS接入的优点

- H-VPLS 对 MTU-s（(Multi-Tenant Unit switch，汇聚设备)的要求比较低，层次鲜明，分工明确。
- H-VPLS 能够减少 PE 全连接带来的逻辑复杂度和配置管理的复杂度。

2. H-VPLS的两种接入方式

- H-VPLS 的 LSP 方式接入

图1-3 H-VPLS 的 LSP 方式接入



如 图 1-3 所示，UPE作为汇聚设备MTU-s，它只跟NPE1 建立一条虚链接接入链路U-PW，跟其他所有的对端都不建立虚链接。

数据转发流程如下：

(1) UPE 负责将 CE 上送的报文发给 NPE1，同时打上 U-PW 对应的多路复用分离标记（MPLS 标签）；

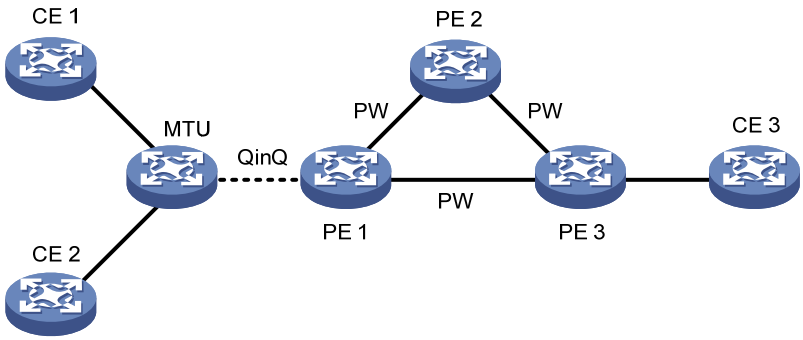
(2) NPE1 收到报文后，先根据多路复用分离标记判断报文所属的 VSI，再根据该报文的目的 MAC 压入 N-PW 对应的多路复用分离标记，然后转发该报文；

(3) NPE1 从 N-PW 侧收到报文后，打上 U-PW 对应的多路复用分离标记将报文发送给 UPE，UPE 再将报文转发给 CE。

如果 CE1 与 CE2 之间的数据交换为本地 CE 之间交换，由于 UPE 本身具有桥接功能，UPE 将直接完成两者间的报文转发，而无需将报文上送给 NPE1。不过对于目的 MAC 未知的第一个数据报文或广播报文，UPE 在将数据通过桥广播到 CE2 的同时，仍然会通过 U-PW 转发给 NPE1，由 NPE1 来完成报文的复制并转发到各个对端 CE。

- H-VPLS 的 QinQ 方式接入

图1-4 H-VPLS 的 QinQ 方式接入



如 图 1-4 所示，MTU为标准的桥接设备，数据转发流程如下：

(1) 在 CE 接入端口使能 QinQ，为收到的报文添加压入 VLAN Tag 作为多路复用分离标记，在 MTU 与 PE1 之间通过 QinQ 隧道将报文透明传输到 PE1 上；

(2) PE1 先根据报文携带 MTU 压入的 VLAN Tag 判断所属的 VSI，再根据该报文的目的 MAC 为其压入 PW 对应的多路复用分离标记（MPLS 标签），然后将其转发；

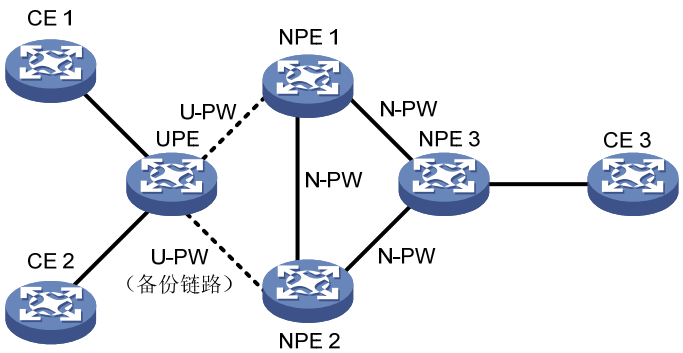
(3) PE1 从 PW 侧收到报文后，根据多路复用分离标记（MPLS 标签）判断报文所属的 VSI，再根据用户报文的目的 MAC 打上 VLAN Tag 通过 QinQ 隧道将报文转发给 MTU，由 MTU 将报文转发给 CE。

如果 CE1 与 CE2 之间的数据交换为本地 CE 之间交换，由于 MTU 本身具有桥接功能，MTU 将直接完成两者间的报文转发，而无需将报文上送给 PE1。不过对于目的 MAC 未知的第一个数据报文或广播报文，MTU 在通过桥广播到 CE2 的同时，仍然会通过 QinQ 隧道转发给 PE1，由 PE1 来完成报文的复制并转发到各个对端 CE。

3. 主备PW切换

UPE与NPE之间只有单条链路连接的方案具有明显的弱点：一旦该接入链路出现故障，汇聚设备连接的所有VPN都将丧失连通性。如 图 1-5 所示，LSP接入方式的H-VPLS提供了链路冗余备份方案。在正常情况下，设备只使用一条链路（主链路）接入。当主链路出现故障时，将启用备用链路继续提供VPN业务。

图1-5 LSP 接入方式的冗余保护



LSP 接入方式的 H-VPLS 根据 LDP 会话状态和 BFD 检测结果等判断主链路是否失效。在以下情况下，将启用备份链路：

- 主 PW 经过的隧道被删除，导致此 PW 的状态变为 down；
- 利用 BFD 协议等链路检测机制，检测到主链路故障；
- 主 PW 对应的对等体间 LDP 会话 down 导致该 PW 删除。

1.2 VPLS配置任务简介

表1-1 VPLS 配置任务简介

配置任务	说明	详细配置
使能L2VPN和MPLS L2VPN	必选	1.3
配置LDP方式的VPLS	根据实际组网情况，选择一种方式的VPLS	1.4
配置BGP方式的VPLS		1.5
绑定VPLS实例	必选	1.6

配置任务	说明	详细配置
配置MAC地址学习功能	可选	1.7
配置MAC地址迁移功能	可选	1.8
配置VPLS实例的属性	可选	1.9
检测PW	可选	1.10

1.3 使能L2VPN和MPLS L2VPN

只有使能了 L2VPN 和 MPLS L2VPN，才允许进行 VPLS 相关配置。

表1-2 使能 L2VPN 和 MPLS L2VPN

操作	命令	说明
进入系统视图	**system-view**	-
使能L2VPN，并进入L2VPN视图	**l2vpn**	必选
使能MPLS L2VPN	**mpls l2vpn**	必选

说明

l2vpn 和 **mpls l2vpn** 命令的详细介绍请参见“MPLS 命令参考”中的“MPLS L2VPN”。

1.4 配置LDP方式的VPLS

1.4.1 配置准备

- 在 MPLS 骨干网设备（PE、P）上配置 IGP，实现骨干网的 IP 连通性。具体配置方法请参见“三层技术-IP 路由配置指导”。
- 在 MPLS 骨干网设备（PE、P）上配置 MPLS 基本能力，用于创建公网的 LSP 隧道。具体配置方法请参见“MPLS 配置指导”中的“MPLS 基础”。
- 在 PE 设备上配置 LDP 远端对等体，用于建立远端 LDP 会话。具体配置方法请参见“MPLS 配置指导”中的“MPLS 基础”。

1.4.2 配置LDP方式的VPLS实例

在创建 LDP 方式的 VPLS 实例时，需要进行以下配置：

(1) 指定全局唯一的 VPLS 实例名，并指明对端发现机制是静态手工配置。

(2) 指明所使用的 PW 信令协议为 LDP。

(3) 指定 VPLS 实例的 ID 号。

(4) 通过 **peer** 命令创建一个实例中包含的 VPLS 对端 PE，并指定：

- 对端 PE 的 IP 地址。
- 到对端 PE 的 PW 的 ID，该 PW ID 必须与对端 PE 上的配置保持一致。
- 对等体类型。如果指定对等体类型为 UPE，则表示该对等体为分层 VPLS 模型中的用户汇聚节点 UPE；如果创建对等体时指定了 **backup-peer** 参数，则表示在 UPE 上创建了一对主备 NPE。汇聚节点 UPE 上只允许用户配置一对主备 NPE。指定的多个远程对等体 NPE 间需要全连接，UPE 与 NPE 之间无需全连接。
- 引用的 PW 模板。通过引用 PW 模板，可以指定 PW 传输模式和到达对端的隧道使用的隧道选用策略。

表1-3 配置 LDP 方式的 VPLS 实例

操作	命令	说明
进入系统视图	**system-view**	-
创建PW模板，并进入PW模板视图	**pw-class** *pw-class-name*	可选 缺省情况下，不存在任何PW模板
配置PW传输模式	**trans-mode** { **ethernet** | **vlan** }	可选 缺省情况下，PW传输模式为VLAN
配置隧道选用策略	**pw-tunnel-policy** *policy-name*	可选 缺省情况下，采用VSI视图下，通过**tnl-policy**命令指定的隧道策略 隧道策略的配置方法，请参见“MPLS配置指导”中的“MPLS L3VPN”
退回系统视图	**quit**	-
创建LDP方式的VPLS实例，并进入VSI视图	**vsi** *vsi-name* **static**	必选
配置使用的PW信令协议为LDP，并进入VSI-LDP视图	**pwsignal ldp**	必选
指定VPLS实例的ID号	**vsi-id** *vsi-id*	必选
创建一个实例中包含的VPLS对端PE，并进入L2VPN peer视图	**peer** *ip-address* [**pw-class** *class-name* | [**pw-id** *pw-id*] [**upe** | **backup-peer** *ip-address* [**backup-pw-id** *pw-id*]]] *	必选
退回VSI-LDP视图	**quit**	-
配置VPLS实例的主备PW回切时间	**dual-npe revertive** [**wtr-time** *wtr-time*]	可选 缺省情况下，主备切换后不进行回切

1.5 配置BGP方式的VPLS

1.5.1 配置准备

- 在 MPLS 骨干网设备（PE、P）上配置 IGP，实现骨干网的 IP 连通性。具体配置方法请参见“三层技术-IP 路由配置指导”。
- 在 MPLS 骨干网设备（PE、P）上配置 MPLS 基本能力，用于创建公网的 LSP 隧道。具体配置方法请参见“MPLS 配置指导”中的“MPLS 基础”。

1.5.2 配置BGP扩展

配置 BGP 方式的 VPLS 之前，需要在 PE 上配置 BGP 参数。具体配置方法请参见“三层技术-IP 路由配置指导”中的“BGP”。

表1-4 配置 BGP 扩展

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPLS地址族视图	**vpls-family**	必选
激活指定对等体	**peer** *peer-address* **enable**	必选 缺省情况下，无激活的对等体

说明

有关 BGP-VPLS 地址族下的配置请参见“MPLS 配置指导”中的“MPLS L3VPN”。

1.5.3 配置BGP方式的VPLS实例

在创建 BGP 方式的 VPLS 实例时，必须指定全局唯一 VPLS 实例名，并指明对端发现机制是自动配置。

在配置 BGP 方式的 VPLS 实例时，需要指明所使用的 PW 信令协议为 BGP。

表1-5 配置 BGP 方式的 VPLS 实例

操作	命令	说明
进入系统视图	**system-view**	-
创建BGP方式的VPLS实例，并进入VSI视图	**vsi** *vsi-name* **auto**	必选
配置使用的PW信令协议为BGP，并进入VSI-BGP视图	**pwsignal bgp**	必选
配置VPLS实例的RD	**route-distinguisher** *route-distinguisher*	必选

操作	命令	说明
将一个指定VPLS实例和一个或多个VPN Target相关联	**vpn-target** *vpn-target*&<1-16> [**both** | **import-extcommunity** | **export-extcommunity**]	必选
创建VPLS实例的站点	**site** *site-id* [**range** *site-range*] [**default-offset** { 0 | 1 }]	必选

1.5.4 复位VPLS的BGP连接

当 BGP 路由策略或协议发生变化后，如果需要通过复位 VPLS 的 BGP 连接，使新的配置生效，请在用户视图下进行以下配置。

表1-6 复位 VPLS 的 BGP 连接

字段	描述
复位VPLS的BGP连接	**reset bgp vpls** { *as-number* | *ip-address* | **all** | **external** | **internal** }

1.6 绑定VPLS实例

配置服务实例与 VPLS 实例绑定后，设备根据二层以太网接口或二层聚合接口上创建的服务实例对该接口接收到的报文进行匹配，与服务实例匹配的报文将通过绑定的 VPLS 实例转发。服务实例提供了多种报文匹配规则（包括接口接收到的所有报文、所有携带 VLAN Tag 的报文和所有不携带 VLAN Tag 的报文等），为报文接入 VPLS 实例提供了更加灵活的方式。VLAN 接口连接的用户属于不同的 VPN 时，可以采用此方式。

1.6.1 配置服务实例与VPLS实例绑定

配置服务实例与 VPLS 实例绑定，需要在二层以太网接口或二层聚合接口上创建服务实例，为服务实例配置报文匹配规则，并与一个 VPLS 实例建立绑定关系。这样，接口收到的符合匹配规则的报文，将进入绑定好的 VPLS 实例进行转发。

表1-7 配置服务实例与 VPLS 实例绑定

操作		命令	说明
进入系统视图		**system-view**	-
进入连接CE的二层以太网接口或二层聚合接口视图	进入二层以太网接口视图	**interface** *interface-type interface-number*	-
	进入二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
创建服务实例，并进入服务实例视图		**service-instance** *service-instance-id*	必选 缺省情况下，不存在任何服务实例

操作	命令	说明
配置报文匹配规则	**encapsulation** { **s-vid** *vlan-id* [**only-tagged**] | **port-based** | **tagged** | **untagged** }	必选 缺省情况下，不存在任何报文匹配规则
将指定接口下的服务实例与VPLS实例进行关联	**xconnect vsi** *vsi-name* [**access-mode** { **ethernet** | **vlan** }] *	必选 缺省情况下，服务实例没有与VPLS实例关联

1.7 配置MAC地址学习功能

操作	命令	说明
进入系统视图	**system-view**	-
进入VSI视图	**vsi** *vsi-name*	-
开启或关闭MAC地址学习功能	**mac-learning** { **enable** | **disable** }	可选 缺省情况下，MAC地址学习功能是开启状态

1.8 配置MAC地址迁移功能

PE 设备通过 VPLS 实例内的 MAC 地址学习，维护一张桥 MAC 地址表。在该 MAC 地址表中，MAC 地址表项会随着 MAC 地址所对应报文的入端口变化而变化，称为 MAC 地址迁移功能。

- 开启此功能后，MAC 地址所对应的报文从其它端口被接收到时，此 MAC 地址仍可被学习到该端口所对应的 VPLS 实例中。
- 禁用 MAC 地址迁移功能后，在 MAC 地址表项老化时间内，MAC 地址所对应的报文从其它端口被接收到时，MAC 地址表项中的入端口不会改变，即所属的 VPLS 实例也不会改变。

通过关闭 MAC 地址迁移功能，可对 VPLS 实例的接入用户进行控制，防止非法用户仿冒 MAC 地址接入网络，提高了网络的安全性。

表1-8 配置 MAC 地址迁移功能

操作	命令	说明
进入系统视图	**system-view**	-
进入VSI视图	**vsi** *vsi-name*	-
开启MAC迁移功能	**mac-move enable**	可选 缺省情况下，MAC迁移功能为开启状态

1.9 配置VPLS实例的属性

表1-9 配置 VPLS 实例的属性

操作	命令	说明
进入系统视图	**system-view**	-
进入VSI视图	**vsi** *vsi-name*	-
设置指定VPLS实例的封装类型	**encapsulation** { **bgp-vpls** | **ethernet** | **vlan** }	可选 缺省情况下，VPLS实例的封装类型为**vlan**，对应VSI的PW封装类型为Tagged模式
设置指定VPLS实例的描述信息	**description** *text*	可选 缺省情况下，不设置VPLS实例的描述信息
关闭指定VPLS实例的VPLS服务	**shutdown**	可选 缺省情况下，VPLS实例的VPLS服务是开启状态
配置指定VPLS实例的隧道策略	**tnl-policy** *tunnel-policy-name*	可选 缺省情况下，未指定VPLS实例选用的隧道策略，采用缺省的隧道策略，即按照LSP隧道—> CR-LSP隧道的优先级顺序选择隧道，负载分担条数为1 隧道策略的配置方法，请参见“MPLS配置指导”中的“MPLS L3VPN”

1.10 检测PW

在 VPLS 网络中，通过 MPLS LSP Ping 功能，可以对 PW 的可达性进行检测，并提供必要的诊断信息，以便对 PW 的故障进行定位。

MPLS LSP Ping 功能采取的方法是：在本地 PE 设备上为 MPLS Echo Request 报文压入待检测的 PW 对应的标签，使得 MPLS Echo Request 报文沿着 PW 转发，本地 PE 设备根据收到的对端 PE 设备的应答报文，判断 PW 的可达性。

表1-10 利用 MPLS LSP Ping 功能检测 PW

操作	命令	说明
通过MPLS LSP Ping检测PW的可达性	**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*	必选 可在任意视图下执行本命令

说明

MPLS LSP Ping 只能用来检测 LDP 方式建立的 PW 的可达性。

1.11 VPLS显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 VPLS 的运行情况，通过查看显示信息验证配置的效果。

在用户视图下执行 **reset** 命令可以清除 VPLS 实例的 MAC 地址表信息。

表1-11 VPLS 显示和维护

字段	描述
显示BGP路由表中的VPLS信息	**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*]
显示VPLS实例的MAC地址表信息	**display mac-address vsi** [*vsi-name*] [**blackhole** | **dynamic** | **static**] [**count**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示VPLS连接信息	**display vpls connection** [**bgp** | **ldp** | **static** | **vsi** *vsi-name*] [**block** | **down** | **up**] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示VPLS实例的AC表项信息	**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*]
显示VPLS实例的PW表项信息	**display mpls l2vpn fib pw vpls** [**vsi** *vsi-name* [**link** *link-id*]] [**slot** *slot-number*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示端口上指定服务实例的信息	**display service-instance interface** *interface-type interface-number* [**service-instance** *instance-id*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示VPLS实例的信息	**display vsi** [*vsi-name*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示VPLS远程连接信息	**display vsi remote** { **bgp** | **ldp** } [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示PW模板的信息	**display pw-class** [*pw-class-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
清除指定VSI的MAC地址转发表	**reset mac-address vsi** [*vsi-name*]

1.12 VPLS典型配置举例

1.12.1 服务实例与VPLS实例绑定配置举例

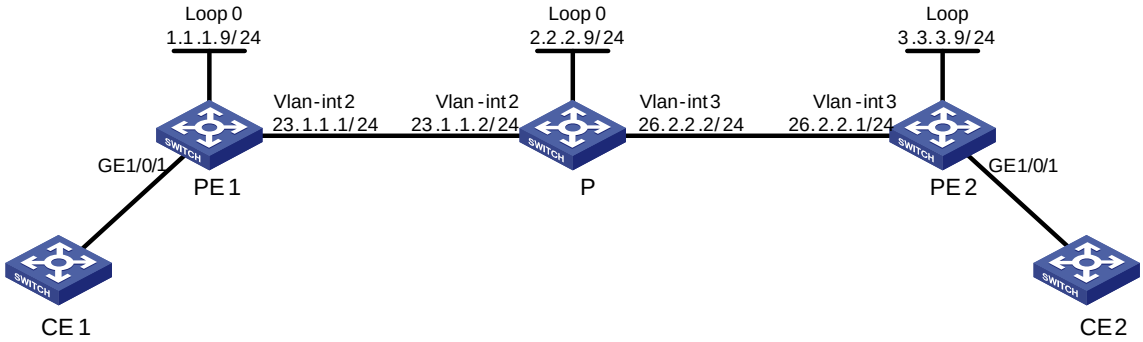
1. 组网需求

- CE 1、CE 2 分别通过 VLAN 方式接入 PE 1 和 PE 2。
- 配置 VPLS 实例 aaa 为 LDP 方式（Martini 方式），bbb 为 BGP 方式（Kompella 方式），AS 号为 100。
- PE 1 和 PE 2 通过服务实例将报文与 VPLS 实例绑定：服务实例 1000 用来匹配接口 GigabitEthernet1/0/1 接收到的 VLAN Tag 为 100 的报文，并通过 VPLS 实例 aaa 转发；服务

实例 2000 用来匹配接口 GigabitEthernet1/0/1 接收到的 VLAN Tag 为 200 的报文，并通过 VPLS 实例 bbb 转发。

2. 组网图

图1-6 服务实例与 VPLS 实例绑定配置举例组网图



3. 配置步骤

(1) PE 1 的配置

```
<Sysname> system-view
[Sysname] sysname PE1
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 1.1.1.9 32
[PE1-LoopBack0] quit
```

配置 LSR ID，全局使能 MPLS。

```
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls
[PE1-mpls] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

全局使能 LDP。

```
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

配置 PE 1 与 PE 2 建立 LDP 远程会话。

```
[PE1] mpls ldp remote-peer 1
[PE1-mpls-ldp-remote-1] remote-ip 3.3.3.9
[PE1-mpls-ldp-remote-1] quit
```

配置连接 P 的接口 Vlan-interface2，在此接口上使能 LDP。

```
[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
```

在 PE 1 上运行 OSPF，用于建立 LSP。

```
[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
```

配置 BGP 扩展。

```
[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
```

配置 LDP 方式下的 VPLS 实例 aaa 基本属性。

```
[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
```

配置 BGP 方式下的 VPLS 实例 bbb 基本属性。

```
[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 1
[PE1-vsi-bbb-bgp] quit
[PE1-vsi-bbb] quit
```

在接入 CE 1 的接口 GigabitEthernet1/0/1 上创建服务实例，并绑定 L2VPN。

```
[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) 配置 P

```
<Sysname> system-view
[Sysname] sysname P
[P] interface loopback 0
[P-LoopBack0] ip address 2.2.2.9 32
[P-LoopBack0] quit
```

配置 LSR ID，全局使能 MPLS。

```
[P] mpls lsr-id 2.2.2.9
[P] mpls
[P-mpls] quit
```
全局使能 LDP。
```
[P] mpls ldp
[P-mpls-ldp] quit
```
配置连接 PE 1 的接口 Vlan-interface2，在此接口上使能 LDP。
```
[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
```
配置连接 PE 2 的接口 Vlan-interface3，在此接口上使能 LDP。
```
[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
```
在 P 上运行 OSPF，用于建立 LSP。
```
[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
[P-ospf-1] quit
```
(3) PE 2 的配置
```
<Sysname> system-view
[Sysname] sysname PE2
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 3.3.3.9 32
[PE2-LoopBack0] quit
```
配置 LSR ID，全局使能 MPLS。
```
[PE2] mpls lsr-id 3.3.3.9
[PE2] mpls
[PE2-mpls] quit
```
使能 L2VPN 和 MPLS L2VPN。
```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```
全局使能 LDP。
```
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```
配置 PE 2 与 PE 1 建立 LDP 远程会话。
```
[PE2] mpls ldp remote-peer 2
[PE2-mpls-ldp-remote-2] remote-ip 1.1.1.9
```

```
[PE2-mpls-ldp-remote-2] quit
```
配置连接 P 的接口 Vlan-interface3，在此接口上使能 LDP。
```
[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
```
在 PE 2 上运行 OSPF，用于建立 LSP。
```
[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
```
配置 BGP 扩展。
```
[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
```
配置 LDP 方式下的 VPLS 实例 aaa 基本属性。
```
[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
```
配置 BGP 方式下的 VPLS 实例 bbb 基本属性。
```
[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 2
[PE2-vsi-bbb-bgp] quit
[PE2-vsi-bbb] quit
```
在接入 CE 2 的接口 GigabitEthernet1/0/1 上创建服务实例，并绑定 L2VPN。
```
[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
```

(4) 配置完成后的检验

完成上述配置后，在各个 PE 上执行 **display vpls connection** 命令，可以看到建立了 PW 连接，且状态为 up。

```
[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
```

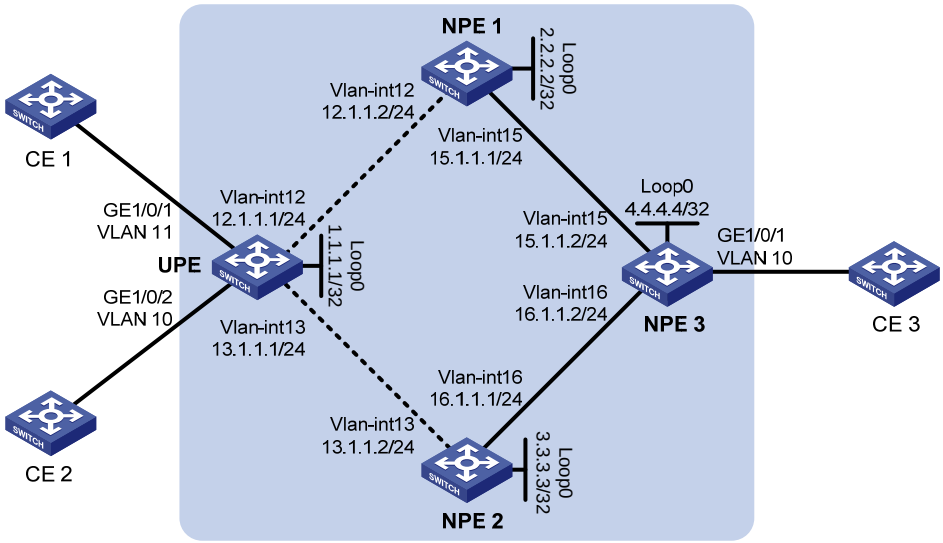
1.12.2 H-VPLS备份链路配置举例

1. 组网需求

- CE 1、CE 2 分别通过 VLAN 方式接入 UPE；
- UPE 与 NPE1 和 NPE2 建立 PW 连接——U-PW，NPE2 作为备份链路；
- NPE1、NPE2 与 NPE3 之间建立 PW 连接——N-PW，CE3 通过 NPE3 连接到网络；
- UPE 与 NPE1 和 NPE2 设备之间以接口 Vlan-interface12 和 Vlan-interface13 连接；
- NPE1 与 NPE3 之间以接口 Vlan-interface15 连接；NPE2 与 NPE3 之间以接口 Vlan-interface16 连接；
- 配置 VPLS 实例支持 H-VPLS 组网方式。

2. 组网图

图1-7 配置 H-VPLS 备份链路组网图



3. 配置步骤

(1) 在 MPLS 骨干网络上配置 IGP 协议，此例选择 OSPF，具体配置略。

(2) UPE 的配置

配置 MPLS 基本能力。

```
<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
```

配置与 NPE1 相连接口的 IP 地址，并使能 MPLS 和 MPLS LDP。

```
[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
```

配置与 NPE2 相连接口的 IP 地址，并使能 MPLS 和 MPLS LDP。

```
[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
```

配置与 NPE1 的远端 LDP 会话。

```
[UPE] mpls ldp remote-peer 1
[UPE-mpls-remote-1] remote-ip 2.2.2.2
[UPE-mpls-remote-1] quit
```

配置与 NPE2 的远端 LDP 会话。

```
[UPE] mpls ldp remote-peer 2
[UPE-mpls-remote-1] remote-ip 3.3.3.3
[UPE-mpls-remote-1] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[UPE] l2vpn
[UPE-l2vpn] mpls l2vpn
[UPE-l2vpn] quit
```

配置 LDP 方式下 VPLS 实例 aaa 的基本属性。

```
[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
```

在接入 CE 1 的接口 GigabitEthernet1/0/1 上创建服务实例，并绑定 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
```

在接入 CE 2 的接口 GigabitEthernet1/0/2 上创建服务实例，并绑定 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) NPE1 的配置

配置 MPLS 基本能力。

```
<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
```

配置与 UPE 相连接口的 IP 地址，并使能 MPLS 和 MPLS LDP。

```
[NPE1] interface vlan-interface 12
[NPE1-Vlan-interface12] ip address 12.1.1.2 24
```

```
[NPE1-Vlan-interface12] mpls
[NPE1-Vlan-interface12] mpls ldp
[NPE1-Vlan-interface12] quit
```

配置与 NPE3 相连接口的 IP 地址，并使能 MPLS 和 MPLS LDP。

```
[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
```

配置与 UPE 的远端 LDP 会话。

```
[NPE1] mpls ldp remote-peer 2
[NPE1-mpls-remote-2] remote-ip 1.1.1.1
[NPE1-mpls-remote-2] quit
```

配置与 NPE3 的远端 LDP 会话。

```
[NPE1] mpls ldp remote-peer 3
[NPE1-mpls-remote-3] remote-ip 4.4.4.4
[NPE1-mpls-remote-3] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[NPE1] l2vpn
[NPE1-l2vpn] mpls l2vpn
[NPE1-l2vpn] quit
```

配置 LDP 方式下 VPLS 实例 aaa 的基本属性。

```
[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
```

NPE2 和 NPE1 的配置相似，配置过程略。

(4) NPE3 的配置

配置 MPLS 基本能力。

```
<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
```

配置与 NPE1 相连接口的 IP 地址，并使能 MPLS 和 MPLS LDP。

```
[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
```

配置与 NPE2 相连接口的 IP 地址，并使能 MPLS 和 MPLS LDP。

```
[NPE3] interface vlan-interface 16
[NPE3-Vlan-interface16] ip address 16.1.1.2 255.255.255.0
[NPE3-Vlan-interface16] mpls
[NPE3-Vlan-interface16] mpls ldp
[NPE3-Vlan-interface16] quit
```

配置远端 LDP 会话。

```
[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
```

使能 L2VPN 和 MPLS L2VPN。

```
[NPE3] l2vpn
[NPE3-l2vpn] mpls l2vpn
[NPE3-l2vpn] quit
```

配置 LDP 方式下 VPLS 实例 aaa 的基本属性。

```
[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
```

在接入 CE 3 的接口 GigabitEthernet1/0/1 上创建服务实例，并绑定 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
```

(5) 配置完成后的检验

完成上述配置后，在各个 PE 上执行 **display vpls connection** 命令，可以看到 PW 连接，状态为 up。

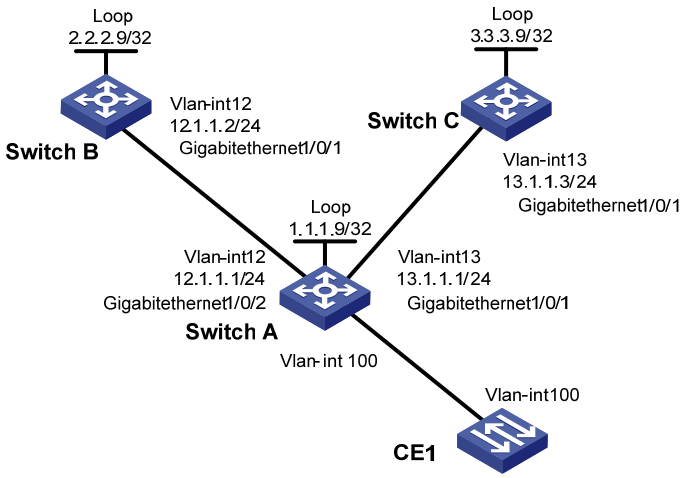
1.12.3 H-VPLS组网中通过BFD检测主链路故障配置举例

1. 组网需求

- H-VPLS 组网，Switch A 作为 UPE 设备，Switch B 作为主 NPE 设备、Switch C 作为备份 NPE 设备，在双方直连口上使能 MPLS 应用，在设备上运行 OSPF，网络层相互可达。
- 在断开 Switch A 和 Switch B 之间的链路后，BFD 能够快速检测并通告 MPLS LDP 协议，从而快速进行主备 PW 间的切换。

2. 组网图

图1-8 H-VPLS 组网中通过 BFD 检测主链路故障配置组网图



3. 配置步骤

(1) 配置 MPLS 基本功能

配置 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
```

配置 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
```

配置 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) 配置各交换机接口地址

配置 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
```

配置 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
```

配置 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) 配置 OSPF 基本功能

配置 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
```

配置 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
```

配置 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) 配置各交换机 VSI 实例

配置 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
```

配置 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
```

配置 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) 检查配置结果

通过 **display bfd session verbose** 显示 Switch A 的 BFD 邻居详细信息。

```
<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
```

通过 **display vpls connection vsi vpna** 显示 Switch A 连接 Switch B 的路径为 up 状态。

```
<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
```

断开 Switch A 和 Switch B 之间的链路。使用 **display vpls connection vsi vpna** 可以看到 3.3.3.9 这条路径的状态为 up。

```
<SwitchA> display vpls connection vsi vpna

Total 1 connection(s),
connection(s): 1 up, 0 block, 0 down

VSI Name: vpna                              Signaling: ldp
VsiID      VsiType           PeerAddr      InLabel OutLabel LinkID  VCState
100        vlan              3.3.3.9       134216  140476   2       up
```

1.13 常见配置错误举例

1.13.1 PW状态不是up

1. 故障现象

VPLS 的 PW 状态不是 up。

2. 故障分析

- 公网 LSP 隧道没有建立（两端）。
- 扩展会话工作不正常。
- 连接私网的接口没有绑定对应的 VPLS 实例，或私网接口状态为 down。

3. 处理过程

- 查看两端 PE 设备的路由表，判断 PE 之间是否存在可达的路由，ping 对端环回端口是否可以 ping 通，LDP 会话是否正常。

- 检查两端的扩展会话配置命令是否有遗漏，配置是否正确。
- 通过 **display interface** 命令查看私网接口的状态，确保私网接口 up。
- 通过 **display current-configuration** 命令查看当前配置，确认对等体之间的 PW-ID 和传输模式是否一致。

目 录

1 MPLS L2VPN ····· 1-1
1.1 MPLS L2VPN简介 ····· 1-1
1.1.1 MPLS L2VPN概述 ····· 1-1
1.1.2 MPLS L2VPN的基本概念 ····· 1-1
1.1.3 MPLS L2VPN基本网络架构 ····· 1-2
1.1.4 MPLS L2VPN远程连接的工作机制 ····· 1-3
1.1.5 MPLS L2VPN的实现方式 ····· 1-4
1.1.6 VC封装类型 ····· 1-9
1.2 MPLS L2VPN配置任务简介 ····· 1-10
1.3 配置MPLS L2VPN基本功能 ····· 1-11
1.4 配置连接CE的接口 ····· 1-11
1.4.1 配置CE侧接口的封装类型为Ethernet ····· 1-11
1.4.2 配置CE侧接口的封装类型为VLAN ····· 1-11
1.5 配置CCC方式MPLS L2VPN ····· 1-12
1.6 配置SVC方式MPLS L2VPN ····· 1-12
1.7 配置Martini方式MPLS L2VPN ····· 1-13
1.7.1 配置远端对等体 ····· 1-14
1.7.2 在三层接口上创建Martini方式的VC连接 ····· 1-14
1.7.3 配置在服务实例上创建Martini方式的VC连接 ····· 1-15
1.7.4 检测VC ····· 1-16
1.8 配置Kompella方式MPLS L2VPN ····· 1-16
1.8.1 配置BGP的L2VPN能力 ····· 1-17
1.8.2 创建并配置MPLS L2VPN ····· 1-17
1.8.3 配置CE连接 ····· 1-18
1.9 MPLS L2VPN显示和维护 ····· 1-20
1.10 MPLS L2VPN典型配置举例 ····· 1-20
1.10.1 配置CCC远程连接示例 ····· 1-20
1.10.2 配置SVC方式MPLS L2VPN示例 ····· 1-24
1.10.3 配置Martini方式MPLS L2VPN示例 ····· 1-27
1.10.4 配置Kompella方式MPLS L2VPN示例 ····· 1-31
1.10.5 配置在服务实例上创建VC连接示例 ····· 1-34
1.11 常见配置错误举例 ····· 1-38
1.11.1 配置后不能Ping通 ····· 1-38

1 MPLS L2VPN

说明

- MPLS L2VPN 既可以提供点到点的连接，也可以提供多点间的连接。本章只介绍提供点到点连接的 MPLS L2VPN 技术。提供多点间连接的 MPLS L2VPN 技术，请参见“MPLS 配置指导”中的“VPLS”。
- 本系列交换机中 S5500-28SC-HI 和 S5500-52SC-HI 不支持 MPLS L2VPN 功能。

1.1 MPLS L2VPN简介

1.1.1 MPLS L2VPN概述

MPLS L2VPN 是基于 MPLS（Multiprotocol Label Switching，多协议标签交换）的二层 VPN（Virtual Private Network，虚拟专用网络）技术，它利用 MPLS 技术在不同用户节点间建立二层连接。

采用 MPLS L2VPN 技术，运营商可以在统一的 MPLS 或 IP 骨干网上透明传输不同数据链路层（包括 VLAN、Ethernet 等）的二层数据，使得数据链路层业务可以跨越 MPLS 或 IP 骨干网传递。

从用户的角度来看，MPLS 或 IP 骨干网是一个二层交换网络，用户感知不到 MPLS 或 IP 骨干网的存在。以 Ethernet 类型的用户网络为例，通过 MPLS L2VPN 连接的 Ethernet 节点感知不到 MPLS 或 IP 骨干网的存在，就好像节点之间直接通过以太网链路相连。

MPLS L2VPN 具有以下优势：

- 可扩展性强：MPLS L2VPN 只建立二层连接关系，不引入和管理用户的路由信息。这大大减轻了服务提供商网络边缘设备甚至整个服务提供商网络的负担，使服务提供商能支持更多的 VPN 和接入更多的用户。
- 可靠性和私网路由的安全性得到保证：由于不引入用户的路由信息，MPLS L2VPN 不能获得和处理用户路由，保证了用户 VPN 路由的安全。
- 支持多种网络层协议：包括 IP、IPX、SNA 等。

1.1.2 MPLS L2VPN的基本概念

- CE（Customer Edge，用户网络边缘）设备

直接与服务提供商网络相连的用户网络侧设备。CE 可以是网络设备（如路由器、交换机），也可以是一台主机。CE“感知”不到 VPN 的存在，不需要支持 MPLS。

- PE（Provider Edge，服务提供商网络边缘）设备

与 CE 相连的服务提供商网络侧设备。PE 主要负责 VPN 业务的接入。它完成报文从用户网络到公网隧道、从公网隧道到用户网络的映射与转发。

在 MPLS 网络中，VPN 相关的所有处理都发生在 PE 上。

- AC（Attachment Circuit，接入电路）

连接 CE 与 PE 的链路。

- VC（Virtual Circuit，虚电路）

VC 又称为 PW（Pseudowire，伪线），是将两个 PE 上的 AC 连接起来的一条虚拟的双向连接。MPLS VC 由一对方向相反的单向 LSP 构成。

- 隧道（Tunnel）

又称为公网隧道，是穿越 MPLS 或 IP 骨干网、用来承载 VC 的连接。隧道可以是 LSP、MPLS TE、GRE 隧道等。

- P（Provider，服务提供商网络）设备

不与 CE 直接相连。P 设备只需沿着隧道将用户报文从一端 PE 转发到另一端 PE。

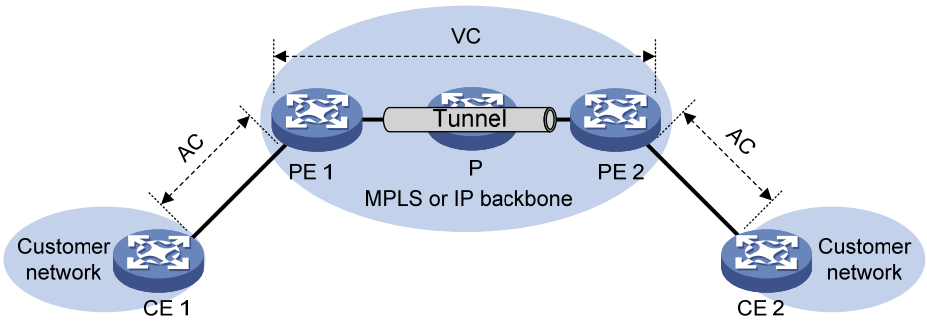
1.1.3 MPLS L2VPN基本网络架构

MPLS L2VPN 的网络架构分为远程连接和本地连接两种。

1. 远程连接网络架构

如 图 1-1 所示，MPLS L2VPN的远程连接组网是指跨越MPLS或IP骨干网络连接两端的用户二层网络。

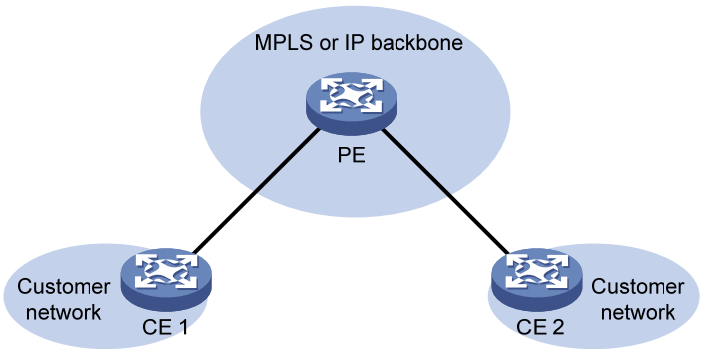
图1-1 远程连接组网图



2. 本地连接网络架构

如 图 1-2 所示，本地连接是指两个用户二层网络通过两个CE连接到同一个PE上，用户网络直接通过PE进行用户报文的交换。PE的作用类似于二层交换机。

图1-2 本地连接组网图



说明

本系列交换机不支持本地连接架构。

1.1.4 MPLS L2VPN远程连接的工作机制

1. 远程连接建立过程

为了在 CE 之间建立 MPLS L2VPN 的远程连接，实现通过 MPLS L2VPN 技术跨越骨干网传递二层用户报文，需要完成以下工作：

- 建立公网隧道，以便将用户网络报文从本地 PE 转发到对端 PE。

  公网隧道可以是 LSP、MPLS TE、GRE 隧道等。LSP 隧道的详细介绍，请参见“MPLS 配置指导”中的“MPLS 基础”；MPLS TE 隧道的详细介绍，请参见“MPLS 配置指导”中的“MPLS TE”；GRE 隧道的详细介绍，请参见“三层技术-IP 业务”中的“GRE”。

  如果在两端 PE 之间存在多条公网隧道，可以配置隧道策略，根据该隧道策略选择承载 VC 的公网隧道。隧道策略的详细介绍，请参见“MPLS 配置指导”中的“MPLS L3VPN”。
- 建立 VC，以便通过 VC 标签标识报文所属的用户网络。

  建立 VC，是指两端的 PE 设备分别为对方分配 VC 标签，建立方向相反的一对单向 LSP，

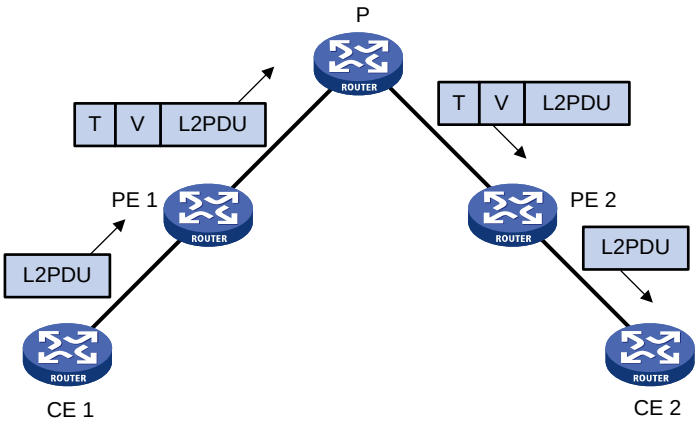
  根据VC建立方式的不同，MPLS L2VPN的实现方式分为CCC（Circuit Cross Connect，电路交叉连接）、SVC（Static Virtual Circuit，静态虚拟电路）、Martini和Kompella四种，详细介绍请参见“MPLS L2VPN的实现方式”。
- 建立 AC，并将 AC 和 VC 绑定，以便确定接收到的用户网络报文通过哪条 VC 转发。
  - 建立 AC：即在 PE 和 CE 上配置链路层协议，以便在 PE 和 CE 之间建立链路层连接。
  - AC 和 VC 绑定：AC 和 VC 绑定后，从该 AC 接收到的报文将通过绑定的 VC 转发，从绑定的 VC 上接收到的报文将转发给该 AC。在本系列交换机上，可以通过将接口上的服务实例与 VC 绑定的方式，来实现 AC 与 VC 的绑定。

2. 远程连接的报文转发过程

MPLS L2VPN 通过在二层用户报文外封装 Tunnel 标记和 VC 标签，实现用户报文在 MPLS 或 IP 骨干网中的透明传送。

- Tunnel 标记用于将报文从一个 PE 传递到另一个 PE。如果采用 LSP 或 MPLS TE 隧道，则 Tunnel 标记为 MPLS 标签；如果采用 GRE 隧道，则 Tunnel 标记为 GRE 头和传输协议的报文头。
- VC 标签用于区分连接不同 CE 的二层连接。接收方 PE 根据 VC 标签决定将报文转发给哪个 CE。

图1-3 远程连接的报文转发过程



(1) L2PDU是链路层报文，PDU即Protocol Data Unit，协议数据单元

(2) T是Tunnel标记；V是VC标签

如 图 1-3 所示，远程连接建立后，MPLS L2VPN的报文转发过程为：

(1) PE 1 接收到 CE 1 发送的二层用户报文后，根据 PE 1 与 CE 1 之间的 AC 所绑定的 VC 为报文封装 VC 标签，并查找对应的公网隧道，为其封装 Tunnel 标记后将封装后的报文转发给 P 设备。

(2) P 设备根据外层的 Tunnel 标记将报文转发给 PE 2。

(3) PE 2 从公网隧道接收到报文后，根据 VC 标签判断报文所属的 VC。PE 2 删除 Tunnel 标记和 VC 标签，还原二层用户报文后，将其转发给与该 VC 绑定的 AC，即转发给 CE 2。

说明

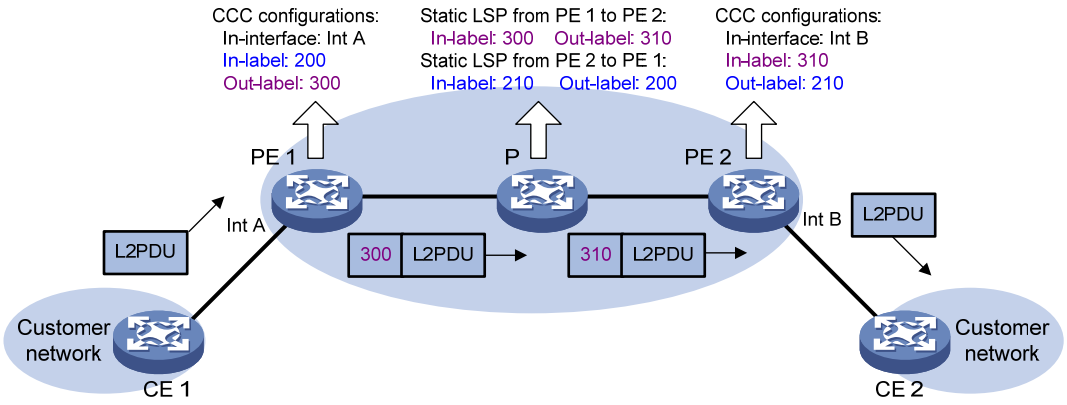
上文描述的连接建立和报文转发过程不适用于CCC方式MPLS L2VPN。CCC方式MPLS L2VPN的详细介绍，请参见“1.1.5 1. CCC方式MPLS L2VPN”。

1.1.5 MPLS L2VPN的实现方式

1. CCC方式MPLS L2VPN

CCC 方式通过建立方向相反的两条静态 LSP，并通过配置将静态 LSP 与 AC 绑定的方法来建立 VC 连接。采用 CCC 方式建立的 VC 连接称为 CCC 连接。

图1-4 CCC 方式示意图



如 图 1-4 所示，完成图中所示的各项配置后，将建立从PE 1 到PE 2 和从PE 2 到PE 1 的两条静态LSP，在PE 1 上这两条静态LSP与Interface A绑定，在PE 2 上与Interface B绑定，至此，一条CCC远程连接成功建立。

以 CE 1 发送报文给 CE 2 为例，CCC 方式的报文转发过程为：

(1) PE 1 接收到 CE 1 发送的报文后，由于 Interface A 与静态 LSP 绑定，因此 PE 1 为其添加从 PE 1 到 PE 2 的静态 LSP 的标签 300。

(2) P 接收到报文后，查找标签转发表，将入标签 300 替换为静态 LSP 的出标签 310。

(3) 由于静态 LSP 与 Interface B 绑定，因此，PE 2 接收到报文后，删除标签，将原始报文通过 Interface B 转发给 CE 2。

与其他方式的 MPLS L2VPN 不同，CCC 采用一层标签传送用户报文，只有与该静态 LSP 绑定的 AC 才能通过该静态 LSP 转发报文。因此，CCC 对 LSP 的使用是独占性的。LSP 只用于传递这个 CCC 连接的数据，不能用于其他连接，也不能用于 MPLS L3VPN。

2. SVC方式MPLS L2VPN

SVC 是一种静态的 MPLS L2VPN 实现方式。SVC 方式采用两层标签传递用户报文，内层的 VC 标签是在两端的 PE 上静态配置的，不需要使用信令协议传递。

3. Martini方式MPLS L2VPN

Martini 方式采用两层标签传递用户报文，以 LDP（Label Distribution Protocol，标签分发协议）为信令协议分发内层的 VC 标签。

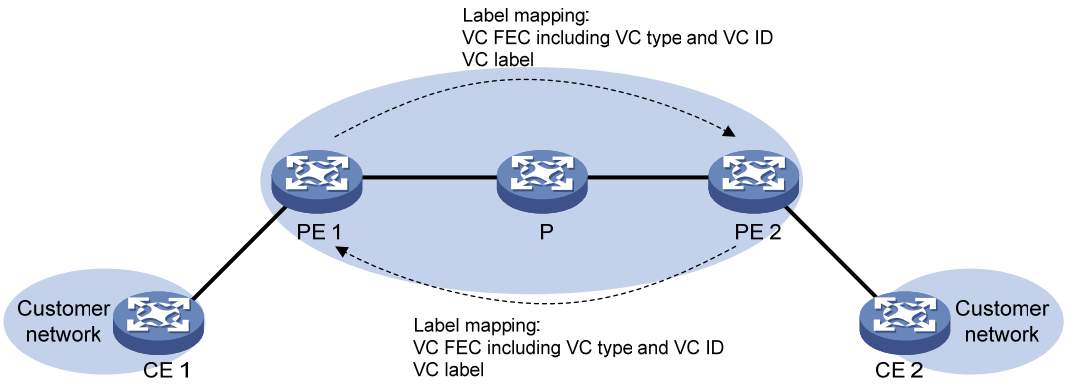
为了在 PE 之间交换 VC 标签，Martini 方式对 LDP 进行了扩展，增加了 VC FEC（Forwarding Equivalence Class，转发等价类）的 FEC 类型。该类型的 FEC 包含以下信息：

- VC type：VC的封装类型。详细介绍请参见“1.1.6 VC封装类型”。
- VC ID：PE 上为 VC 指定的连接标识。

VC type+VC ID 唯一标识了一个 VC。在一台 PE 上，相同 VC type 的所有 VC，其 VC ID 必须唯一。

如 图 1-5 所示，两端PE分别发送标签映射消息，通过VC FEC（包含VC type、VC ID）和VC label将用于标识VC的信息和对应的VC标签发送给对端，即可实现VC标签的发布，从而在PE之间建立VC连接。

图1-5 Martini 方式标签分发过程



4. Kompella方式MPLS L2VPN

Kompella 方式采用两层标签传递用户报文，以扩展的 BGP 为信令协议分发内层的 VC 标签。

与其他方式不同，Kompella 方式引入了 VPN 的概念，属于同一个 VPN 的 CE 之间可以建立连接，属于不同 VPN 的 CE 之间不能建立连接。

(1) 基本概念

- CE ID

Kompella 方式在 VPN 内部对 CE 进行编号，通过 CE ID 唯一标识 VPN 内的一个 CE。不同 VPN 内 CE 编号可以相同。

- RD（Route Distinguisher，路由标识符）

为了区分不同 VPN 内编号相同的 CE，Kompella 方式定义了 RD。在 CE ID 前增加 RD，通过 RD+CE ID 可以唯一标识网络中的一个 CE。

- VPN target

Kompella 方式使用 BGP 的 VPN target（也称为 Route target）属性来区分不同的 VPN，确保属于同一个 VPN 的 CE 之间可以建立连接，属于不同 VPN 的 CE 之间不能建立连接。

PE 上的 VPN target 属性分为以下两种：

  - Export target 属性：本地 PE 在通过 BGP 的 Update 消息将 L2VPN 信息（如本地 CE ID、RD 等）发送给对端 PE 时，将 Update 消息中携带的 VPN target 属性设置为 Export target。
  - Import target 属性：PE 收到其它 PE 发布的 Update 消息时，检查消息中携带的 VPN target 属性，只有当此属性与 PE 上设置的 Import target 匹配时，才会接收该消息中的 L2VPN 信息。

也就是说，VPN target 属性定义了本地发送的 L2VPN 信息可以为哪些 PE 所接收，PE 可以接收哪些对端 PE 发送来的 L2VPN 信息。

(2) 标签块

与 Martini 方式不同，Kompella 方式并不是将本地 PE 分配的 VC 标签通过消息直接发送给对端 PE，而是采用标签块的方式，一次为多个连接分配标签。PE 将自己分配的标签块通告给同一个 VPN 内的所有 PE，每个 PE 都根据其他 PE 通告的标签块计算出 VC 标签。

标签块包含以下参数：

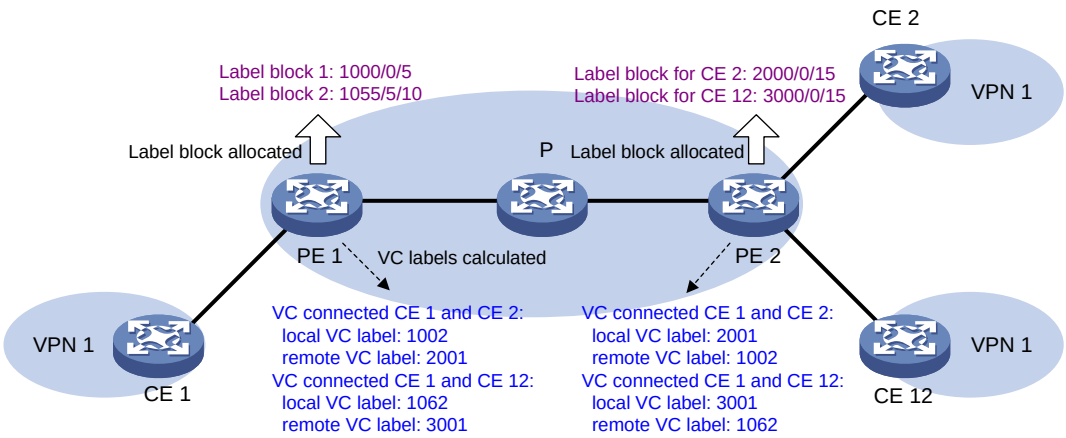
- LB（Label Base，初始标签）：标签块的标签初始值。该值为 PE 设备自动选取，不可手动修改。
- LR（Label Range，标签范围）：标签块包含的标签数目。LB 和 LR 确定了标签块中包含哪些标签。例如，LB 为 1000、LR 为 5，则该标签块包含的标签为 1000～1004。
- LO（Label-block Offset，标签块偏移）：VPN 网络中 CE 的数量增加，原有的标签块大小无法满足要求时，PE 无需撤销原有的标签块，只要在原有标签块的基础上再分配一个新的标签块就可以扩大标签范围，满足扩展需要。在这种情况下，PE 通过 LO 来标识某个标签块在所有为 CE 分配的标签块中的位置，并根据 LO 来判断从哪个标签块中分配标签。LO 的取值为之前分配的所有标签块大小的总合。例如，PE 为 CE 分配的第一个标签块的 LR 为 10、LO 为 0，则第二个标签块的 LO 为 10；如果第二个标签块的 LR 为 20，则第三个标签块的 LO 为 30。

为了方便描述，下文利用 LB/LO/LR 来描述一个标签块，即 LB 为 1000、LO 为 10、LR 为 5 的标签块可以表示为 1000/10/5。

采用标签块的方式允许用户为 VPN 分配一些额外的标签，留待以后使用。这样短期来看会造成标签资源的浪费，但是却带来一个很大的好处，即可以减少 VPN 部署和扩容时的配置工作量。假设一个企业的 VPN 包括 10 个 CE，但是考虑到企业会扩展业务，将来可能会有 20 个 CE。这样可以把 LR 设置为 20，系统会预先为未来的 10 个 CE 分配标签。以后 VPN 添加 CE 节点时，配置的修改仅限于与新 CE 直接相连的 PE，其他 PE 不需要作任何修改。这使得 VPN 的扩容变得非常简单。

(3) VC 标签的计算方法

图1-6 VC 标签计算方式



如 图 1-6 所示，以计算连接CE 1 和CE 12 的VC的VC标签为例，Kompella方式中VC标签的计算方法为：

- PE 1 计算本地为该 VC 分配的 VC 标签：
  a. 将对端 CE（CE 12）的 ID（12）与 PE 1 分配的两个标签块进行比较，如果存在满足 LO<=CE ID<LO+LR 的标签块，则从该标签块中分配标签。在本例中，标签块 2（1055/5/10）满足 LO<=CE ID<LO+LR（即 5<=12<5+10），故从该标签块中为此 VC 分配标签。
  b. 本地为此 VC 分配的标签值为：LB+CE ID-LO，即 1055+12-5=1062。
- PE 1 计算对端 PE（PE 2）为该 VC 分配的 VC 标签：

c. 将本地 CE（CE 1）的 ID（1）与 PE 2 分配的标签块进行比较，如果存在满足 LO<=CE ID<LO+LR 的标签块，则从该标签块中分配标签。在本例中，标签块（2000/0/15）满足 LO<=CE ID<LO+LR（即 0<=1<0+15），故从该标签块中为此 VC 分配标签。

d. 对端 PE 为此 VC 分配的标签值为：LB+CE ID-LO，即 2000+1-0=2001。

- 在 PE 2 上也进行类似地计算，可以得到相同的结果：本地 PE（PE 2）分配的 VC 标签为 2001、对端 PE（PE 1）分配的 VC 标签为 1062。

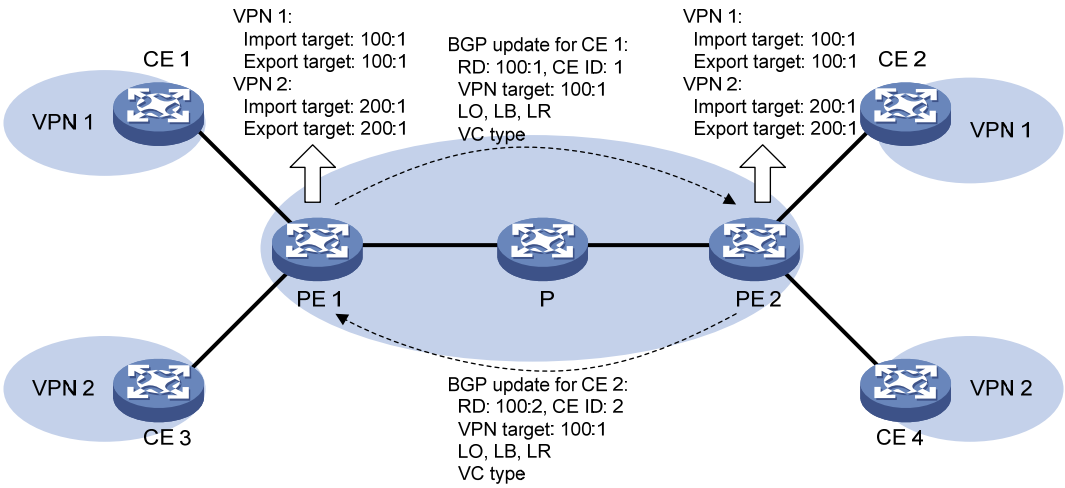
说明

PE 为来自 CE 的二层用户报文封装 VC 标签时，封装的是对端 PE 为该 VC 分配的标签。例如，PE 1 将来自 CE 1 的报文转发给 CE 2 时，封装的 VC 标签为 2001。

---

(4) VC 建立过程

图1-7 Kompella 方式标签分发过程



如 图 1-7 所示，CE 1 和CE 2 属于VPN 1，CE 3 和CE 4 属于VPN 2。通过为两个VPN分别配置VPN target属性，实现VPN 1 内的两个CE、VPN 2 内的两个CE分别可以建立VC连接，但VPN 1 内的CE与VPN 2 内的CE不能建立VC连接。

VC 建立过程为：

a. PE 1 通过 BGP 的 Update 消息将 RD、CE ID、VPN target（PE 1 上为 VPN 配置的 Export target）、标签块等信息发送给 PE 2。

b. PE 2 接收到 Update 消息后，比较该消息中的 VPN target 是否与本地为各 VPN 配置的 Import target 匹配。如果与某个 VPN 的 Import target 匹配，则为该 VPN 学习 Update 消息中的 L2VPN 信息；否则，不会学习该 Update 消息中的信息，不进行后续的处理。在本例中，传递 CE 1 信息的 Update 消息的 VPN target 为 100:1，与 VPN 1 的 Import target 相同，则将 CE 1 的信息学习到 VPN 1 的 L2VPN 信息中；传递 CE 2 信息的 Update 消息的 VPN target 为 200:1，与 VPN 2 的 Import target 相同，则将 CE 2 的信息学习到 VPN 2 的 L2VPN 信息中。属于不同 VPN 的 CE 信息学习到不同的 VPN 中，从而实现不同 VPN 中 CE 的隔离。

c. 同样地，PE 2 也向 PE 1 发送 Update 消息，PE 1 也进行类似地处理。

d. PE 1 和 PE 2 根据学习到的 L2VPN 信息，按照上文描述的方法为每个 VPN 内的 CE 分别计算 VC 标签。PE 1 和 PE 2 均计算出 VC 标签后，就成功建立了 VC 连接。

5. 四种实现方式的比较

四种实现方式的对比如 表 1-1 所示。

表1-1 四种实现方式对比

实现方式	封装层数	VC 标签分发方式	优点	缺点	适用场景
CCC	一层	静态配置	• 不需要任何信令协议传递标签，占用网络资源少 • 网络设备只要能支持 MPLS 转发即可 • 由于 LSP 是专用的，采用 CCC 方式，可以为业务流量提供 QoS 保证	• 配置维护复杂。每台 P 设备上都需要为每条 CCC 连接配置两条方向相反的静态 LSP • 不能自动适应网络的变化	拓扑简单、且其中的设备都支持MPLS的小型网络
SVC	两层	静态配置	不需要任何信令协议传递VC标签，占用网络资源少	• 不能自动适应网络的变化 • 只支持远程连接	拓扑简单的小型网络
Martini	两层	LDP	• 在运营商网络中，只有 PE 设备需要保存少量的 VC 标签与 LSP 的映射关系等信息，P 设备不包含任何二层 VPN 信息 • 当需要新增一条 VC 时，只在相关的两端 PE 上进行配置即可，不影响网络的运行	只支持远程连接	稀疏的二层连接，例如星型连接
Kompella	两层	BGP	• 引入 VPN 的概念，限制属于不同 VPN 的 CE 之间的通信 • 采用标签块的方式允许用户为 VPN 分配一些额外的标签，留待以后使用，减少 VPN 部署和扩容时的配置工作量 • 网络中新增一台 CE 时，只需在与其连接的 PE 上进行配置，维护工作量小，且不影响网络的运行	实现相对复杂，不易于理解	全连接的网络

1.1.6 VC封装类型

为二层报文封装 VC 标签前，PE 对不同链路层协议的二层报文的处理方式有所不同。VC 封装类型（VC type）用来标识 PE 对二层报文的处理方式。VC 封装类型与 AC 的链路类型（PE—CE 之间的链路类型）密切相关，对于以太网类型的 AC 链路，VC 封装类型可以为 Ethernet 和 VLAN 两种。

- Ethernet：采用该封装类型时，PW 上传输的报文不能携带 P-Tag。对于从 CE 侧接收到的报文，如果带有 P-Tag，则删除 P-tag 后对报文进行封装；如果不带 P-Tag，则直接对报文进行封装。对于 PE 转发给 CE 的下行报文，如果用户接入方式为 VLAN，则 PE 添加 P-Tag 后转

发给 CE；如果用户接入方式为 Ethernet，则 PE 不添加 P-Tag，直接将报文转发给 CE；PE 不允许重写或去除报文中已经存在的任何 Tag。

- VLAN：采用该封装类型时，PW 上传输的报文必须带 P-Tag。对于从 CE 侧接收的报文，如果带有 P-Tag，则对端 PE 不要求 Ingress 改写 P-Tag 时，保留 P-Tag；对端 PE 要求 Ingress 改写 P-Tag 时，将 P-Tag 改写为对端 PE 期望的 VLAN Tag（Tag 可能是值为 0 的空 Tag），再对报文进行封装。如果不带 P-Tag，则对端 PE 不要求 Ingress 改写 P-Tag 时，添加值为 0 的空 P-Tag；对端 PE 要求 Ingress 改写 P-Tag 时，添加一个对端 PE 期望的 VLAN Tag（Tag 可能是值为 0 的空 Tag）后，再对报文进行封装。对于 PE 转发给 CE 的下行报文，如果用户接入方式为 VLAN，则转发给 CE 时需要重写或保留 P-Tag；如果用户接入方式为 Ethernet，则去除 P-Tag 后转发给 CE。

1.2 MPLS L2VPN配置任务简介

MPLS L2VPN 远程连接组网中，需要进行以下配置：

- 在 PE 和 P 上配置 IGP，实现骨干网的 IP 连通性
- 在 PE 和 P 上配置 MPLS、GRE 或 MPLS TE，在骨干网上建立公网隧道
- 在两端的 PE 上配置隧道策略，以便 PE 根据该隧道策略选择承载 VC 的公网隧道
- 在两端的 PE 上配置 MPLS L2VPN，建立 VC，并将 AC 与 VC 绑定

MPLS L2VPN 本地连接组网中，只需在 PE 上将两个 AC 绑定。

本文只介绍 PE 设备上的 MPLS L2VPN 相关配置，其余配置请参考相关手册。

说明

MPLS L2VPN 对于用户网络来说是透明的，因此 CE 上无需进行特殊配置。

表1-2 MPLS L2VPN 配置任务简介

操作	说明	详细配置
配置MPLS L2VPN基本功能	必选 本配置用来使能MPLS L2VPN功能，以便进行MPLS L2VPN的其他配置	1.3
配置连接CE的接口	必选 本配置用来在PE和CE之间建立AC	1.4
配置CCC方式MPLS L2VPN	选择其一 根据实际需求选择合适的实现方式 本配置用来在PE之间建立VC，并将AC和VC绑定	1.5
配置SVC方式MPLS L2VPN		1.6
配置Martini方式MPLS L2VPN		1.7
配置Kompella方式MPLS L2VPN		1.8

1.3 配置MPLS L2VPN基本功能

表1-3 配置 MPLS L2VPN 基本功能

操作	命令	说明
进入系统视图	**system-view**	-
配置LSR-ID	**mpls lsr-id** *lsr-id*	必选
配置MPLS基本能力，并进入MPLS视图	**mpls**	必选
退回系统视图	**quit**	-
使能L2VPN，并进入L2VPN视图	**l2vpn**	必选 缺省情况下，L2VPN功能处于关闭状态
使能MPLS L2VPN	**mpls l2vpn**	必选 缺省情况下，MPLS L2VPN功能处于关闭状态

1.4 配置连接CE的接口

CE 侧接口为 PE 设备上与 CE 相连的接口。

如 表 1-4 所示，CE侧接口的封装类型有两种，请查看相应章节了解各种封装类型的配置方法。

表1-4 CE 侧接口的封装类型

CE 侧接口的封装类型	详细配置
Ethernet	1.4.1
VLAN	1.4.2

1.4.1 配置CE侧接口的封装类型为Ethernet

在服务实例上配置 Martini 方式的 MPLS L2VPN 时，用户可以修改 CE 侧接口的封装类型；在配置其它类型的 MPLS L2VPN 时，均只能使用缺省的封装类型。

缺省情况下，当接口类型为三层以太网接口时，封装类型为 Ethernet。有关三层以太网接口的配置请参见“二层技术-以太网交换配置指导”中的“以太网接口”；

1.4.2 配置CE侧接口的封装类型为VLAN

在服务实例上配置 Martini 方式的 MPLS L2VPN 时，用户可以修改 CE 侧接口的封装类型；在配置其它类型的 MPLS L2VPN 时，均只能使用缺省的封装类型。

缺省情况下，当接口类型为 VLAN 接口时，封装类型为 VLAN，其中该接口所属的 VLAN ID 与 CE 的 VLAN ID 一致。有关 VLAN 接口的配置请参见“二层技术-以太网交换配置指导”中的“VLAN”。

1.5 配置CCC方式MPLS L2VPN

配置 CCC 远程连接时，需要在 PE 和 P 上分别进行如下配置：

- 在两端 PE 上分别通过 **ccc interface in-label out-label** 命令指定入标签和出标签等。PE 上不需要执行 **static-lsp** 命令为每个 CCC 远程连接配置静态 LSP。
- 在 PE 之间的所有 P 设备上执行 **static-lsp** 命令为两个数据传输方向分别配置一条静态 LSP，用于专门传输 CCC 连接的数据。
- **static-lsp** 命令的详细介绍，请参见“MPLS 命令参考”中的“MPLS 基础”。

表1-5 配置 PE

操作	命令	说明
进入系统视图	**system-view**	-
在PE上创建一条连接两个CE的远程CCC连接	**ccc** *ccc-connection-name* **interface** *interface-type interface-number* **in-label** *in-label-value* **out-label** *out-label-value* **nexthop** *ip-address*	必选 **interface** *interface-type interface-number*参数指定的接口为PE上连接CE的接口，通过指定该接口可以实现将该接口所在的AC链路与此CCC远程连接绑定

注意

- CCC 使用的标签范围是 16～1023，即保留给静态 LSP 使用的标签。
- CCC 方式下，如果 PE 上连接 CE 的接口为 VLAN 接口，则 PE 发送给 CE 的报文都会携带 VLAN Tag。因此，在这种情况下，需要配置 CE 上连接 PE 的接口可以接收携带 VLAN Tag 的报文。

表1-6 配置 P

操作	命令	说明
进入系统视图	**system-view**	-
为Transit节点配置静态LSP	**static-lsp transit** *lsp-name* **incoming-interface** *interface-type interface-number* **in-label** *in-label* **nexthop** *next-hop-addr* **out-label** *out-label*	必选

1.6 配置SVC方式MPLS L2VPN

SVC 方式 MPLS L2VPN 不使用信令协议传输 L2VPN 信息，数据是通过隧道在 PE 之间传递。

SVC 支持的隧道类型包括 LDP LSP、CR-LSP，缺省情况下使用 LDP LSP 隧道。

SVC 在三层接口上创建，包括三层以太网端口和 VLAN 接口。在三层以太网端口上配置 SVC 后，从该端口接收到的报文将通过创建的 VC 连接转发。如果在 VLAN 接口上配置 SVC，则不同二层以太网端口接收的带有该 VLAN 接口对应 VLAN Tag 的报文均通过创建的 VC 连接转发，即只能根据接收报文中的 VLAN Tag 匹配绑定的 VC 连接，无法区分不同二层以太网端口连接的不同用户和业务。因此，当三层接口连接的用户都通过同一个 VC 连接转发报文时，可以使用 SVC 方式。

表1-7 配置 SVC 方式 MPLS L2VPN

操作	命令	说明
进入系统视图	**system-view**	-
进入连接CE的接口视图	**interface** *interface-type interface-number*	-
创建普通方式SVC的VC连接	**mpls static-l2vc destination** *destination-router-id* **transmit-vpn-label** *transmit-label-value* **receive-vpn-label** *receive-label-value* [{ **ethernet** | **vlan** } | **tunnel-policy** *tunnel-policy-name*] *	必选

说明

- SVC 使用的标签范围是 16～1023，即保留给静态 LSP 使用的标签。
- 本地 PE 上配置的 **transmit-vpn-label** 与对端 PE 上配置的 **receive-vpn-label** 应该相同；本地 PE 上配置的 **receive-vpn-label** 与对端 PE 上配置的 **transmit-vpn-label** 应该相同。

1.7 配置Martini方式MPLS L2VPN

配置 Martini 方式的 MPLS L2VPN 时需要：

(1) 配置远端对等体

Martini 方式的 MPLS L2VPN 中，需要在 PE 之间交换 VC 标签。由于交换 VC 标签的两个 PE 可能不是直接相连的，所以需要在两个 PE 上分别将对端 PE 配置为远端对等体，以便在 PE 之间建立 LDP 远端会话，并在这个会话上传递 VC FEC 和 VC 标签。

(2) 创建 Martini 方式 VC 连接

用户可以通过以下两种方式创建 Martini 方式的 VC 连接：

- 三层接口上创建：在三层接口上创建 Martini 方式的 VC 连接后，从该接口接收到的报文将通过创建的 VC 连接转发。如果三层接口为 VLAN 接口，则不同二层以太网接口接收的相同 VLAN Tag 的报文均通过创建的 VC 连接转发，即只能根据接收报文中的 VLAN Tag 匹配绑定的 VC 连接，无法区分不同二层以太网接口连接的不同用户和业务。三层接口连接的用户都通过同一个 VC 连接转发报文时，可以采用此方式。
- 服务实例（Service Instance）上创建：在服务实例上创建 Martini 方式的 VC 连接后，设备根据二层以太网接口或二层聚合接口上创建的服务实例对该接口接收到的报文进行匹配，与服务实例匹配的报文将通过创建的 VC 连接转发。服务实例提供了多种报文匹配规则（包括接口接收到的所有报文、所有携带 VLAN Tag 的报文和所有不携带 VLAN Tag 的报文等），为报文

接入 VC 连接提供了更加灵活的方式。VLAN 接口连接的用户需要通过不同的 VC 连接转发报文时，可以采用此方式。服务实例的详细介绍请参见“MPLS 配置指导”中的“VPLS”。

说明

服务实例只能在二层以太网端口或二层聚合接口上创建。

1.7.1 配置远端对等体

下述配置需要在 PE 上进行。

表1-8 配置远端对等体

操作	命令	说明
进入系统视图	**system-view**	-
创建远端对等体实体并进入MPLS-LDP远端对等体视图	**mpls ldp remote-peer** *remote-peer-name*	必选
将对端PE的地址指定为LDP远端对等体的IP地址	**remote-ip** *ip-address*	必选

说明

有关远端对等体的配置请参见“MPLS 配置指导”中的“MPLS 基础”。

1.7.2 在三层接口上创建Martini方式的VC连接

表1-9 在三层接口上创建 Martini 方式的 VC 连接

操作	命令	说明
进入系统视图	**system-view**	-
进入连接CE的接口视图	**interface** *interface-type interface-number*	-
创建Martini方式VC连接	**mpls l2vc** *destination vcid* [{ **ethernet** | **vlan** } | [**tunnel-policy** *tunnel-policy-name*]] *	必选

注意

创建 Martini 方式 VC 连接的命令主要参数有两个：一个是对端 PE 的 IP 地址，一个是 VC ID。其中，VC ID 与封装类型的组合必须在 PE 上唯一，修改封装类型有可能会造成 VC ID 的冲突。

1.7.3 配置在服务实例上创建Martini方式的VC连接

完成本配置任务，需要在 PE 上执行以下操作：

- 在二层以太网接口或二层聚合接口上创建服务实例
- 为服务实例配置报文匹配规则
- 在服务实例上创建 Martini 方式的 VC 连接

配置完成后，如果接口接收到的报文符合服务实例的报文匹配规则，则该报文将通过该服务实例下创建的 VC 连接进行转发。

为了简化 VC 属性（如 VC 封装类型、VC 隧道选用策略）的配置，可以先创建 PW 模板，在 PW 模板下配置 VC 属性，具有相同属性的 VC 通过引用相同的 PW 模板实现对 VC 属性的配置，无需逐个配置每个属性。

表1-10 在服务实例上创建 Martini 方式的 VC 连接

操作		命令	说明
进入系统视图		**system-view**	-
创建PW模板，并进入PW模板视图		**pw-class** *pw-class-name*	可选 缺省情况下，不存在任何PW模板
设置VC封装类型		**trans-mode** { **ethernet** | **vlan** }	可选 缺省情况下，VC封装类型为VLAN
设置隧道选用策略		**pw-tunnel-policy** *policy-name*	可选 缺省情况下，采用缺省策略，即按照LSP隧道－> CR-LSP隧道的优先级顺序选择隧道，并指定负载分担个数为1 隧道策略的配置方法，请参见“MPLS配置指导”中的“MPLS L3VPN”
退回系统视图		**quit**	-
进入连接CE的二层以太网接口或二层聚合接口视图	进去二层以太网接口视图	**interface** *interface-type interface-number*	-
	进去二层聚合接口视图	**interface bridge-aggregation** *interface-number*	
创建服务实例，并进入服务实例视图		**service-instance** *instance-id*	必选 缺省情况下，不存在任何服务实例
配置报文匹配规则		**encapsulation** { **s-vid** { *vlan-id* | *vlan-list* } [**only-tagged**] | **port-based** | **tagged** | **untagged** }	必选 缺省情况下，不存在任何报文匹配规则
在服务实例上创建Martini方式的VC连接		**xconnect peer** *peer-ip-address* **pw-id** *pw-id* [**access-mode** { **ethernet** | **vlan** } | **mtu** *mtu-value* | [**pw-class** *class-name*]] *	必选 此命令执行后，服务实例下匹配的VLAN ID、接入模式和MTU值均不可更改，只有执行**undo xconnect peer**命令删除VC连接后，才可以修改这些参数

操作	命令	说明
显示接口上服务实例的信息	**display service-instance interface** *interface-type interface-number* [**service-instance** *instance-id*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]	**display**命令可以在任意视图执行

说明

- 目前，MPLS L2VPN 不支持多条隧道的负载分担。
- **xconnect peer** 命令只能在编号为 1～4094 的服务实例视图下进行配置，否则系统会提示出错。

1.7.4 检测VC

说明

MPLS LSP Ping 只能用来检测 Martini 方式 VC 的可达性。

在 MPLS L2VPN 网络中，通过 MPLS LSP Ping 功能，可以对 VC 的可达性进行检测，并提供必要的诊断信息，以便对 VC 的故障进行定位。

MPLS LSP Ping 功能采取的方法是：在本地 PE 设备上为 MPLS Echo Request 报文压入待检测的 VC 对应的标签，使得 MPLS Echo Request 报文沿着 VC 转发，本地 PE 设备根据收到的对端 PE 设备的应答报文，判断 VC 的可达性。

表1-11 利用 MPLS LSP Ping 功能检测 VC

操作	命令	说明
通过MPLS LSP Ping检测VC的可达性	**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*	必选 可在任意视图下执行本命令

1.8 配置Kompella方式MPLS L2VPN

配置 Kompella 方式远程连接时，需要在 PE 上进行以下配置：

- 配置 BGP 的 L2VPN 能力
- 创建并配置 MPLS L2VPN
- 配置 CE 连接
- 配置 Kompella 方式本地连接时，不需要配置 BGP 的 L2VPN 能力，只需在 PE 上创建并配置 MPLS L2VPN、配置 CE 连接。

1.8.1 配置BGP的L2VPN能力

表1-12 配置 BGP 的 L2VPN 能力

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
与对端PE建立对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
指定建立TCP连接的接口	**peer** { *group-name* | *ip-address* } **connect-interface** *interface-type interface-number*	必选
进入BGP-L2VPN地址族视图	**l2vpn-family**	必选
对接收到的VPNv4路由使能VPN-Target过滤功能	**policy vpn-target**	可选 缺省情况下，对接收的路由信息进行VPN-target扩展团体属性的过滤
使能对等体，并使能交换BGP-L2VPN地址族的BGP路由信息的能力	**peer** { *group-name* | *ip-address* } **enable**	必选

说明

BGP-L2VPN 地址族下的详细配置，请参见“MPLS 配置指导”中的“MPLS L3VPN”。

1.8.2 创建并配置MPLS L2VPN

表1-13 配置 VPN

操作	命令	说明
进入系统视图	**system-view**	-
创建MPLS L2VPN，并进入MPLS-L2VPN视图	**mpls l2vpn** *vpn-name* [**encapsulation** { **ethernet** | **vlan** }]	必选
为MPLS L2VPN配置RD	**route-distinguisher** *route-distinguisher*	必选
将指定MPLS L2VPN和一个或多个VPN Target相关联	**vpn-target** *vpn-target*&<1-16> [**both** | **export-extcommunity** | **import-extcommunity**]	必选
配置MPLS L2VPN的MTU值	**mtu** *mtu*	可选 缺省情况下，MPLS L2VPN的MTU值为1500

注意

- **mtu** 命令只进行有可能存在的协议上的参数协商，并不指导转发，因此不建议使用此命令。
- Kompella 方式 MPLS L2VPN 必须在 PE 上为每个直接相连的 CE 所在的 VPN 创建 L2VPN。创建 L2VPN 时指定的封装类型应与 AC 链路类型对应。
- 对于 Kompella L2VPN，必须配置 RD。RD 配置后不能修改，除非先删除创建的 VPN，然后重新创建。

1.8.3 配置CE连接

1. 简介

配置 CE 连接时的主要参数解释如下：

(1) **id** *ce-id*

PE 连接的本地 CE 的 CE ID。

(2) **range** *ce-range*

当前 CE 在一个 VPN 内最多可以连接的 CE 数——CE 范围（CE range）。

建议根据对 VPN 规模发展的预计，把 CE range 设置得比实际需要大一些。这样当以后对 VPN 进行扩容，增加 VPN 中的 CE 数目时，就可以尽量少的修改配置。

(3) **default-offset** *default-offset*

VPN 中 CE 的起始编号，取值为 0 或 1。取值为 0 时，表示 VPN 内的 CE 从 0 开始编号；取值为 1 时，表示 VPN 内的 CE 从 1 开始编号。

本参数和 CE range 决定了 PE 为 CE 分配的标签块：

- 第一次执行 **ce** 命令时指定 CE range 为 *ce-range1*，则分配第一个标签块，其 LR 与 CE range 相同，为 *ce-range1*。如果 *default-offset* 为 0，则 LO 为 0；否则，LO 为 1。
- 再次执行 **ce** 命令时将 CE range 增加为 *ce-range2*（大于 *ce-range1*），则分配第二个标签块，LR 为 *ce-range2*－*ce-range1*。如果 *default-offset* 为 0，则 LO 为 *ce-range1*；否则，LO 为 *ce-range1*＋1。以此类推。

例如，在 PE 上先后执行如下命令，则 PE 分配三个标签块，分别为：LB1/0/10、LB2/10/12、LB3/22/14。其中，LB1、LB2、LB3 为 PE 自动选取的标签值。

```
ce ce1 id 1 range 10 default-offset 0
ce ce1 id 1 range 22
ce ce1 id 1 range 36
```

(4) **ce-offset** *ce-id*

与本地 CE 建立远程连接或本地连接的对端 CE 的 CE ID。

如果执行 **connection** 命令时没有指定本参数，则：

- 第一次执行 **connection** 命令时，为本地 CE 与 ID 为 *default-offset* 的对端 CE 建立连接。如果 *default-offset* 的值与本地 CE ID 相同，则为本地 CE 与 ID 为 *default-offset*＋1 的对端 CE 建立连接。

- 再次执行 **connection** 命令时，为本地 CE 与 ID 为<上一个连接的 CE ID＋1>的对端 CE 建立连接。如果<上一个连接的 CE ID＋1>与本地 CE ID 相同，则为本地 CE 与 ID 为<上一个连接的 CE ID＋2>的对端 CE 建立连接。以此类推。

在规划 VPN 时，建议 CE ID 编号顺序递增，在配置连接时按 CE ID 顺序配置，这样，大多数连接都可以省略 **ce-offset** 参数，使用缺省值，从而简化配置。

2. 配置过程

表1-14 配置 CE 连接

操作	命令	说明
进入系统视图	**system-view**	-
进入MPLS-L2VPN视图	**mpls l2vpn** *vpn-name*	-
创建CE，指定CE名称、CE ID、CE range、CE起始编号，并进入MPLS-L2VPN-CE视图	**ce** *ce-name* [**id** *ce-id* [**range** *ce-range*] [**default-offset** *default-offset*]]	必选
创建Kompella方式连接	**connection** [**ce-offset** ce-*id*] **interface** *interface-type interface-number* [**tunnel-policy** *tunnel-policy-name*]	必选 **ce-offset** *ce-id*参数指定的CE ID值决定了此连接是本地连接还是远程连接。如果该CE ID标识的CE与本CE连接在同一个PE上，则建立的为本地连接；否则，为远程连接 本配置中指定的接口为PE上连接CE的接口，通过指定该接口可以实现将该接口所在的AC链路与此连接绑定

说明

- 通过重复执行 **ce** 命令的方式修改 CE range 时，只能把 CE range 改得更大，不能改小。例如：原来的 CE range 为 10，则可以把它改为 20；如果改为 5，则会失败。要想将 CE range 改小，则需要删除这个 CE，并重新创建。
- 通过重复执行 **ce** 命令的方式将 CE range 改得更大时，不会导致原有业务的中断。

3. 复位L2VPN的BGP会话

当影响 BGP 路由选择的配置发生变化后，如果需要通过复位 BGP 会话使新的配置生效，请在用户视图下进行以下配置。

表1-15 复位 L2VPN 的 BGP 会话

操作	命令
复位L2VPN的BGP会话	**reset bgp l2vpn** { *as-number* | *ip-address* | **all** | **external** | **internal** }

1.9 MPLS L2VPN显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 MPLS L2VPN 的运行情况，通过查看显示信息验证配置的效果。

表1-16 MPLS L2VPN 显示和维护

操作	命令
显示CCC连接信息	**display ccc** [**ccc-name** *ccc-name* | **type** { **local** | **remote** }] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示L2VPN VC使用的接口信息	**display l2vpn ccc-interface vc-type** { **all** | **bgp-vc** | **ccc** | **ldp-vc** | **static-vc** } [**up** | **down**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示设备上创建的SVC的相关信息	**display mpls static-l2vc** [**interface** *interface-type interface-number* [**service-instance** *instance-id*]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示设备上Martini方式VC的相关信息	**display mpls l2vc**[**interface** *interface-type interface-number* [**service-instance** *instance-id*] | **remote-info**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示Kompella方式的VC连接信息	**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*]
显示BGP路由表中的L2VPN信息	**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*]
显示PE上的L2VPN信息	**display mpls l2vpn** [**export-route-target-list** | **import-route-target-list** | **vpn-name** *vpn-name* [**local-ce** | **remote-ce**]] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示MPLS L2VPN的AC表项信息	**display mpls l2vpn fib ac vpws** [**interface** *interface-type interface-number* [**service-instance** *service-instanceid*]] [**slot** *slot-number*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示MPLS L2VPN的PW表项信息	**display mpls l2vpn fib pw vpws** [**interface** *interface-type interface-number* [**service-instance** *service-instanceid*]] [**slot** *slot-number*] [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示PW模板的信息	**display pw-class** [*pw-class-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]

1.10 MPLS L2VPN典型配置举例

1.10.1 配置CCC远程连接示例

1. 组网需求

- PE 通过 VLAN 接口连接 CE。
- 通过建立 CCC 远程连接，实现 CE 1 和 CE 2 之间的二层报文跨越骨干网络传递。

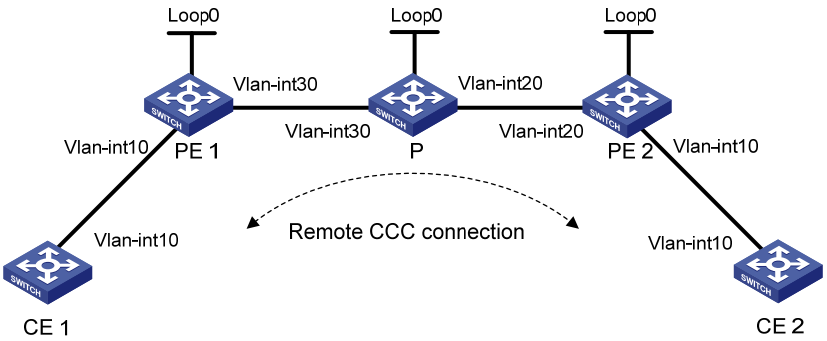
2. 配置思路

配置 CCC 远程连接的关键步骤包括：

- 在 PE 上创建 CCC 远程连接（不需要配置静态 LSP）。
- 在 P 上配置两条静态 LSP，用于双向传递报文。

3. 组网图

图1-8 配置 CCC 远程连接组网图



设备	接口	IP地址	设备	接口	IP地址
CE 1	Vlan-int10	100.1.1.1/24	CE 2	Vlan-int10	100.1.1.2/24
PE 1	Loop0	10.0.0.1/32	P	Loop0	10.0.0.2/32
	Vlan-int30	10.1.1.1/24		Vlan-int20	10.2.2.2/24
PE 2	Loop0	10.0.0.3/32		Vlan-int30	10.1.1.2/24
	Vlan-int20	10.2.2.1/24			

4. 配置步骤

(1) 配置 CE 1

为连接 PE 1 的接口 Vlan-interface10 配置 IP 地址。

```
<Sysname> system-view
[Sysname] sysname CE1
[CE1] interface vlan-interface 10
[CE1-Vlan-interface10] ip address 100.1.1.1 24
```

(2) 配置 PE 1

配置 LSR ID，全局使能 MPLS。

```
<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
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

配置接口 Vlan-interface30，使能 MPLS。

```
[PE1] interface vlan-interface 30
[PE1-Vlan-interface30] ip address 10.1.1.1 24
[PE1-Vlan-interface30] mpls
[PE1-Vlan-interface30] quit
```

创建 CE 1 到 CE 2 的远程连接：入接口为连接 CE 1 的接口，出接口为连接 P 的接口；入标签为 100，出标签为 200。

```
[PE1] ccc ce1-ce2 interface vlan-interface 10 in-label 100 out-label 200 nexthop 10.1.1.2
```

(3) 配置 P

配置 LSR ID，全局使能 MPLS。

```
<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
```

配置接口 Vlan-interface30，使能 MPLS。

```
[P] interface vlan-interface 30
[P-Vlan-interface30] ip address 10.1.1.2 24
[P-Vlan-interface30] mpls
[P-Vlan-interface30] quit
```

配置接口 Van-interface20，使能 MPLS。

```
[P] interface vlan-interface 20
[P-Vlan-interface20] ip address 10.2.2.2 24
[P-Vlan-interface20] mpls
[P-Vlan-interface20] quit
```

配置一条静态 LSP 用于转发由 PE 1 去往 PE 2 的报文。

```
[P] static-lsp transit pe1_pe2 incoming-interface vlan-interface 30 in-label 200 nexthop
10.2.2.1 out-label 201
```

配置另一条静态 LSP 用于转发由 PE 2 去往 PE 1 的报文。

```
[P] static-lsp transit pe2_pe1 incoming-interface vlan-interface 20 in-label 101 nexthop
10.1.1.1 out-label 100
```

(4) 配置 PE 2

配置 LSR ID，全局使能 MPLS。

```
<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
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
```

```
[PE2-l2vpn] quit
```

配置接口 Vlan-interface10。

```
[PE2] interface vlan-interface 10
[PE2-Vlan-interface10] quit
```

配置接口 Vlan-interface20，使能 MPLS。

```
[PE2] interface vlan-interface 20
[PE2-Vlan-interface20] ip address 10.2.2.1 24
[PE2-Vlan-interface20] mpls
[PE2-Vlan-interface20] quit
```

创建 CE 2 到 CE 1 的远程连接：入接口为连接 CE 2 的接口，出接口为连接 P 的接口；入标签为 201，出标签为 101。

```
[PE2] ccc ce2-ce1 interface vlan-interface 10 in-label 201 out-label 101  nexthop 10.2.2.2
```

(5)　配置 CE 2

为连接 PE 2 的接口 Vlan-interface10 配置 IP 地址。

```
<Sysname> system-view
[Sysname] sysname CE2
[CE2] interface vlan-interface 10
[CE2-Vlan-interface10] ip address 100.1.1.2 24
```

5. 验证配置

配置完成后，在 PE 1 上查看 CCC 连接信息，可以看到建立了一条 CCC 远程连接。

```
[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
```

CE 1 与 CE 2 之间能够 ping 通。

```
[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
```

1.10.2 配置SVC方式MPLS L2VPN示例

1. 组网需求

- PE 通过 VLAN 接口连接 CE。
- 通过建立 SVC 方式的 VC 连接，实现 CE 1 和 CE 2 之间的二层报文跨越骨干网络传递。

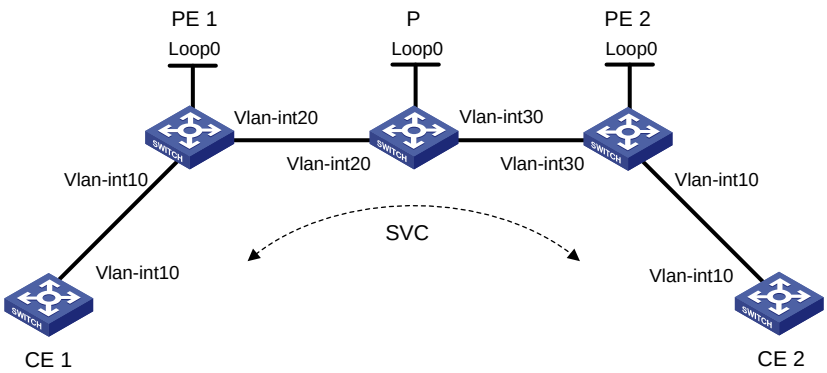
2. 配置思路

主要的配置步骤可分为两部分：

- 在 PE 和 P 上配置 MPLS 基本转发能力：包括配置 LSR ID、使能 MPLS 和 LDP、在 PE 1－P－PE 2 之间运行 IGP（本配置举例中使用 OSPF）以建立 LSP。
- 建立 SVC 方式的 VC 连接：包括在 PE 1 和 PE 2 上使能 MPLS L2VPN、创建 SVC 连接并指定 VC 标签。

3. 组网图

图1-9 配置 SVC 方式 MPLS L2VPN 组网图



设备	接口	IP地址	设备	接口	IP地址
CE 1	Vlan-int10	100.1.1.1/24	CE 2	Vlan-int10	100.1.1.2/24
PE 1	Loop0	192.2.2.2/32	P	Loop0	192.4.4.4/32
	Vlan-int20	10.1.1.1/24		Vlan-int30	10.2.2.2/24
PE 2	Loop0	192.3.3.3/32		Vlan-int20	10.1.1.2/24
	Vlan-int30	10.2.2.1/24			

4. 配置步骤

(1) 配置 CE 1

为连接 PE 1 的接口 Vlan-interface10 配置 IP 地址。

```
<Sysname> system-view
[Sysname] sysname CE1
[CE1] interface vlan-interface 10
[CE1-Vlan-interface10] ip address 100.1.1.1 24
```

(2) 配置 PE 1

配置 LSR ID，全局使能 MPLS。

```
<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
[PE1-mpls] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

全局使能 LDP。

```
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

配置连接 P 的接口 Vlan-interface20，在此接口上使能 LDP。

```
[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
```

在 PE 1 上运行 OSPF，用于建立 LSP。

```
[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
```

在接入 CE 1 的接口 Vlan-interface10 上创建一条 SVC 方式 VC 连接。此接口不需配置 IP 地址。

```
[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) 配置 P

配置 LSR ID，全局使能 MPLS。

```
<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
[P-mpls] quit
```

全局使能 LDP。

```
[P] mpls ldp
[P-mpls-ldp] quit
```

配置连接 PE 1 的接口 Vlan-interface20，在此接口上使能 LDP。

```
[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
```

配置连接 PE 2 的接口 Vlan-interface30，在此接口上使能 LDP。

```
[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
```

在 P 上运行 OSPF，用于建立 LSP。

```
[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) 配置 PE 2

配置 LSR ID，全局使能 MPLS。

```
<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
[PE2-mpls] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```

全局使能 LDP。

```
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

配置连接 P 的接口 Vlan-interface30，在此接口上使能 LDP。

```
[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
```

在 PE 2 上运行 OSPF，用于建立 LSP。

```
[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
```

在接入 CE 2 的接口 Vlan-interface10 上创建一条 SVC 方式 VC 连接。此接口不需配置 IP 地址。

```
[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) 配置 CE 2

为连接 PE 2 的接口 Vlan-interface10 配置 IP 地址。

```
<Sysname> system-view
[Sysname] sysname CE2
[CE2] interface vlan-interface 10
[CE2-Vlan-interface10] ip address 100.1.1.2 24
```

5. 验证配置

在 PE 1 上查看 SVC 的 VC 连接信息，可以看到建立了一条 VC 连接。

```
[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       -
```

在 PE 2 上也可以看到 SVC 的 VC 连接。

```
[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       -
```

CE 1 与 CE 2 之间能够 ping 通。

```
[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
```

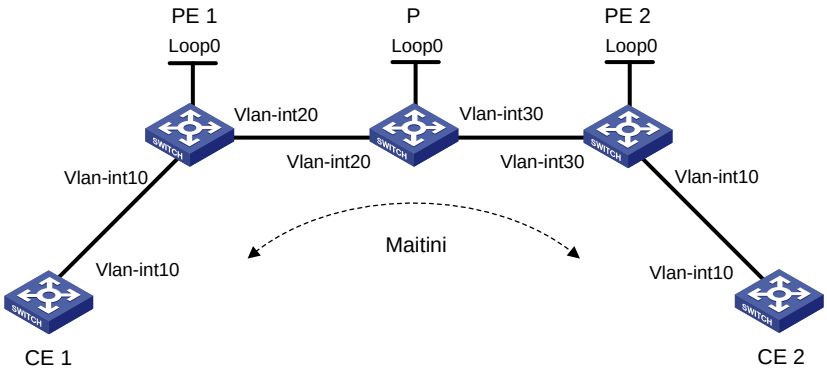
1.10.3 配置Martini方式MPLS L2VPN示例

1. 组网需求

- PE 通过 VLAN 接口连接 CE。
- 通过建立 Martini 方式的 VC 连接，实现 CE 1 和 CE 2 之间的二层报文跨越骨干网络传递。

2. 组网图

图1-10 配置 Martini 方式 MPLS L2VPN 组网图



设备	接口	IP地址	设备	接口	IP地址
CE 1	Vlan-int10	100.1.1.1/24	CE 2	Vlan-int10	100.1.1.2/24
PE 1	Loop0	192.2.2.2/32	P	Loop0	192.4.4.4/32
	Vlan-int20	10.1.1.1/24		Vlan-int20	10.1.1.2/24
PE 2	Loop0	192.3.3.3/32		Vlan-int30	10.2.2.2/24
	Vlan-int30	10.2.2.1/24			

3. 配置步骤

(1) 配置 CE 1

为连接 PE 1 的接口 Vlan-interface10 配置 IP 地址。

```
<Sysname> system-view
[Sysname] sysname CE1
[CE1] interface vlan-interface 10
[CE1-Vlan-interface10] ip address 100.1.1.1 24
```

(2) 配置 PE 1

配置 LSR ID，全局使能 MPLS。

```
<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
[PE1-mpls] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

全局使能 LDP。

```
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

配置 PE 1 与 PE 2 建立 LDP 远程会话。

```
[PE1] mpls ldp remote-peer 1
[PE1-mpls-ldp-remote-1] remote-ip 192.3.3.3
```

```
[PE1-mpls-ldp-remote-1] quit
```
配置连接 P 的接口 Vlan-interface20，在此接口上使能 LDP。
```
[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
```
在 PE 1 上运行 OSPF，用于建立 LSP。
```
[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
```
在接入 CE 1 的接口 Vlan-interface10 上创建 VC 连接。此接口不需配置 IP 地址。
```
[PE1] interface vlan-interface 10
[PE1-Vlan-interface10] mpls l2vc 192.3.3.3 101
[PE1-Vlan-interface10] quit
```
(3) 配置 P

配置 LSR ID，全局使能 MPLS。
```
<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
[P-mpls] quit
```
全局使能 LDP。
```
[P] mpls ldp
[P-mpls-ldp] quit
```
配置连接 PE 1 的接口 Vlan-interface20，在此接口上使能 LDP。
```
[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
```
配置连接 PE 2 的接口 Vlan-interface30，在此接口上使能 LDP。
```
[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
```
在 P 上运行 OSPF，用于建立 LSP。
```
[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) 配置 PE 2

配置 LSR ID，全局使能 MPLS。

```
<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
[PE2-mpls] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE2] l2vpn
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```

全局使能 LDP。

```
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

配置 PE 2 与 PE 1 建立 LDP 远程会话。

```
[PE2] mpls ldp remote-peer 2
[PE2-mpls-ldp-remote-2] remote-ip 192.2.2.2
[PE2-mpls-ldp-remote-2] quit
```

配置连接 P 的接口 Vlan-interface30，在此接口上使能 LDP。

```
[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
```

在 PE 2 上运行 OSPF，用于建立 LSP。

```
[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
```

在接入 CE 2 的接口 Vlan-interface10 上创建 VC 连接。此接口不需配置 IP 地址。

```
[PE2] interface vlan-interface 10
[PE2-Vlan-interface10] mpls l2vc 192.2.2.2 101
[PE2-Vlan-interface10] quit
```

(5) 配置 CE 2

为连接 PE 2 的接口 Vlan-interface10 配置 IP 地址。

```
<Sysname> system-view
```

```
[Sysname] sysname CE2
[CE2] interface vlan-interface 10
[CE2-Vlan-interface10] ip address 100.1.1.2 24
```

4. 验证配置

在 PE 1 上查看 VC 连接信息，可以看到建立了一条 VC 连接。

```
[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
```

在 PE 2 上也可以看到 VC 连接。

```
[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
```

CE 1 与 CE 2 之间能够 ping 通。

```
[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
```

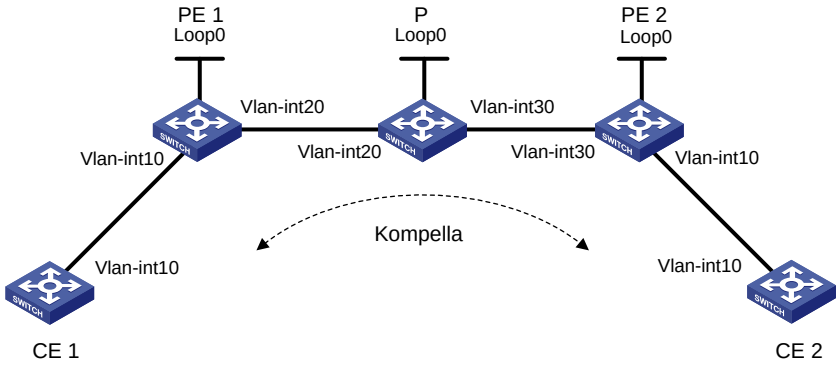
1.10.4 配置Kompella方式MPLS L2VPN示例

1. 组网需求

- PE 通过 VLAN 接口连接 CE。
- 通过建立 Kompella 方式的 VC 连接，实现 CE 1 和 CE 2 之间的二层报文跨越骨干网络传递。

2. 组网图

图1-11 配置 Kompella 方式 MPLS L2VPN 组网图



设备	接口	IP地址	设备	接口	IP地址
CE 1	Vlan-int10	100.1.1.1/24	CE 2	Vlan-int10	100.1.1.2/24
PE 1	Loop0	2.2.2.2/32	P	Loop0	3.3.3.3/32
	Vlan-int20	10.1.1.1/24		Vlan-int20	10.1.1.2/24
PE 2	Loop0	4.4.4.4/32		Vlan-int30	10.2.2.2/24
	Vlan-int30	10.2.2.1/24			

3. 配置步骤

(1) 在 MPLS 骨干网上配置 IGP

本例中使用 OSPF，具体配置步骤略。

配置完成后，在各 LSR 上执行 **display ip routing-table** 命令可以看到都已学到彼此 LSR ID 的路由；执行 **display ospf peer** 命令可以建立了 OSPF 邻居关系，状态为 FULL。

(2) 配置 MPLS 基本能力和 LDP，建立 LDP LSP

具体配置步骤略。

配置完成后，在各 LSR 上执行 **display mpls ldp session** 和 **display mpls ldp peer** 命令可以看到 LDP 会话和对等体的建立情况，执行 **display mpls lsp** 命令可以看到 LSP 的建立情况。

(3) 配置 BGP 的 L2VPN 能力

配置 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
```

配置 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
```

配置完成后，在 PE 1 和 PE 2 上执行 **display bgp l2vpn peer** 命令可以看到 PE 之间建立了对等体关系，状态为 Established。

以 PE 1 为例：

```
[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) 配置 L2VPN 和 CE 连接

配置 PE 1。

```
[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-interface10
[PE1-mpls-l2vpn-ce-vpn1-ce1] quit
[PE1-mpls-l2vpn-vpn1] quit
```

配置 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
```

4. 验证配置

完成上述配置后，在 PE 上执行 **display mpls l2vpn connection** 命令，可以看到建立了一条 VC 连接，状态为 up。

以 PE 1 为例：

```
[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
```

CE 1 与 CE 2 之间能够 ping 通。

```
[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
```

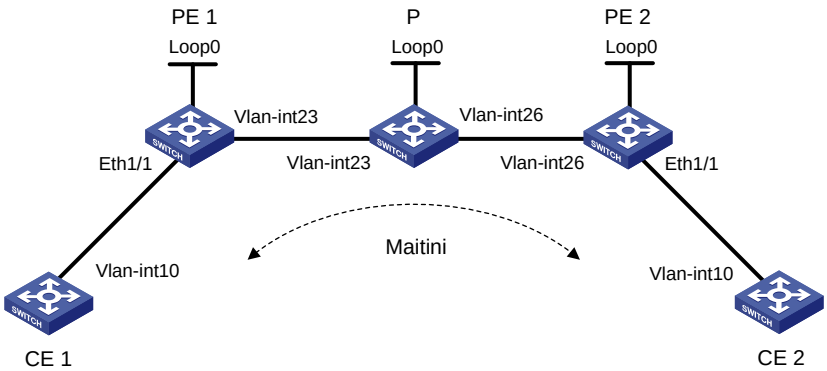
1.10.5 配置在服务实例上创建VC连接示例

1. 组网需求

- CE 1、CE 2 分别通过 VLAN 方式接入 PE 1 和 PE 2。
- PE 1 和 PE 2 在服务实例上为 CE 1 和 CE 2 创建 VC 连接，实现 CE 1 和 CE 2 之间的二层报文跨越骨干网络传递。

2. 组网图

图1-12 配置在服务实例上创建 VC 连接组网图



设备	接口	IP地址	设备	接口	IP地址
CE 1	Vlan-int10	100.1.1.1/24	CE 2	Vlan-int10	100.1.1.2/24
PE 1	Loop0	192.2.2.2/32	P	Loop0	192.4.4.4/32
	Vlan-int23	23.1.1.1/24		Vlan-int23	23.1.1.2/24
PE 2	Loop0	192.3.3.3/32		Vlan-int26	26.2.2.2/24
	Vlan-int26	26.2.2.1/24			

3. 配置步骤

(1) 配置 CE 1

为连接 PE 1 的接口 Vlan-interface10 配置 IP 地址。

```
<Sysname> system-view
[Sysname] sysname CE1
[CE1] interface vlan-interface 10
[CE1-Vlan-interface10] ip address 100.1.1.1 24
```

(2) 配置 PE 1

```
<Sysname> system-view
[Sysname] sysname PE1
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 192.2.2.2 32
[PE1-LoopBack0] quit
```

配置 LSR ID，全局使能 MPLS。

```
[PE1] mpls lsr-id 192.2.2.2
[PE1] mpls
[PE1-mpls] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE1] l2vpn
[PE1-l2vpn] mpls l2vpn
[PE1-l2vpn] quit
```

全局使能 LDP。

```
[PE1] mpls ldp
[PE1-mpls-ldp] quit
```

配置 PE 1 与 PE 2 建立 LDP 远程会话。

```
[PE1] mpls ldp remote-peer 1
[PE1-mpls-ldp-remote-1] remote-ip 192.3.3.3
[PE1-mpls-ldp-remote-1] quit
```

配置连接 P 的接口 Vlan-interface23，在此接口上使能 LDP。

```
[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
```

在 PE 1 上运行 OSPF，用于建立 LSP。

```
[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
```

在接入 CE 1 的接口 GigabitEthernet1/0/1 上创建服务实例，并创建 VC 连接。

```
[PE1] interface gigabitethernet1/0/1
[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) 配置 P

```
<Sysname> system-view
[Sysname] sysname P
[P] interface loopback 0
[P-LoopBack0] ip address 192.4.4.4 32
[P-LoopBack0] quit
```

配置 LSR ID，全局使能 MPLS。

```
[P] mpls lsr-id 192.4.4.4
[P] mpls
[P-mpls] quit
```

全局使能 LDP。

```
[P] mpls ldp
[P-mpls-ldp] quit
```

配置连接 PE 1 的接口 Vlan-interface23，在此接口上使能 LDP。

```
[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
```

配置连接 PE 2 的接口 Vlan-interface26，在此接口上使能 LDP。

```
[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
```

在 P 上运行 OSPF，用于建立 LSP。

```
[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) 配置 PE 2

```
<Sysname> system-view
[Sysname] sysname PE2
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 192.3.3.3 32
[PE2-LoopBack0] quit
```

配置 LSR ID，全局使能 MPLS。

```
[PE2] mpls lsr-id 192.3.3.3
[PE2] mpls
[PE2-mpls] quit
```

使能 L2VPN 和 MPLS L2VPN。

```
[PE2] l2vpn
```

```
[PE2-l2vpn] mpls l2vpn
[PE2-l2vpn] quit
```

全局使能 LDP。

```
[PE2] mpls ldp
[PE2-mpls-ldp] quit
```

配置 PE 2 与 PE 1 建立 LDP 远程会话。

```
[PE2] mpls ldp remote-peer 2
[PE2-mpls-ldp-remote-2] remote-ip 192.2.2.2
[PE2-mpls-ldp-remote-2] quit
```

配置连接 P 的接口 Vlan-interface26，在此接口上使能 LDP。

```
[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
```

在 PE 2 上运行 OSPF，用于建立 LSP。

```
[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
```

在接入 CE 2 的接口 GigabitEthernet1/0/1 上创建服务实例，并创建 VC 连接。

```
[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) 配置 CE 2

为连接 PE 2 的接口 Vlan-interface10 配置 IP 地址。

```
<Sysname> system-view
[Sysname] sysname CE2
[CE2] interface vlan-interface 10
[CE2-Vlan-interface10] ip address 100.1.1.2 24
```

4. 验证配置

在 PE 1 上查看 VC 连接信息，可以看到建立了一条 L2VC。

```
[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
```

在 PE 2 上也可以看到 VC 连接。

```
[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
```

CE 1 与 CE 2 之间能够 ping 通。

```
[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
```

1.11 常见配置错误举例

1.11.1 配置后不能Ping通

1. 错误现象

L2VPN 配置后，Ping 对端失败。查看 VC 状态，发现 VC 状态为 down，Remote 值为无效值。

2. 分析

VC 状态为 down 可能是因为两端封装类型不一致。

3. 处理过程

(1) 查看本端和对端 PE 设备配置的封装类型是否一致，如果配置的封装类型不一样，请修改两端的封装类型。

(2) 检查两端是否已配置了远端对等体，并正确地将对端 PE 地址设置为远端对等体的地址。如果没有配置远端对等体或地址配置错误，请在两端均配置正确的远端对等体地址。

目 录

1 MPLS L3VPN配置 ····· 1-1
1.1 MPLS L3VPN简介 ····· 1-1
1.1.1 MPLS L3VPN概述 ····· 1-1
1.1.2 MPLS L3VPN的基本概念 ····· 1-2
1.1.3 MPLS L3VPN的报文转发 ····· 1-5
1.1.4 MPLS L3VPN的网络架构 ····· 1-6
1.1.5 MPLS L3VPN的路由信息发布 ····· 1-8
1.1.6 跨域VPN ····· 1-9
1.1.7 运营商的运营商 ····· 1-12
1.1.8 嵌套VPN ····· 1-14
1.1.9 HoVPN ····· 1-16
1.1.10 OSPF VPN扩展 ····· 1-18
1.1.11 BGP的AS号替换和SoO属性 ····· 1-21
1.2 MPLS L3VPN配置任务简介 ····· 1-22
1.3 配置基本MPLS L3VPN ····· 1-22
1.3.1 基本MPLS L3VPN配置任务简介 ····· 1-22
1.3.2 配置准备 ····· 1-22
1.3.3 配置VPN实例 ····· 1-23
1.3.4 配置PE-CE间路由交换 ····· 1-27
1.3.5 配置PE-PE间的路由交换 ····· 1-33
1.3.6 配置BGP-VPNv4 子地址族下的路由特性 ····· 1-34
1.4 配置跨域VPN ····· 1-37
1.4.1 配置准备 ····· 1-37
1.4.2 配置跨域VPN-OptionA ····· 1-37
1.4.3 配置跨域VPN-OptionB ····· 1-37
1.4.4 配置跨域VPN-OptionC ····· 1-38
1.5 配置嵌套VPN ····· 1-40
1.5.1 配置准备 ····· 1-41
1.5.2 配置嵌套VPN ····· 1-41
1.6 配置HoVPN ····· 1-42
1.6.1 配置准备 ····· 1-42
1.6.2 配置HoVPN ····· 1-42
1.7 配置OSPF伪连接 ····· 1-43

1.7.1 配置准备 ····· 1-43
1.7.2 配置Loopback接口 ····· 1-43
1.7.3 配置BGP引入Loopback接口地址和OSPF的路由 ····· 1-44
1.7.4 配置伪连接 ····· 1-44
1.8 配置BGP的AS号替换和SoO属性 ····· 1-45
1.8.1 配置准备 ····· 1-45
1.8.2 配置BGP的AS号替换和SoO属性 ····· 1-45
1.9 MPLS L3VPN显示和维护 ····· 1-46
1.9.1 复位BGP连接 ····· 1-46
1.9.2 显示MPLS L3VPN的运行状态 ····· 1-46
1.10 MPLS L3VPN典型配置举例 ····· 1-48
1.10.1 PE-CE间使用EBGP基本MPLS L3VPN配置示例 ····· 1-48
1.10.2 PE-CE间使用IBGP基本MPLS L3VPN配置示例 ····· 1-56
1.10.3 配置Hub&Spoke组网示例 ····· 1-63
1.10.4 配置跨域VPN-OptionA方式示例 ····· 1-71
1.10.5 配置跨域VPN-OptionB方式示例 ····· 1-76
1.10.6 配置跨域VPN-OptionC方式示例 ····· 1-81
1.10.7 配置运营商的运营商示例 ····· 1-87
1.10.8 配置嵌套VPN示例 ····· 1-95
1.10.9 配置HoVPN示例 ····· 1-105
1.10.10 配置OSPF伪连接 ····· 1-112
1.10.11 配置BGP的AS号替换 ····· 1-117
1.10.12 配置BGP的AS号替换和SoO属性 ····· 1-120
2 IPv6 MPLS L3VPN配置 ····· 2-1
2.1 IPv6 MPLS L3VPN简介 ····· 2-1
2.1.1 IPv6 MPLS L3VPN概述 ····· 2-1
2.1.2 IPv6 MPLS L3VPN的报文转发 ····· 2-2
2.1.3 IPv6 MPLS L3VPN的路由发布 ····· 2-2
2.1.4 IPv6 MPLS L3VPN支持的组网方案及功能 ····· 2-3
2.2 IPv6 MPLS L3VPN配置任务简介 ····· 2-3
2.3 配置基本IPv6 MPLS L3VPN ····· 2-3
2.3.1 基本IPv6 MPLS L3VPN配置任务简介 ····· 2-3
2.3.2 配置准备 ····· 2-4
2.3.3 配置VPN实例 ····· 2-4
2.3.4 配置PE-CE间的路由交换 ····· 2-7
2.3.5 配置PE-PE间的路由交换 ····· 2-11

2.3.6 配置BGP-VPNv6 子地址族下的路由特性 ····· 2-11
2.4 配置IPv6 跨域VPN ····· 2-13
2.4.1 配置准备 ····· 2-13
2.4.2 配置IPv6 跨域VPN-OptionA ····· 2-14
2.4.3 配置IPv6 跨域VPN-OptionC ····· 2-14
2.5 IPv6 MPLS L3VPN显示和维护 ····· 2-15
2.5.1 复位BGP连接 ····· 2-15
2.5.2 显示IPv6 MPLS L3VPN的运行状态 ····· 2-16
2.6 IPv6 MPLS L3VPN典型配置举例 ····· 2-16
2.6.1 配置IPv6 MPLS L3VPN示例 ····· 2-16
2.6.2 配置IPv6 跨域VPN-OptionA方式示例 ····· 2-24
2.6.3 配置IPv6 跨域VPN-OptionC方式示例 ····· 2-29
2.6.4 配置运营商的运营商示例 ····· 2-36

1 MPLS L3VPN配置

说明

- 本章所指的路由器代表了一般意义下的路由器，以及运行了路由协议的三层交换机。为提高可读性，在手册的描述中将不另行说明。
- 本章只涉及MPLS L3VPN的介绍和配置，有关MPLS基本配置的介绍请参见“MPLS基本配置”，有关BGP的介绍请参见“三层技术-IP路由配置指导”中的“BGP”。
- 本系列交换机中S5500-28SC-HI和S5500-52SC-HI不支持MPLS L3VPN功能。

1.1 MPLS L3VPN简介

1.1.1 MPLS L3VPN概述

MPLS L3VPN是服务提供商VPN解决方案中一种基于PE的L3VPN技术，它使用BGP在服务提供商骨干网上发布VPN路由，使用MPLS在服务提供商骨干网上转发VPN报文。

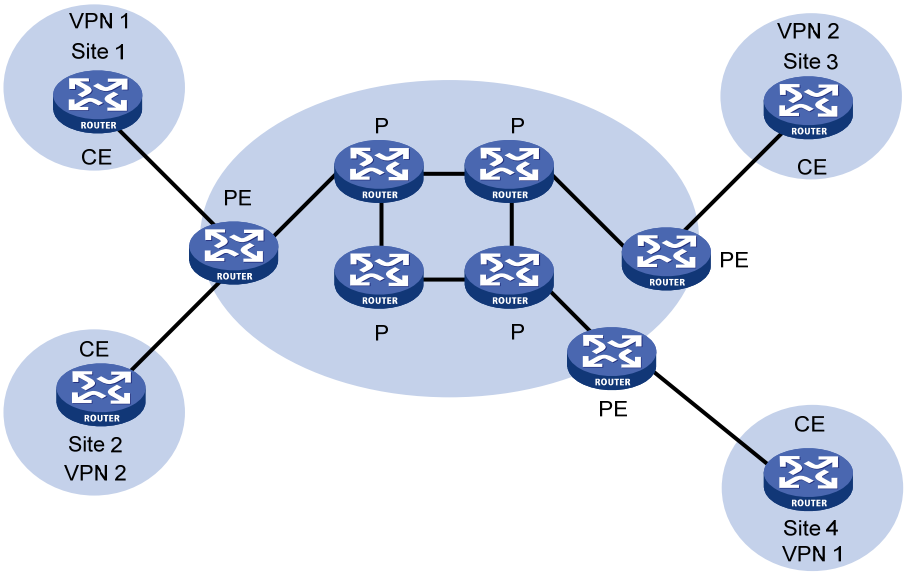
MPLS L3VPN组网方式灵活、可扩展性好，并能够方便地支持MPLS QoS和MPLS TE，因此得到越来越多的应用。

MPLS L3VPN模型由三部分组成：

- CE（Customer Edge）设备：用户网络边缘设备，有接口直接与SP（Service Provider，服务提供商）相连。CE可以是路由器或交换机，也可以是一台主机。CE“感知”不到VPN的存在，也不需要必须支持MPLS。
- PE（Provider Edge）设备：服务提供商网络的边缘设备，与用户的CE直接相连。在MPLS网络中，对VPN的所有处理都发生在PE上。
- P（Provider）设备：服务提供商网络中的骨干设备，不与CE直接相连。P设备只需要具备基本MPLS转发能力。

图 1-1是一个MPLS L3VPN组网方案的示意图。

图1-1 MPLS L3VPN 组网



CE 和 PE 的划分主要是根据服务提供商与用户的管理范围，CE 和 PE 是两者管理范围的边界。

CE 设备通常是一台路由器，当 CE 与直接相连的 PE 建立邻接关系后，CE 把本站点的 VPN 路由发布给 PE，并从 PE 学到远端 VPN 的路由。CE 与 PE 之间使用 BGP/IGP 交换路由信息，也可以使用静态路由。

PE 从 CE 学到 CE 本地的 VPN 路由信息后，通过 BGP 与其它 PE 交换 VPN 路由信息。PE 路由器只维护与它直接相连的 VPN 的路由信息，不维护服务提供商网络中的所有 VPN 路由。

P 路由器只维护到 PE 的路由，不需要了解任何 VPN 路由信息。

当在 MPLS 骨干网上传输 VPN 流量时，入口 PE 做为 Ingress LSR（Label Switch Router，标签交换路由器），出口 PE 做为 Egress LSR，P 路由器则做为 Transit LSR。

1.1.2 MPLS L3VPN的基本概念

1. Site

在介绍 VPN 时经常会提到“Site”，Site（站点）的含义可以从下述几个方面理解：

- Site 是指相互之间具备 IP 连通性的一组 IP 系统，并且，这组 IP 系统的 IP 连通性不需通过服务提供商网络实现；
- Site 的划分是根据设备的拓扑关系，而不是地理位置，尽管在大多数情况下一个 Site 中的设备地理位置相邻；
- 一个 Site 中的设备可以属于多个 VPN，换言之，一个 Site 可以属于多个 VPN；
- Site 通过 CE 连接到服务提供商网络，一个 Site 可以包含多个 CE，但一个 CE 只属于一个 Site。

对于多个连接到同一服务提供商网络的 Sites，通过制定策略，可以将它们划分为不同的集合（set），只有属于相同集合的 Sites 之间才能通过服务提供商网络互访，这种集合就是 VPN。

2. 地址空间重叠

VPN 是一种私有网络，不同的 VPN 独立管理自己使用的地址范围，也称为地址空间（Address Space）。

不同 VPN 的地址空间可能会在一定范围内重合，比如，VPN 1 和 VPN 2 都使用了 10.110.10.0/24 网段的地址，这就发生了地址空间重叠（Overlapping Address Spaces）。

3. VPN实例

在 MPLS VPN 中，不同 VPN 之间的路由隔离通过 VPN 实例（VPN-instance）实现。

PE 为直接相连的 Site 建立并维护 VPN 实例。VPN 实例中包含对应 Site 的 VPN 成员关系和路由规则。如果一个 Site 中的用户同时属于多个 VPN，则该 Site 的 VPN 实例中将包括所有这些 VPN 的信息。

为保证 VPN 数据的独立性和安全性，PE 上每个 VPN 实例都有相对独立的路由表和 LFIB（Label Forwarding Information Base，标签转发表）。

具体来说，VPN 实例中的信息包括：标签转发表、IP 路由表、与 VPN 实例绑定的接口以及 VPN 实例的管理信息。VPN 实例的管理信息包括 RD（Route Distinguisher，路由标识符）、路由过滤策略、成员接口列表等。

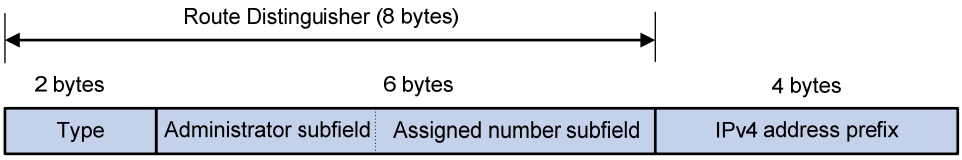
4. VPN-IPv4 地址

传统 BGP 无法正确处理地址空间重叠的 VPN 的路由。假设 VPN 1 和 VPN 2 都使用了 10.110.10.0/24 网段的地址，并各自发布了一条去往此网段的路由，BGP 将只会选择其中一条路由，从而导致去往另一个 VPN 的路由丢失。

PE 路由器之间使用 MP-BGP 来发布 VPN 路由，并使用 VPN-IPv4 地址族来解决上述问题。

VPN-IPv4 地址共有 12 个字节，包括 8 字节的RD和 4 字节的IPv4 地址前缀，如 图 1-2 所示。

图1-2 VPN-IPv4 地址结构



PE 从 CE 接收到普通 IPv4 路由后，需要将这些私网 VPN 路由发布给对端 PE。私网路由的独立性是通过为这些路由附加 RD 实现的。

SP 可以独立地分配 RD，但必须保证 RD 的全局唯一性。这样，即使来自不同服务提供商的 VPN 使用了同样的 IPv4 地址空间，PE 路由器也可以向各 VPN 发布不同的路由。

建议为 PE 上每个 VPN 实例配置专门的 RD，以保证到达同一 CE 的路由都使用相同的 RD。RD 为 0 的 VPN-IPv4 地址相当于全局唯一的 IPv4 地址。

RD 的作用是添加到一个特定的 IPv4 前缀，使之成为全局唯一的 VPN IPv4 前缀。

RD 或者是与自治系统号（ASN）相关的，在这种情况下，RD 是由一个自治系统号和一个任意的数组成；或者是与 IP 地址相关的，在这种情况下，RD 是由一个 IP 地址和一个任意的数组成。

RD 有三种格式，通过 2 字节的 Type 字段区分：

- Type 为 0 时，Administrator 子字段占 2 字节，Assigned number 子字段占 4 字节，格式为：16bits 自治系统号:32bits 用户自定义数字。例如：100:1
- Type 为 1 时，Administrator 子字段占 4 字节，Assigned number 子字段占 2 字节，格式为：32bitsIPv4 地址:16bits 用户自定义数字。例如：172.1.1.1:1

- Type 为 2 时，Administrator 子字段占 4 字节，Assigned number 子字段占 2 字节，格式为：32bits 自治系统号：16bits 用户自定义数字，其中的自治系统号最小值为 65536。例如：65536:1

为保证 RD 的全局唯一性，建议不要将 Administrator 子字段的值设置为私有 AS 号或私有 IP 地址。

5. BGP扩展团体属性

(1) VPN Target 属性

MPLS L3VPN 使用 BGP 扩展团体属性——VPN Target（也称为 Route Target）来控制 VPN 路由信息的发布。

PE 路由器上的 VPN 实例有两类 VPN Target 属性：

- Export Target 属性：在本地 PE 将从与自己直接相连的 Site 学到的 VPN-IPv4 路由发布给其它 PE 之前，为这些路由设置 Export Target 属性；
- Import Target 属性：PE 在接收到其它 PE 路由器发布的 VPN-IPv4 路由时，检查其 Export Target 属性，只有当此属性与 PE 上 VPN 实例的 Import Target 属性匹配时，才把路由加入到相应的 VPN 路由表中。

也就是说，VPN Target 属性定义了一条 VPN-IPv4 路由可以为哪些 Site 所接收，PE 路由器可以接收哪些 Site 发送来的路由。

与 RD 类似，VPN Target 也有三种格式：

- 16bits 自治系统号:32bits 用户自定义数字，例如：100:1。
- 32bits IPv4 地址:16bits 用户自定义数字，例如：172.1.1.1:1。
- 32bits 自治系统号：16bits 用户自定义数字，其中的自治系统号最小值为 65536。例如：65536:1。

(2) SoO 属性

SoO（Site of Origin，路由源）是 BGP 的一种扩展团体属性，用来标识路由的原始站点。路由器不会将带有 SoO 属性的路由发布给 SoO 指定的站点。AS 路径丢失时，通过 SoO 属性可以防止自己发布的路由被其他邻居转发回来，避免发生环路。

SoO 属性有三种格式：

- 16bits 自治系统号:32bits 用户自定义数字，例如：100:1。
- 32bits IPv4 地址:16bits 用户自定义数字，例如：172.1.1.1:1。
- 32bits 自治系统号:16bits 用户自定义数字，其中的自治系统号最小值为 65536。例如：65536:1。

说明

每条路由只能携带一个 SoO 属性。

---

6. MP-BGP

MP-BGP（Multiprotocol Border Gateway Protocol，多协议边界网关协议）在 PE 路由器之间传播 VPN 组成信息和路由。MP-BGP 向下兼容，既可以支持传统的 IPv4 地址族，又可以支持其他地址族（比如 VPN-IPv4 地址族）。使用 MP-BGP 既确保 VPN 的私网路由只在 VPN 内发布，又实现了 MPLS VPN 成员间的通信。

7. 路由策略（Routing Policy）

在通过入口、出口扩展团体来控制 VPN 路由发布的基础上，如果需要更精确地控制 VPN 路由的引入和发布，可以使用入方向或出方向路由策略。

入方向路由策略根据路由的 VPN Target 属性进一步过滤可引入到 VPN 实例的路由，它可以拒绝接收引入列表中的团体选定的路由，而出方向路由策略则可以拒绝发布输出列表中的团体选定的路由。

VPN 实例创建完成后，可以选择是否需要配置入方向或出方向路由策略。

8. 隧道策略（Tunneling Policy）

隧道策略用于选择给特定 VPN 实例的报文使用的隧道。

隧道策略是可选配的，VPN 实例创建完成后，就可以配置隧道策略。缺省情况下，按照 LSP 隧道—>CR-LSP 隧道的优先级顺序选择隧道，负载分担条数为 1，即不进行负载分担。另外，隧道策略只在同一 AS 域内生效。

1.1.3 MPLS L3VPN的报文转发

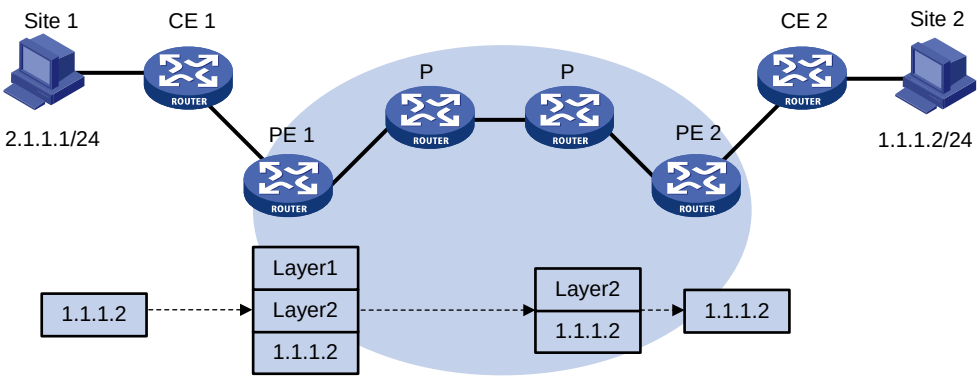
在基本 MPLS L3VPN 应用中（不包括跨域的情况），VPN 报文转发采用两层标签方式：

- 第一层（外层）标签在骨干网内部进行交换，指示从 PE 到对端 PE 的一条 LSP。VPN 报文利用这层标签，可以沿 LSP 到达对端 PE；
- 第二层（内层）标签在从对端 PE 到达 CE 时使用，指示报文应被送到哪个 Site，或者更具体一些，到达哪一个 CE。这样，对端 PE 根据内层标签可以找到转发报文的接口。

特殊情况下，属于同一个 VPN 的两个 Site 连接到同一个 PE，这种情况下只需要知道如何到达对端 CE。

以 图 1-3 为例，说明VPN报文的转发：

图1-3 VPN 报文转发示意图



(1) Site 1 发出一个目的地址为 1.1.1.2 的 IP 报文，由 CE 1 将报文发送至 PE 1。

(2) PE 1 根据报文到达的接口及目的地址查找 VPN 实例表项，匹配后将报文转发出去，同时打上内层和外层两个标签。

(3) MPLS 网络利用报文的外层标签，将报文传送到 PE 2（报文在到达 PE 2 前一跳时已经被剥离外层标签，仅含内层标签）。

(4) PE 2 根据内层标签和目的地址查找 VPN 实例表项，确定报文的出接口，将报文转发至 CE 2。

(5) CE 2 根据正常的 IP 转发过程将报文传送到目的地。

1.1.4 MPLS L3VPN的网络架构

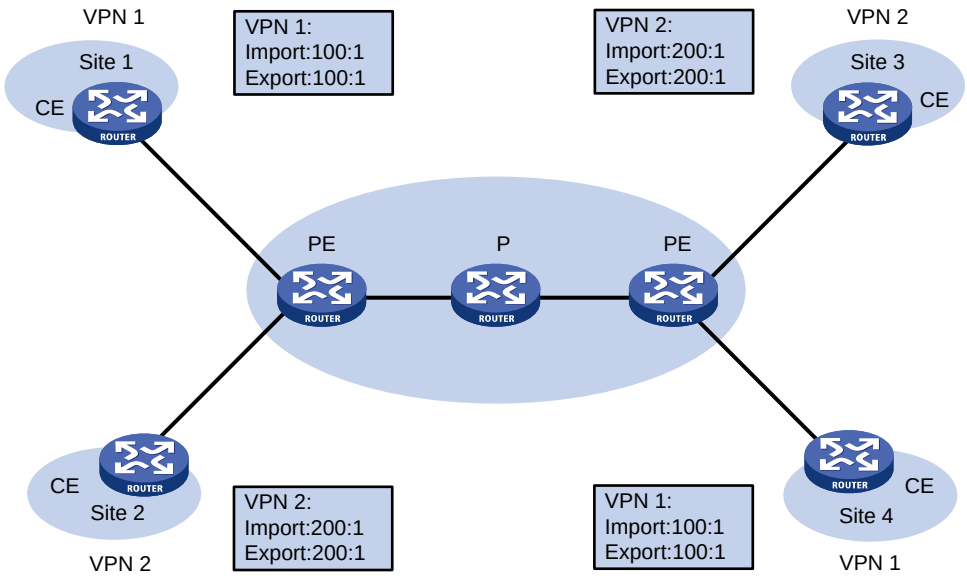
在 MPLS L3VPN 网络中，通过 VPN Target 属性来控制 VPN 路由信息在各 Site 之间的发布和接收。VPN Export Target 和 Import Target 的设置相互独立，并且都可以设置多个值，能够实现灵活的 VPN 访问控制，从而实现多种 VPN 组网方案。

1. 基本的VPN组网方案

最简单的情况下，一个 VPN 中的所有用户形成闭合用户群，相互之间能够进行流量转发，VPN 中的用户不能与任何本 VPN 以外的用户通信。

对于这种组网，需要为每个 VPN 分配一个 VPN Target，作为该 VPN 的 Export Target 和 Import Target，并且，此 VPN Target 不能被其他 VPN 使用。

图1-4 基本的 VPN 组网方案



在 图 1-4 中，PE上为VPN 1 分配的VPN Target值为 100:1，为VPN 2 分配的VPN Target值为 200:1。VPN 1 的两个Site之间可以互访，VPN 2 的两个Site之间也可以互访，但VPN 1 和VPN 2 的Site之间不能互访。

2. Hub&Spoke组网方案

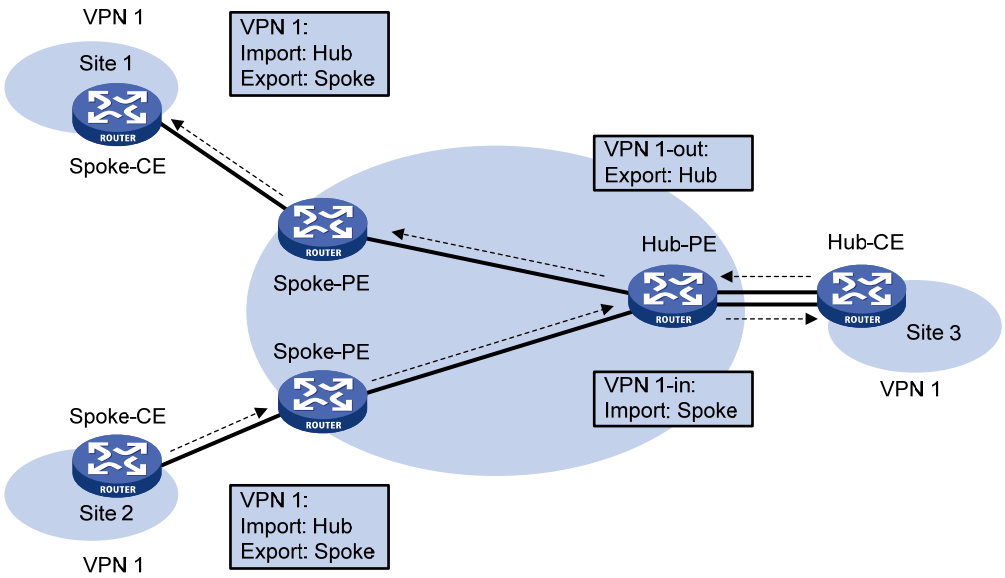
如果希望在 VPN 中设置中心访问控制设备，其它用户的互访都通过中心访问控制设备进行，可以使用 Hub&Spoke 组网方案，从而实现中心设备对两端设备之间的互访进行监控和过滤等功能。

对于这种组网，需要设置两个 VPN Target，一个表示“Hub”，另一个表示“Spoke”。

各 Site 在 PE 上的 VPN 实例的 VPN Target 设置规则为：

- 连接 Spoke 站点（Site 1 和 Site 2）的 Spoke-PE：Export Target 为“Spoke”，Import Target 为“Hub”；
- 连接 Hub 站点（Site 3）的 Hub-PE：Hub-PE 上需要使用两个接口，一个用于接收 Spoke-PE 发来的路由，其 VPN 实例的 Import Target 为“Spoke”；另一个用于向 Spoke-PE 发布路由，其 VPN 实例的 Export Target 为“Hub”。

图1-5 Hub&Spoke 组网方案



在 图 1-5 中，Spoke站点之间的通信通过Hub站点进行（图中箭头所示为Site 2 的路由向Site 1 的发布过程）：

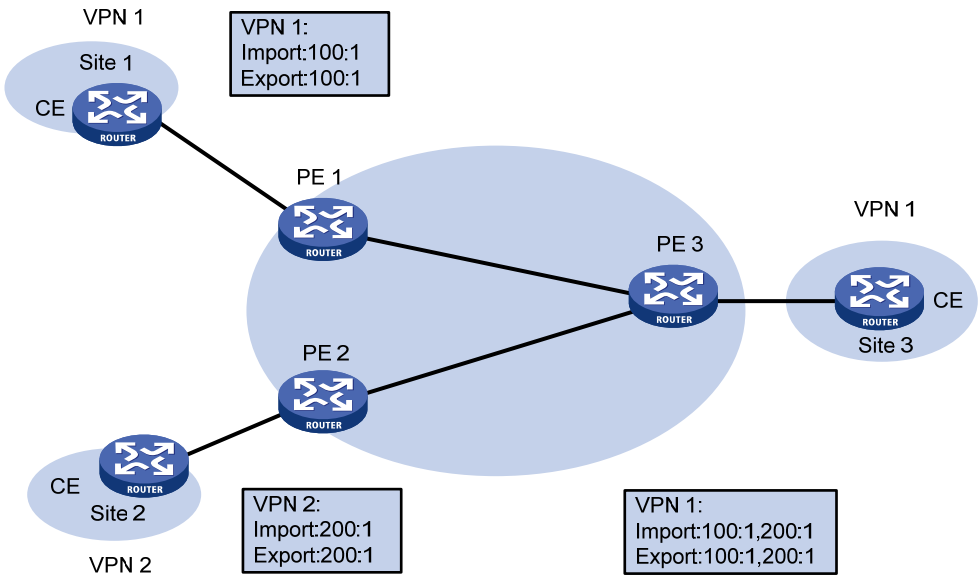
- Hub-PE 能够接收所有 Spoke-PE 发布的 VPN-IPv4 路由；
- Hub-PE 发布的 VPN-IPv4 路由能够为所有 Spoke-PE 接收；
- Hub-PE 将从 Spoke-PE 学到的路由发布给其他 Spoke-PE，因此，Spoke 站点之间可以通过 Hub 站点互访。
- 任意 Spoke-PE 的 Import Target 属性不与其它 Spoke-PE 的 Export Target 属性相同。因此，任意两个 Spoke-PE 之间不直接发布 VPN-IPv4 路由，Spoke 站点之间不能直接互访。

3. Extranet组网方案

如果一个 VPN 用户希望提供部分本 VPN 的站点资源给非本 VPN 的用户访问，可以使用 Extranet 组网方案。

对于这种组网，如果某个 VPN 需要访问共享站点，则该 VPN 的 Export Target 必须包含在共享站点的 VPN 实例的 Import Target 中，而其 Import Target 必须包含在共享站点 VPN 实例的 Export Target 中。

图1-6 Extranet 组网方案



在 图 1-6 中，VPN 1 的Site 3 能够被VPN 1 和VPN 2 访问：

- PE 3 能够接受 PE 1 和 PE 2 发布的 VPN-IPv4 路由；
- PE 3 发布的 VPN-IPv4 路由能够为 PE 1 和 PE 2 接受；
- 基于以上两点，VPN 1 的 Site 1 和 Site 3 之间能够互访，VPN 2 的 Site 2 和 VPN 1 的 Site 3 之间能够互访。

PE 3 不把从 PE 1 接收的 VPN-IPv4 路由发布给 PE 2，也不把从 PE 2 接收的 VPN-IPv4 路由发布给 PE 1（IBGP 邻居学来的条目是不会再发送给别的 IBGP 邻居），因此，VPN 1 的 Site 1 和 VPN 2 的 Site 2 之间不能互访。

1.1.5 MPLS L3VPN的路由信息发布

在基本 MPLS L3VPN 组网中，VPN 路由信息的发布涉及 CE 和 PE，P 路由器只维护骨干网的路由，不需要了解任何 VPN 路由信息。PE 路由器也只维护与它直接相连的 VPN 的路由信息，不维护所有 VPN 路由。因此，MPLS L3VPN 网络具有良好的可扩展性。

VPN 路由信息的发布过程包括三部分：本地 CE 到入口 PE、入口 PE 到出口 PE、出口 PE 到远端 CE。完成这三部分后，本地 CE 与远端 CE 之间将建立可达路由，VPN 私网路由信息能够在骨干网上发布。

下面分别对这三部分进行介绍。

1. 本地CE到入口PE的路由信息交换

CE 与直接相连的 PE 建立邻接关系后，把本站点的 VPN 路由发布给 PE。

CE 与 PE 之间可以使用静态路由、RIP、OSPF、IS-IS、EBGP 或 IBGP。无论使用哪种路由协议，CE 发布给 PE 的都是标准的 IPv4 路由。

2. 入口PE到出口PE的路由信息交换

PE 从 CE 学到 VPN 路由信息后，为这些标准 IPv4 路由增加 RD 和 VPN Target 属性，形成 VPN-IPv4 路由，存放到为 CE 创建的 VPN 实例的路由表中，并触发 MPLS 为其分配私网标签。

入口 PE 通过 MP-BGP 把 VPN-IPv4 路由发布给出口 PE。出口 PE 根据 VPN-IPv4 路由的 Export Target 属性与自己维护的 VPN 实例的 Import Target 属性，决定是否将该路由加入到 VPN 实例的路由表。

PE 之间通过 IGP 来保证内部的连通性。

3. 出口PE到远端CE的路由信息交换

远端 CE 有多种方式可以从出口 PE 学习 VPN 路由，包括静态路由、RIP、OSPF、IS-IS、EBGP 和 IBGP，与本地 CE 到入口 PE 的路由信息交换相同。

1.1.6 跨域VPN

实际组网应用中，某用户一个 VPN 的多个 Site 可能会连接到使用不同 AS 号的多个服务提供商，或者连接到一个服务提供商的多个 AS。这种 VPN 跨越多个自治系统的应用方式被称为跨域 VPN（Multi-AS VPN）。

RFC 2547bis 中提出了三种跨域 VPN 解决方案，分别是：

- VRF-to-VRF：ASBR 间使用 VLAN 接口管理 VPN 路由，也称为 Inter-Provider Option A；
- EBGP Redistribution of labeled VPN-IPv4 routes：ASBR 间通过 MP-EBGP 发布标签 VPN-IPv4 路由，也称为 Inter-Provider Option B；
- Multihop EBGP redistribution of labeled VPN-IPv4 routes：PE 间通过 MP-EBGP 发布标签 VPN-IPv4 路由，也称为 Inter-Provider Option C。

下面逐一对这些方案进行介绍。

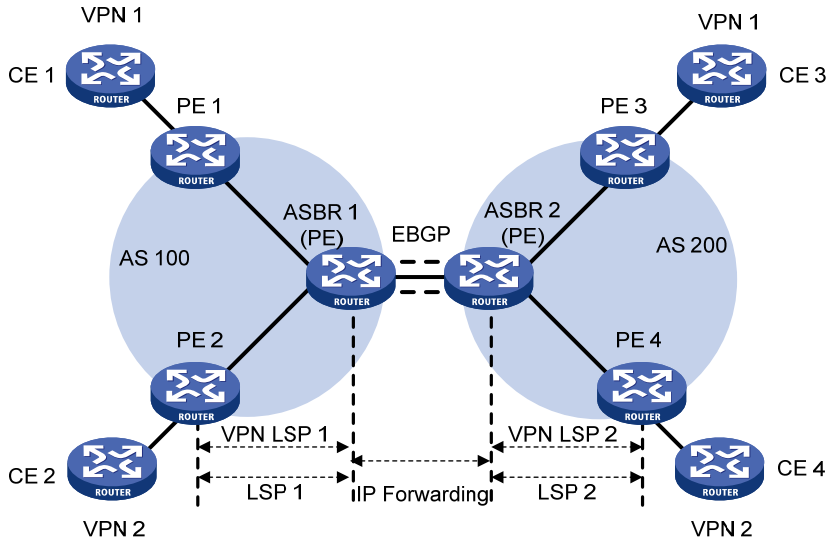
1. ASBR间使用接口管理VPN路由

这种方式下，两个 AS 的 PE 路由器直接相连，PE 路由器同时也是各自所在自治系统的边界路由器 ASBR。

作为 ASBR 的 PE 之间通过接口相连，两个 PE 都把对方作为自己的 CE 设备对待，使用传统的 EBGP 方式向对端发布 IPv4 路由。报文在 AS 内部作为 VPN 报文，采用两层标签转发方式；在 ASBR 之间则采用普通 IP 转发方式。

理想情况下，每个跨域的 VPN 都有一对接口与之对应，用来交换 VPN 路由信息。

图1-7 ASBR 间使用接口管理 VPN 路由组网图



使用接口实现跨域 VPN 的优点是实现简单：两个作为 ASBR 的 PE 之间不需要为跨域进行特殊配置。

缺点是可扩展性差：作为 ASBR 的 PE 需要管理所有 VPN 路由，为每个 VPN 创建 VPN 实例。这将导致 PE 上的 VPN-IPv4 路由数量过于庞大。并且，为每个 VPN 单独创建接口也提高了对 PE 设备的要求。

2. ASBR间通过MP-EBGP发布标签VPN-IPv4 路由

这种方式下，两个 ASBR 通过 MP-EBGP 交换它们从各自 AS 的 PE 路由器接收的标签 VPN-IPv4 路由。

路由发布过程可分为以下步骤：

(1) AS 100 内的 PE 先通过 MP-IBGP 方式把标签 VPN-IPv4 路由发布给 AS 100 的边界路由器 PE，或发布给为 ASBR PE 反射路由的路由反射器；

(2) 作为 ASBR 的 PE 通过 MP-EBGP 方式把标签 VPN-IPv4 路由发布给 AS 200 的 PE（也是 AS 200 的边界路由器）；

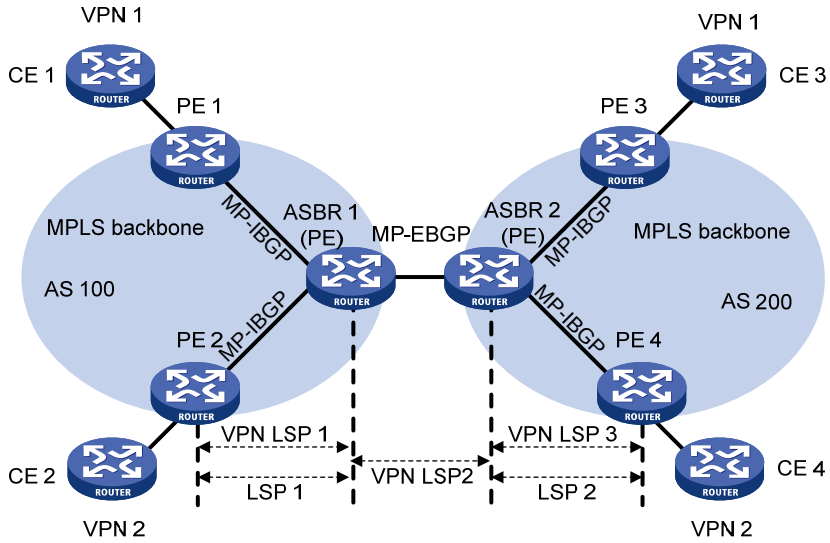
(3) AS 200 的 ASBR PE 再通过 MP-IBGP 方式把标签 VPN-IPv4 路由发布给 AS 200 内的 PE，或发布给为 PE 反射路由的路由反射器。

说明

路由反射器的介绍，请参见“三层技术-IP 路由配置指导”中的“BGP”。

这种方式的 ASBR 需要对标签 VPN-IPv4 路由进行特殊处理，因此也称为 ASBR 扩展方式。

图1-8 ASBR 间通过 MP-EBGP 发布标签 VPN-IPv4 路由组网图



在可扩展性方面，通过 MP-EBGP 发布标签 VPN-IPv4 路由优于 ASBR 间通过 VLAN 接口管理 VPN。

采用 MP-EBGP 方式时，需要注意：

- ASBR 之间不对接收的 VPN-IPv4 路由进行 VPN Target 过滤，因此，交换 VPN-IPv4 路由的各 AS 服务提供商之间需要就这种路由交换达成信任协议；
- VPN-IPv4 路由交换仅发生在私网对等点之间，不能与公网交换 VPN-IPv4 路由，也不能与没有达成信任协议的 MP-EBGP 对等体交换 VPN-IPv4 路由。

3. PE间通过MP-EBGP发布标签VPN-IPv4 路由

前面介绍的两种方式都能够满足跨域 VPN 的组网需求，但这两种方式也都需要 ASBR 参与 VPN-IPv4 路由的维护和发布。当每个 AS 都有大量的 VPN 路由需要交换时，ASBR 就很可能成为阻碍网络进一步扩展的瓶颈。

解决上述可扩展性问题的方案是：ASBR 不维护或发布 VPN-IPv4 路由，PE 之间直接交换 VPN-IPv4 路由。

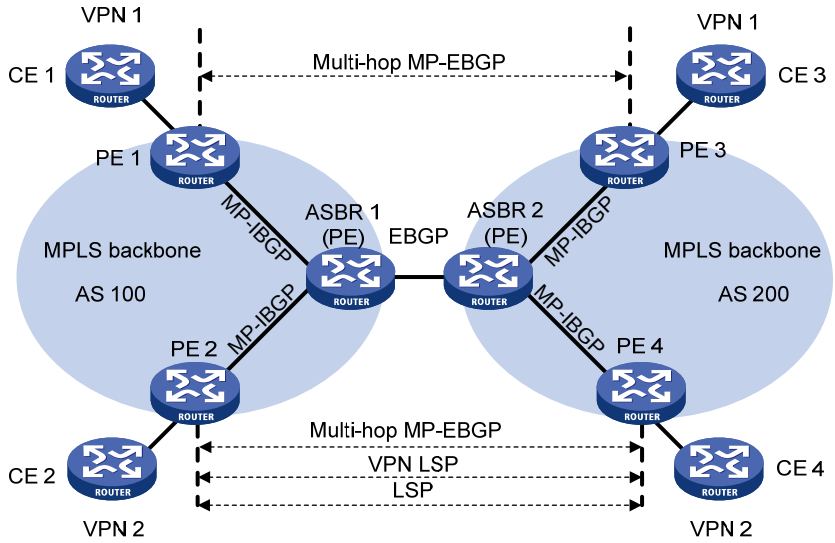
两个 ASBR 通过 MP-IBGP 向各自 AS 内的 PE 路由器发布标签 IPv4 路由。

ASBR 上不保存 VPN-IPv4 路由，相互之间也不通告 VPN-IPv4 路由。

ASBR 保存 AS 内 PE 的带标签的 IPv4 路由，并通告给其它 AS 的对等体。另一个自治系统中的 ASBR 也通告带标签的 IPv4 路由。这样，在入口 PE 和出口 PE 之间建立起一条 LSP。

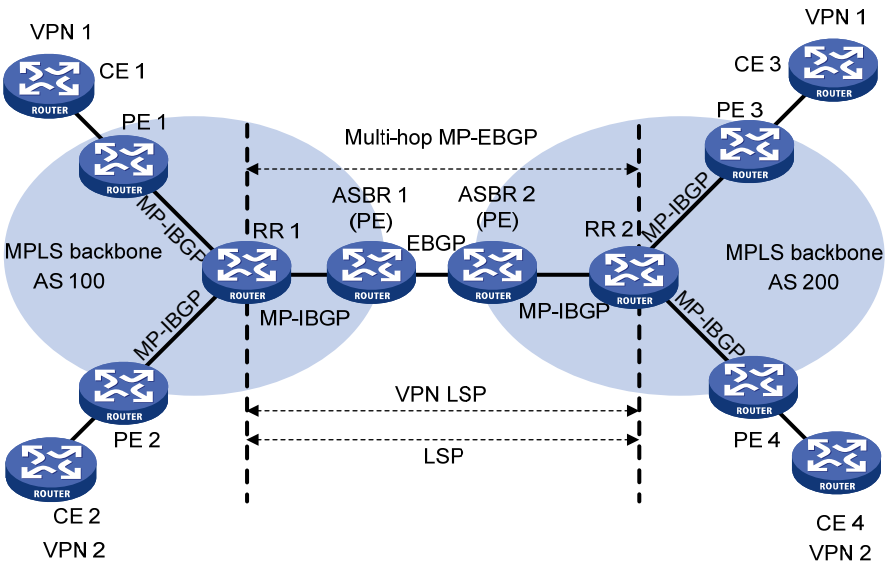
不同 AS 的 PE 之间建立 Multihop 方式的 EBGP 连接，交换 VPN-IPv4 路由。

图1-9 PE 间通过 Multi-hop MP-EBGP 发布标签 VPN-IPv4 路由组网图



为提高可扩展性，可以在每个AS中指定一个路由反射器RR（Route Reflector），由RR保存所有VPN-IPv4 路由，与AS的PE交换VPN-IPv4 路由信息。两个AS的RR之间建立跨域VPNv4 连接，通告VPN-IPv4 路由。如 图 1-10 所示。

图1-10 采用 RR 的跨域 VPN OptionC 方式组网图



1.1.7 运营商的运营商

1. 简介

MPLS L3VPN 服务提供商的用户可能也是一个服务提供商。这种情况下，前者称为提供商运营商（Provider Carrier）或一级运营商（First Carrier），后者称为客户运营商（Customer Carrier）或二级运营商（Second Carrier）。这种组网模型称为运营商的运营商（Carriers' carriers），低级别的 SP 作为更高级别 SP 的 CE。

为保持良好的可扩展性，一级运营商并不引入二级运营商连接的用户网络的路由，只引入为不同站点的二级运营商交换报文的路由。二级运营商连接的用户网络的路由通过在相关二级运营商路由器间建立 BGP 会话交换。这可以大大减少一级运营商网络中需要维护的路由数量。

2. 实现

与普通 MPLS L3VPN 相比，配置运营商的运营商的主要区别在于一级运营商 PE 接入二级运营商 CE 这一部分：

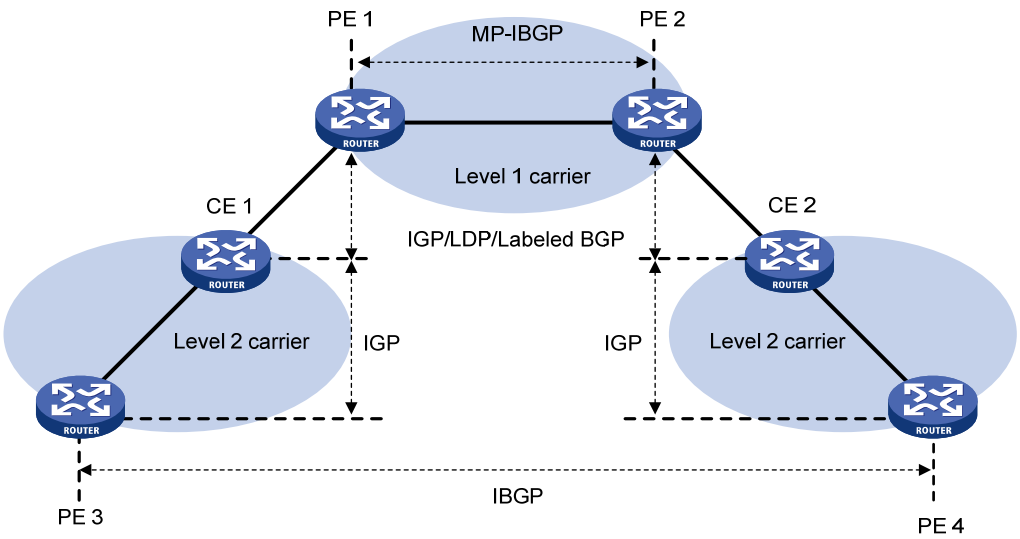
- 如果一级运营商 PE 与二级运营商 CE 位于同一个 AS 内，则它们之间配置 IGP 和 LDP；
- 如果一级运营商 PE 与二级运营商 CE 位于不同的 AS，则它们之间配置 MP-EBGP。MP-EBGP 为 PE 与 CE 之间交换的路由分配标签。

无论哪种情况，CE 上都需要使能 MPLS。并且，CE 上虽然有二级运营商的 VPN 路由，但它们并不把这些路由发布给一级运营商的 PE，只在二级运营商 PE 之间交换。

二级运营商可能只是普通 ISP，也可能是 MPLS L3VPN 服务提供商。

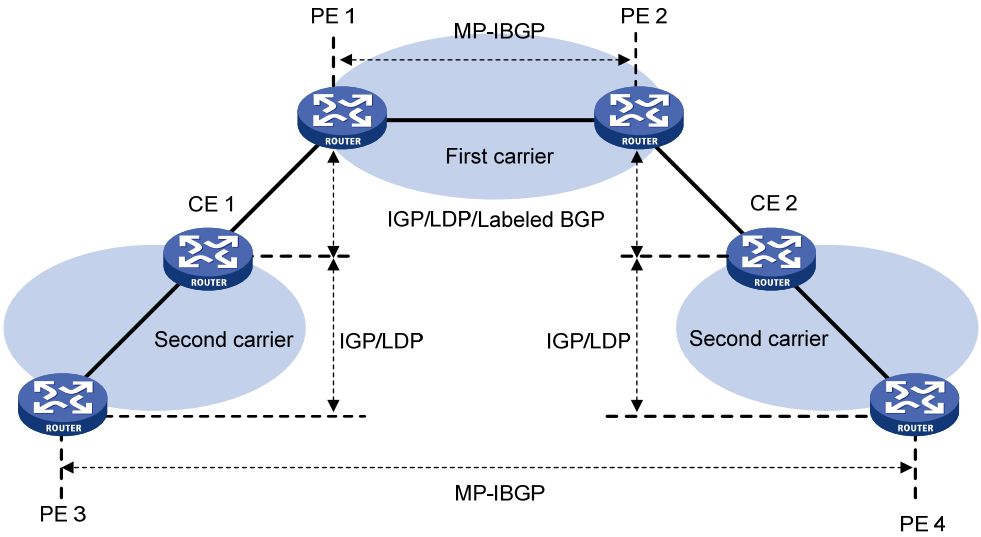
二级运营商是普通ISP时，其PE不需要运行MPLS，与CE之间运行IGP。PE 3 和PE 4 之间通过IBGP会话交换二级运营商的VPN路由。如 图 1-11 所示。

图1-11 二级运营商是 ISP



二级运营商是MPLS L3VPN服务提供商时，其PE也需要运行MPLS，与CE之间运行IGP和LDP。PE 3 和PE 4 之间通过MP-IBGP会话交换二级运营商的VPN路由。如 图 1-12 所示。

图1-12 二级运营商是 MPLS L3VPN 服务提供商



注意

如果一级运营商和二级运营商之间存在等价路由，则建议在一级运营商和二级运营商之间建立对应的等价 LSP。

1.1.8 嵌套VPN

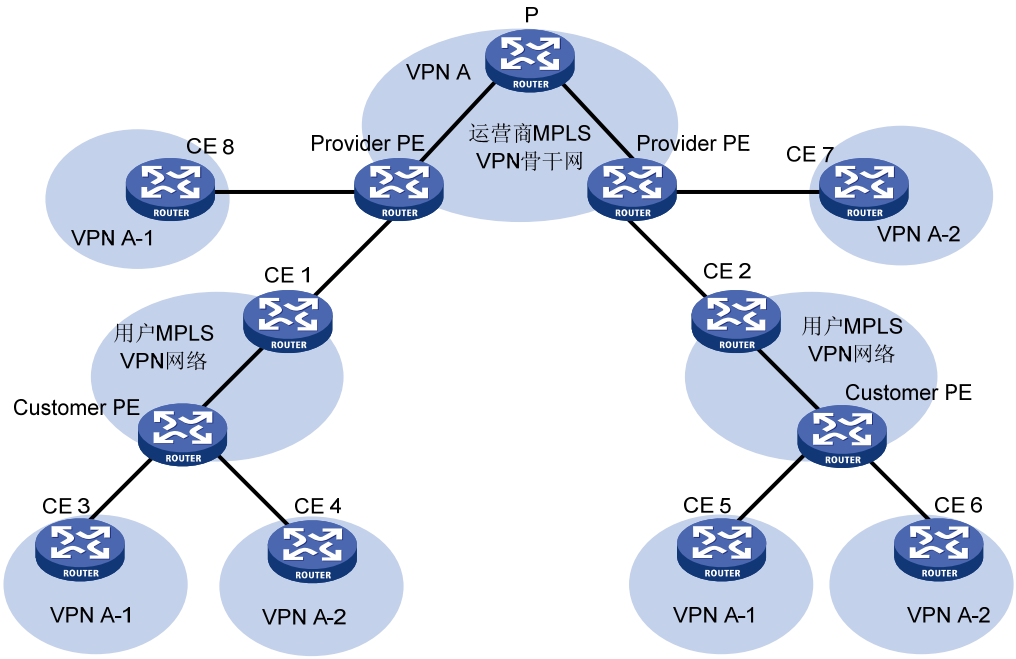
1. 产生背景

在基本的 MPLS L3VPN 模型中，由运营商经营 MPLS VPN 骨干网，通过 PE 设备提供 VPN 服务。VPN 用户通过 CE 设备与运营商的 PE 设备互连，接入 MPLS VPN 网络，实现属于用户 VPN 的不同 Site 之间的通信。在这种情况下，用户网络是普通的 IP 网络，VPN 用户不能再划分子 VPN。

实际组网中，VPN 用户网络复杂多样，繁简不同。当 VPN 用户内部需要进一步划分多个 VPN 时，传统的解决方案是直接在运营商的 PE 设备上配置用户内部的 VPN。这种方案实现简单，缺点是将导致 PE 上承载的 VPN 数量急剧增加，并且，如果用户内部需要调整 VPN 关系，也必须通过运营商操作。这不仅增加了网络的运营成本，也带来了管理和安全方面的问题。

嵌套VPN是一种更为完善的解决方案，它的主要思想是在普通MPLS L3VPN的PE和CE之间传递VPNv4 路由，由用户管理自己内部的VPN划分，运营商不参与用户内部VPN的管理。如 图 1-13 所示，用户在运营商MPLS VPN网络上所属的VPN为VPN A，在用户网络内部划分子VPN：VPN A-1和VPN A-2。运营商PE设备把用户网络当作普通VPN用户对待，不参与子VPN的划分。VPN用户的CE设备（CE 1、CE 2、CE 7 和CE 8）和运营商PE设备之间传递包括子VPN路由信息的VPNv4 路由，从而实现用户网络内部子VPN路由信息的传递。

图1-13 嵌套 VPN 组网应用



2. 路由信息的传播过程

嵌套 VPN 中，路由信息的传播过程为：

(1) 运营商的 PE 设备与 VPN 用户的 CE 设备交互 VPNv4 路由，VPNv4 路由携带用户内部的 VPN 信息。

(2) 运营商的 PE 设备收到 VPNv4 路由后，保留用户内部的 VPN 信息，并附加用户在运营商网络上的 MPLS VPN 属性，即将该 VPNv4 路由的 RD 更换为用户所处运营商网络 VPN 的 RD，同时将用户所处运营商网络 VPN 的 ERT（Export Route-target，引出路由标识）添加到路由的扩展团体属性列表中。运营商的 PE 设备维护用户内部的 VPN 信息。

(3) 运营商的 PE 设备向其他运营商 PE 设备发布这些携带综合 VPN 信息的 VPNv4 路由。

(4) 其他的运营商PE设备收到VPNv4 路由后，与本地的VPN进行匹配，每个VPN接收属于自己的路由，然后将路由扩散到自己连接的子VPNCE设备（如 图 1-13 中的CE 3、CE 4、CE 5 和 CE 6）：如果运营商PE和CE设备之间是IPv4 连接，直接扩散IPv4 路由；如果和CE设备之间是VPNv4 连接，表示通过私网连接的是一个用户MPLS VPN网络，向CE扩散VPNv4 路由。

3. 技术优点

嵌套 VPN 技术的主要优点是：

- 实现了 VPN 聚合功能，可以把用户的多个内部 VPN 聚合成一个用户 VPN，接入运营商的 MPLS VPN 网络；
- 支持对称组网方式和非对称组网方式，即属于同一 VPN 用户的不同 Site 包括的用户内部 VPN 数目可以相同，也可以不同；
- 支持用户内部 VPN 的多层嵌套。

嵌套 VPN 技术降低了用户接入 VPN 网络的复杂度和成本，为用户提供多样化的 VPN 组网方式，实现了用户对内部 VPN 以及多层 VPN 之间的互访权限控制管理。

1.1.9 HoVPN

1. 产生背景

(1) 分层模型与平面模型

在 MPLS L3VPN 解决方案中，PE 设备最为关键，它完成两方面的功能：首先是为用户提供接入功能，这需要 PE 具有大量接口；然后是管理和发布 VPN 路由，处理用户报文，这需要 PE 设备具有大容量存储和高转发能力。

目前的网络设计大多采用经典的分层结构，例如，城域网的典型结构是三层模型：核心层、汇聚层、接入层。从核心层到接入层，对设备的性能要求依次下降，网络的规模则依次扩大。

而 MPLS L3VPN 是一种平面模型，对网络中所有 PE 设备的性能要求相同，当网络中某些 PE 在性能和可扩展性方面存在问题时，整个网络的性能和可扩展性将受到影响。

由于 MPLS L3VPN 的平面模型与典型的分层网络模型不相符，在每一个层次上部署 PE 都会遇到扩展性问题，不利于大规模部署 VPN。

(2) HoVPN

为解决可扩展性问题，MPLS L3VPN 必然要从平面模型转变为分层模型。

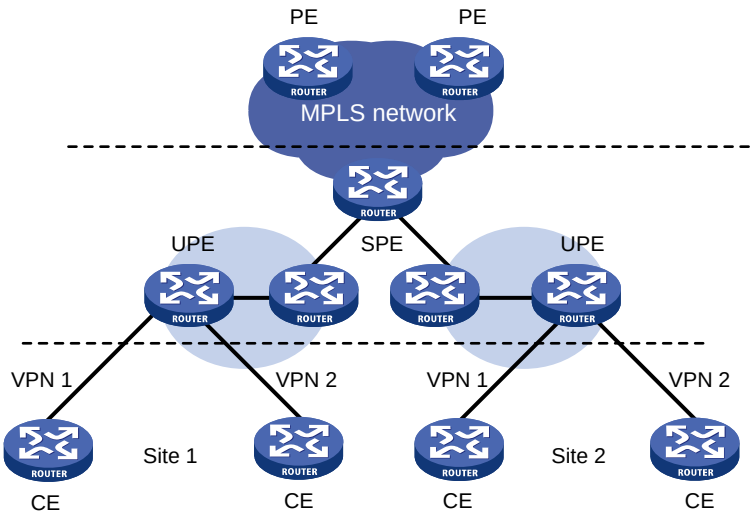
在 MPLS L3VPN 领域，分层 VPN（Hierarchy of VPN，简称 HoVPN）解决方案的提出，实现了将 PE 的功能分布到多个 PE 设备上，多个 PE 承担不同的角色，并形成层次结构，共同完成一个 PE 的功能。

HoVPN 对处于较高层次的设备的路由能力和转发性能要求较高，而对处于较低层次的设备的相应要求也较低，符合典型的分层网络模型。

2. HoVPN的实现

(1) HoVPN 的基本结构

图1-14 HoVPN 的基本结构



在 图 1-14 中，直接连结用户的设备称为UPE（Underlayer PE or User-end PE，下层PE或用户侧PE）；连结UPE并位于网络内部的设备称为SPE（Superstratum PE or Sevice Provider-end PE，上层PE或运营商侧PE）。

多个 UPE 与 SPE 构成分层式 PE，共同完成传统上一个 PE 的功能。

说明

HoVPN 把 PE 的功能分层实现，因此，这种解决方案有时也被称为 HoPE（Hiberarchy of PE，分层 PE）。

SPE 与 UPE 的分工是：

- UPE 主要完成用户接入功能。UPE 维护其直接相连的 VPN Site 的路由，但不维护 VPN 中其它远程 Site 的路由或仅维护它们的聚合路由；UPE 为其直接相连的 Site 的路由分配内层标签，并通过 MP-BGP 随 VPN 路由发布此标签给 SPE；
- SPE 主要完成 VPN 路由的管理和发布。SPE 维护其通过 UPE 连接的 VPN 所有路由，包括本地和远程 Site 的路由，SPE 将路由信息发布给 UPE，并携带标签。SPE 发布的路由信息可以是 VPN 实例的缺省路由（或聚合路由），也可以是通过路由策略的路由信息。通过后者可以实现对同一 VPN 下不同站点之间互访的控制。

由于分工的不同，对 SPE 和 UPE 的要求也不同：SPE 的路由表容量大，转发性能强，但接口资源较少；UPE 的路由容量和转发性能较低，但接入能力强。HoVPN 充分利用了 SPE 的性能和 UPE 的接入能力。

需要说明的是，SPE 和 UPE 是相对的概念。在多个层次的 PE 结构中，上层 PE 相对于下层就是 SPE，下层 PE 相对于上层就是 UPE。

分层式 PE 从外部来看同传统上的 PE 没有区别，可以同普通 PE 共存于一个 MPLS 网络。

(2) SPE-UPE

SPE 和 UPE 之间运行 MP-BGP，可以是 MP-IBGP，也可以是 MP-EBGP，这取决于 UPE 和 SPE 是否属于同一个 AS。

采用 MP-IBGP 时，为了在 IBGP 对等体之间通告路由，SPE 将作为路由反射器，把来自 IBGP 对等体 UPE 的 VPN 路由发布给 IBGP 对等体 SPE，但 SPE 不作为其它 PE 的路由反射器。

(3) HoVPN 的嵌套与扩展

HoVPN 支持分层式 PE 的嵌套：

- 一个分层式 PE 可以作为 UPE，同另一个 SPE 组成新的分层式 PE；
- 一个分层式 PE 可以作为 SPE，同多个 UPE 组成新的分层式 PE；
- 以上这种嵌套可以多次进行。

通过分层式 PE 的嵌套，理论上可以将 VPN 无限扩展与延伸。

图1-15 分层式 PE 的嵌套

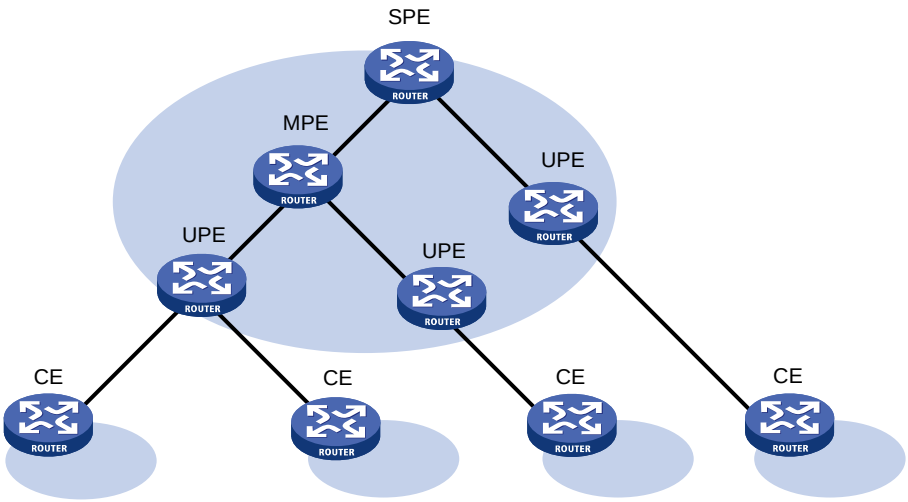


图 1-15 是一个三层的分层式PE，称中间的PE为MPE（Middle-level PE）。SPE和MPE之间，以及MPE和UPE之间，均运行MP-BGP。

说明

“MPE”这种说法只是为了表述方便，在 HoVPN 模型中并没有 MPE。

MP-BGP 为上层 PE 发布下层 PE 上的所有 VPN 路由，为下层 PE 发布上层 PE 的 VPN 实例缺省路由或通过路由策略的 VPN 路由。

SPE 维护了这个分层式 PE 接入的所有 Site 的 VPN 路由，路由数目最多；UPE 只维护它所直接连接的 Site 的 VPN 路由，路由数目最少；MPE 的路由数目介于 SPE 和 UPE 之间。

1.1.10 OSPF VPN扩展

说明

本节重点介绍 OSPF 对 VPN 的扩展，如果需要了解 OSPF 的基本知识，请参见“三层技术-IP 路由配置指导”中的“OSPF 配置”。

1. PE上的OSPF多实例

OSPF 是应用广泛的一种 IGP 协议，很多情况下，VPN 客户端通过 BGP 对等体连接，客户端内部则经常以 OSPF 作为内部路由协议。如果能够在 PE-CE 之间使用 OSPF，则 CE 上就不需要再为到 PE 的连接支持其它路由协议，从而简化 CE 的管理和配置。并且，如果客户需要通过传统的 OSPF 骨干区域提供 MPLS L3VPN 服务，在 PE 和 CE 之间使用 OSPF 可以简化这种转换。

为了在 PE-CE 间运行 OSPF，PE 必须支持 OSPF 多实例，每个 OSPF 实例与一个 VPN 实例对应，使用自己的接口、路由表。

下面具体介绍在 PE-CE 间配置 OSPF 需要了解的知识。

(1)　PE 和 CE 间的 OSPF 区域配置

PE 与 CE 之间的 OSPF 区域可以是非骨干区域，也可以是骨干区域。

在 OSPF VPN 扩展应用中，MPLS VPN 骨干网被看作是骨干区域 area 0。由于 OSPF 要求骨干区域连续，因此，所有 VPN 站点的 area 0 必须与 MPLS VPN 骨干网相连。

即：如果 VPN 站点存在 OSPF area 0，则 CE 接入的 PE 必须通过 area 0 与这个 VPN 站点的骨干区域相连（可以通过 Virtual-link 实现逻辑上的连通）。

(2)　BGP/OSPF 交互

在 PE-CE 间运行 OSPF 后，PE 与 PE 通过 BGP 发布 VPN 路由，PE 通过 OSPF 向 CE 发布 VPN 路由。

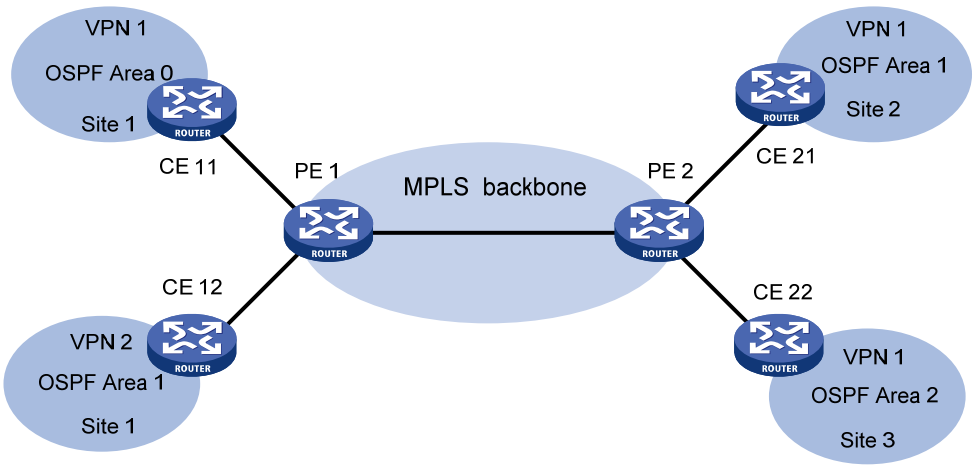
对于普通 OSPF，即使两个不同站点属于同一 VPN，它们也会被看作属于不同的自治系统。这样，在一个站点学到的路由，将被作为外部路由传送给另一站点。这种处理方式导致了比较高的 OSPF 路由协议流量，并带来了一些原本可以避免的网络管理问题。

目前的 OSPF 可以解决上述问题。通过适当配置，运行 OSPF 的不同站点之间彼此看作是直接相连的。这样，PE 路由器交换 OSPF 路由信息时就好像是通过一条专线相连。改善了网络管理并使 OSPF 的应用更为有效。

以 图 1-16 为例，PE 1 和PE 2 通过MPLS骨干网相连，CE 11、CE 21 和CE 22 都属于VPN 1。假设图中所有路由器属于同一个域，即CE 11、CE 21、CE 22 属于同一个OSPF域（OSPF domain）。

VPN 1 路由的发布过程可以描述为：首先在 PE 1 上将 CE 11 的 OSPF 路由引入 BGP；然后通过 BGP 将这些 VPN 路由发布给 PE 2；在 PE 2 上将 BGP 的 VPN 路由引入到 OSPF，再发布给 CE 21 和 CE 22。

图1-16 OSPF 在 VPN 中的使用



如果使用标准的 BGP/OSPF 交互过程，PE 2 将把 BGP VPN 路由通过 Type5 LSAs（即 ASE LSAs）发布给 CE 21 和 CE 22。但 CE 11 与 CE 21、CE 22 是同一个 OSPF 域，它们之间的路由发布应该使用 Type3 LSAs，即区域间路由。

为了解决上述问题，PE 使用一种经过修改的 BGP/OSPF 交互过程（简称为 BGP/OSPF 互操作功能），发布从一个 Site 到另一个 Site 的路由，将这种路由与真正的 AS-External 路由进行区分。这一过程需要 BGP 使用扩展团体属性，携带可以标识 OSPF 属性的信息。

在实现中，要求每个 OSPF 域有一个可配置的域 ID（Domain ID）。一般建议：与每个 VPN 实例相关的网络中的所有 OSPF 实例要么配置一个相同的域 ID，要么都使用缺省的域 ID。这样在收到 BGP 的 VPN 路由时，域 ID 相同的是来自同一 VPN 实例的路由。

(3) 路由环的检测

假设 PE 与 CE 之间通过 OSPF 骨干区域相连，且同一个 VPN 站点（Site）连接到多个不同 PE。这种情况下，当一个 PE 通过 LSA 向 VPN 站点发布从 MPLS/BGP 学的 BGP VPN 路由时，LSA 可能被另一个 PE 接收到，造成路由环。

为了防止产生路由环，对于从 MPLS/BGP 学到的 BGP VPN 路由，无论 PE 与 CE 间是否通过 OSPF 骨干区域相连，PE 在生成 Type3 LSA 时，都会设置标志位 DN。PE 路由器的 OSPF 进程在进行路由计算时，忽略 DN 置位的 Type3 LSAs。

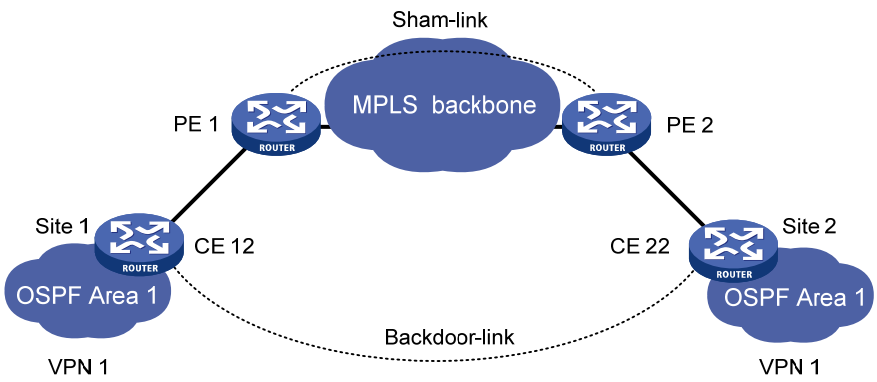
如果 PE 需要向 CE 发布一条来自其它 OSPF 域的路由，则 PE 应表明自己是 ASBR，并将该路由作为 Type5 LSA 发布。

2. OSPF伪连接

通常情况下，BGP 对等体之间通过 BGP 扩展团体属性在 MPLS VPN 骨干网上承载路由信息。另一端 PE 上运行的 OSPF 可利用这些信息来生成 PE 到 CE 的 Type3 summary LSA，这些路由是区域间路由。

如 图 1-17 所示：在OSPF的PE-CE连接中，同一个OSPF区域中有两个Site连接到不同的PE，两个Site属于同一个VPN，它们之间存在一条区域内OSPF链路（backdoor link）。这种情况下，通过PE连接两个Site的路由将作为区域间路由（Inter-Area Route），由于其优先级低于经过backdoor链路的区域内路由（Intra-Area Route），不会被OSPF优选。

图1-17 Sham-link 应用示意图



上述情况导致 VPN 流量总是通过 backdoor 路由转发，而不走骨干网。为了避免这一问题，可以在 PE 路由器之间建立 OSPF 伪连接（Sham-link），使经过 MPLS VPN 骨干网的路由也成为 OSPF 区域内路由。

Sham-link 作为区域内的一条点到点链路，包含在 Type1 LSA 中发布。用户可以通过调整度量值在 Sham-link 和 backdoor 链路之间进行选路。

Sham-link 被看成是两个 VPN 实例之间的链路，每个 VPN 实例中必须有一个 Sham-link 的端点地址，它是 PE 路由器上 VPN 地址空间中的一个有 32 位掩码的 Loopback 接口地址。同一个 OSPF 进程的 Sham-link 可以共用端点地址，但不同 OSPF 进程不能拥有两条端点地址完全相同的 Sham-link。

Sham-link 的端点地址被 BGP 作为 VPN-IPv4 地址发布。如果路由经过了 Sham-link，它就不能再以 VPN-IPv4 路由的形式被引入到 BGP。

Sham-link 可以在任何区域配置。Sham-link 需要手工配置，并且，本端 VPN 实例中必须有到 Sham-link 目的地址的路由。

说明

配置 OSPF 伪连接时，需要在 PE 上配置 BGP 引入 OSPF 的私网路由，但不能配置 OSPF 引入 BGP 路由，否则会引起路由环路。

1.1.11 BGP的AS号替换和SoO属性

在 MPLS L3VPN 中，如果 PE 和 CE 之间运行 EBGP，由于 BGP 使用 AS 号检测路由环路，为保证路由信息的正确发送，需要为物理位置不同的站点分配不同的 AS 号。

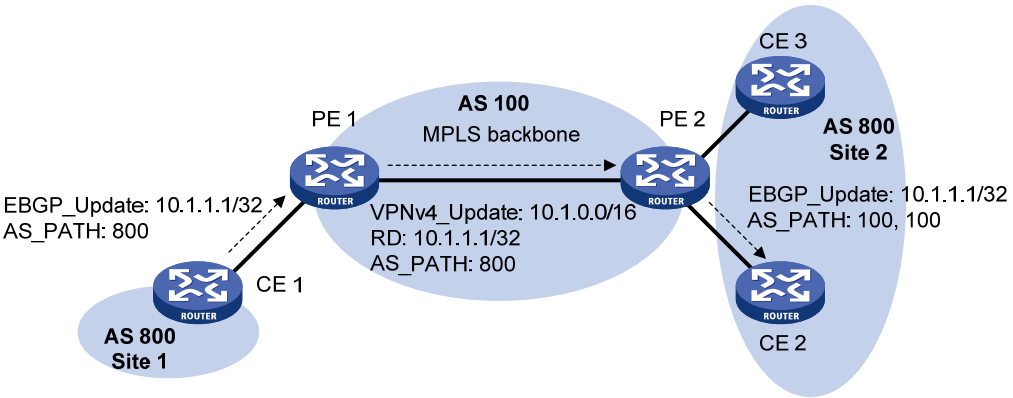
如果物理分散的 CE 复用相同的 AS 号，就应该在 PE 上配置 BGP 的 AS 号替换功能。此功能是 BGP 的出口策略，在发布路由时有效。

使能了 BGP 的 AS 号替换功能后，当 PE 向指定对等体中的 CE 发布路由时，如果路由的 AS_PATH 中有与 CE 相同的 AS 号，将被替换成 PE 的 AS 号后再发布。

说明

使能 BGP 的 AS 号替换功能后，PE 向对等体组中所有已建立连接的 CE 重新发送所有路由，并对发送路由中的 AS_PATH 属性按上述规则替换。

图1-18 BGP AS 号替换和 SoO 属性应用示意图



在 图 1-18 中，CE 1 和CE 2 都使用AS号 800，在PE 2 上使能针对CE 2 的AS号替换功能。当CE 1 发来的Update信息从PE 2 发布给CE 2 时，PE 2 发现AS_PATH中存在与CE 2 相同的AS号 800，就把它替换为自己的AS号 100，这样，CE 2 就可以正确接收CE 1 的路由信息。

PE使用不同接口连接同一站点的多个CE时，如 图 1-18 中的CE 2 和CE 3，使用BGP的AS号替换功能，会导致路由环路。如 图 1-18 所示，使能针对CE 2 的AS号替换功能后，CE 3 发布的路由会通过PE 2 和CE 2 再次发布到Site 2，导致路由环路。此时，通过在PE 2 上对从CE 2 和CE 3 学习

的路由应用路由策略，为学习到的路由添加相同的SoO属性，可以防止PE 2 将CE 3 发布的路由再次通过CE 2 发布给Site 2，避免路由环路。

1.2 MPLS L3VPN配置任务简介

表1-1 MPLS L3VPN 配置任务简介

配置任务	详细配置	说明
配置基本MPLS L3VPN	1.3	完成基本MPLS L3VPN配置后，即可实现通过MPLS技术构建简单的VPN网络 如果部署一些特殊的MPLS L3VPN组网方案，例如跨域VPN、嵌套VPN等，除了配置基本MPLS L3VPN外，还需要根据相应章节进行其他的配置
配置跨域VPN	1.4	
配置嵌套VPN	1.5	
配置HoVPN	1.6	
配置OSPF伪连接	1.7	
配置BGP的AS号替换和SoO属性	1.8	

1.3 配置基本MPLS L3VPN

1.3.1 基本MPLS L3VPN配置任务简介

配置 MPLS L3VPN 的关键在于管理 VPN 私网路由在 MPLS 骨干网上的发布，包括 PE-CE 间的路由发布管理以及 PE-PE 间的路由发布管理。

表1-2 基本 MPLS L3VPN 配置任务简介

配置任务		说明	详细配置
配置VPN实例	创建VPN实例	必选	1.3.3 1.
	配置VPN实例与接口关联	必选	1.3.3 2.
	配置VPN实例的路由相关属性	可选	1.3.3 3.
	配置VPN实例的隧道策略	可选	1.3.3 4.
	配置LDP实例	可选	1.3.3 5.
配置PE-CE间的路由交换		必选	1.3.4
配置PE-PE间的路由交换		必选	1.3.5
配置BGP-VPNv4子地址族下的路由特性		可选	1.3.6

1.3.2 配置准备

在配置基本 MPLS L3VPN 之前，需完成以下任务：

- 对 MPLS 骨干网（PE、P）配置 IGP，实现骨干网的 IP 连通性
- 对 MPLS 骨干网（PE、P）配置 MPLS 基本能力

- 对 MPLS 骨干网（PE、P）配置 MPLS LDP，建立 LDP LSP

1.3.3 配置VPN实例

VPN 实例不仅可以隔离 VPN 私网路由与公网路由，还可以隔离不同 VPN 实例的路由，这一特点使得 VPN 实例的使用不限于 MPLS L3VPN。

配置 VPN 实例的操作是在 PE 或 MCE 设备上进行的。

1. 创建VPN实例

VPN 实例在实现中与 Site 关联。VPN 实例不是直接对应于 VPN，一个 VPN 实例综合了和它所对应 Site 的 VPN 成员关系和路由规则。

一个 VPN 实例只有配置了 RD 后才生效。

描述信息用于描述 VPN 实例，可以用来记录 VPN 实例与某个 VPN 的关系等信息。

表1-3 创建 VPN 实例

操作	命令	说明
进入系统视图	**system-view**	-
创建VPN实例，并进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	必选
配置VPN实例的RD	**route-distinguisher** *route-distinguisher*	必选 一个VPN实例只有配置了RD后才会生效
配置VPN实例的描述信息	**description** *text*	可选 描述信息用于描述VPN实例，可以用来记录VPN实例与某个VPN的关系等信息

2. 配置VPN实例与接口关联

VPN 实例配置完成后，还需要与连接 CE 的接口进行关联。支持 MPLS LDP 能力的接口类型都可以与 VPN 实例相关联。具体接口类型请参见 “MPLS 基本配置”。

说明

- 配置三层聚合接口与 VPN 实例关联时，还需要同时配置其所有成员端口与相同的 VPN 实例关联，否则会影响流量的正常转发。
- 执行 **ip binding vpn-instance** 命令将删除接口上已经配置的 IP 地址，因此需要重新配置接口的 IP 地址。

表1-4 配置 VPN 实例与接口关联

操作	命令	说明
进入系统视图	**system-view**	-

操作	命令	说明
进入要关联接口的接口视图	**interface** *interface-type interface-number*	-
将当前接口与VPN实例关联	**ip binding vpn-instance** *vpn-instance-name*	必选 缺省情况下，接口不关联任何VPN实例

3. 配置VPN实例的路由相关属性

VPN 路由的发布控制过程如下：

- 当从 CE 学习到的一条 VPN 路由引入 BGP 时，BGP 为它关联一个 VPN Target 扩展团体属性列表，通常这个列表是与 CE 相关联的 VPN 实例的输出路由属性列表。
- VPN 实例根据 VPN Target 中 **import-extcommunity** 决定可被接受并引入此 VPN 实例的路由。
- VPN 实例根据 VPN Target 中的 **export-extcommunity** 对向外发布的路由进行 VPN Target 属性的修改。

表1-5 配置 VPN 实例的路由相关属性

操作	命令	说明
进入系统视图	**system-view**	-
进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	-
进入IPv4 VPN视图	**ipv4-family**	可选
配置VPN Target	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	必选
配置支持的最大路由数	**routing-table limit** *number* { *warn-threshold* | **simply-alert** }	可选
应用入方向路由策略	**import route-policy** *route-policy*	可选 缺省情况下，接收所有VPN Target属性匹配的路由
应用出方向路由策略	**export route-policy** *route-policy*	可选 缺省情况下，不对发布的路由进行过滤

说明

- VPN 实例视图下配置的路由相关属性既可以用于 IPv4 VPN，也可以用于 IPv6 VPN。
- 既可以在 VPN 实例视图下，也可以在 IPv4 VPN 视图下，配置 IPv4 VPN 的路由相关属性。如果同时在两个视图下配置了路由相关属性，则 IPv4 VPN 采用 IPv4 VPN 视图下配置的路由相关属性。
- 命令 **vpn-target** 最多可以配置 8 个 VPN Target；一个 VPN 实例最多可以配置 64 个 VPN Target。
- 可以配置一个 VPN 实例支持的最大路由数，以防止 PE 路由器的入接口有过多的路由。一个 PE 路由器最多可以支持的路由条数与具体产品有关。
- 为 VPN 实例配置路由策略之前必须已经创建了路由策略，否则无法对接收和发布的路由进行过滤。

4. 配置VPN实例的隧道策略

在网络中存在多条、多种类型的隧道时，可以通过配置隧道策略对选择的隧道种类和隧道数目进行限制。

通过 **tunnel select-seq** 命令，可以配置隧道的优先级顺序和负载分担条数。

通过 **preferred-path** 命令可以配置优选路径，每条优选路径对应一个隧道接口，指定通过哪个隧道接口转发流量。

应用隧道策略后，隧道的选择方式为：

(1) 从编号最小的优选路径开始匹配，如果对端 PE 地址和任何优选路径的隧道目的地址都不同，则根据 **tunnel select-seq** 命令配置的隧道优先级顺序和负载分担条数选择隧道。如果没有配置 **tunnel select-seq** 命令，则根据缺省的隧道策略选择隧道，即按照 LSP 隧道－>CR-LSP 隧道的优先级顺序选择隧道，负载分担条数为 1。

(2) 如果对端 PE 地址和某条优选路径的隧道目的地址相同，则匹配成功。此优选路径指定的隧道可用时，不再匹配后续的隧道策略规则，使用该优选路径指定的隧道接口转发流量。

(3) 匹配的优选路径指定的隧道不可用时（例如，隧道接口 down、隧道下配置的 ACL 规则不允许转发该流量等原因导致隧道不可用），如果该优选路径中没有指定 **disable-fallback** 参数，则继续匹配后续的隧道策略规则；如果该优选路径中指定了 **disable-fallback** 参数，则不再匹配后续的隧道策略规则，隧道查找失败。

表1-6 配置 VPN 实例的隧道策略

操作	命令	说明
进入系统视图	**system-view**	-
创建隧道策略，并进入隧道策略视图	**tunnel-policy** *tunnel-policy-name*	必选
配置通过隧道接口转发报文的优选路径	**preferred-path** *number* **interface tunnel** *tunnel-number* [**disable-fallback**]	可选 缺省情况下，不存在任何优选路径

操作	命令	说明
配置隧道的优先级顺序和负载分担条数	**tunnel select-seq** { **cr-lsp** | **lsp** } * **load-balance-number** *number*	可选 缺省情况下，按照LSP隧道->CR-LSP隧道的优先级顺序选择隧道，负载分担条数为1
退回系统视图	**quit**	-
进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	必选
进入IPv4 VPN视图	**ipv4-family**	可选
配置VPN实例的隧道策略	**tnl-policy** *tunnel-policy-name*	必选 缺省情况下，按照LSP隧道->CR-LSP隧道的优先级顺序选择隧道，负载分担条数为1

说明：

- 配置的隧道类型不止一种，且负载分担的条数多于单个类型包含的隧道个数时，会使用不同类型的隧道一起进行负载分担。
- 一个隧道策略下最多可以配置 64 条优选路径。不同优选路径对应的隧道接口的目的地址可以相同。隧道接口的隧道类型必须为 MPLS TE。
- 在配置隧道优先级顺序时，隧道类型越靠近关键字 **select-seq** 的，其优先级越高。例如：如果配置了命令 **tunnel select-seq lsp cr-lsp load-balance-number** 1，在没有 LSP 的情况下，VPN 将使用 CR-LSP 隧道，当 LSP 建立后，VPN 将选用 LSP。
- VPN 实例视图下配置的隧道策略既可以用于 IPv4 VPN，也可以用于 IPv6 VPN。
- 既可以在 VPN 实例视图下，也可以在 IPv4 VPN 视图下，配置 IPv4 VPN 的隧道策略。如果同时在两个视图下配置了隧道策略，则 IPv4 VPN 采用 IPv4 VPN 视图下配置的隧道策略。
- 为 VPN 实例配置隧道策略之前必须先创建隧道策略，否则将采用缺省策略，即按照 LSP 隧道->CR-LSP 隧道的优先级顺序选择隧道，负载分担条数为 1。

5. 配置LDP实例

LDP 实例应用于运营商的运营商（Carriers' carriers）组网模型中。

本配置任务用来使能 VPN 实例的 LDP 能力，创建 LDP 实例，并在 MPLS-LDP-VPN 实例视图下为 VPN 实例配置 LDP 参数。

表1-7 配置 LDP 实例

操作	命令	说明
进入系统视图	**system-view**	-
使能指定VPN实例的LDP能力，创建LDP实例，并进入MPLS-LDP-VPN实例视图	**mpls ldp vpn-instance** *vpn-instance-name*	必选 缺省情况下，未使能VPN实例的LDP能力

操作	命令	说明
配置除LDP GR外的其他LDP参数	详细配置请参见“MPLS基本配置”	可选

说明

- 除了 LDP GR 特性外，MPLS LDP 视图下的命令都可以在 MPLS-LDP-VPN 实例视图下使用。MPLS LDP 的详细介绍请参见 “MPLS 基本配置”。
- 对于使能 LDP 能力的接口，MPLS-LDP-VPN 实例视图下的配置只影响绑定 VPN 实例的接口；而 MPLS LDP 视图下的配置对绑定到 VPN 实例的接口没有影响。为 LDP 实例配置传输地址时，也必须使用绑定到 VPN 实例的接口的 IP 地址。
- 对于私网侧的 LDP 邻居关系，缺省情况下是使用使能 LDP 能力的接口地址建立的；而公网上的 LDP 邻居关系，缺省情况下是使用 LDP 实例的 LSR ID 建立的。

1.3.4 配置PE-CE间路由交换

PE-CE 间路由交换可以使用静态路由、RIP、OSPF、IS-IS、EBGP 和 IBGP 路由协议。根据实际组网情况选择一种进行配置即可。

1. 配置准备

在配置 PE-CE 间路由交换之前，需要先完成以下任务：

- 在 CE 上配置连接 PE 接口的 IP 地址
- 在 PE 上配置连接 CE 接口的 IP 地址

2. 配置PE-CE间使用静态路由

表1-8 配置 PE-CE 间使用静态路由

操作	命令	说明
进入系统视图	**system-view**	-
为指定VPN实例配置静态路由	**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*]	必选 该配置在PE上进行，CE上的配置方法与普通静态路由相同
	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*]	

说明

有关静态路由的配置请参见“三层技术-IP 路由配置指导”中的“静态路由”。

3. 配置PE-CE间使用RIP

一个 RIP 进程只能属于一个 VPN 实例。如果在启动 RIP 进程时不绑定到 VPN 实例，则该进程属于公网进程。

表1-9 配置 PE-CE 间使用 RIP

操作	命令	说明
进入系统视图	**system-view**	-
创建PE-CE间的RIP实例，并进入RIP视图	**rip** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在PE上进行，CE上配置普通RIP即可
在指定网段接口上使能RIP	**network** *network-address*	必选 缺省情况下，接口上的RIP功能处于关闭状态

说明

有关 RIP 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“RIP”。

4. 配置PE-CE间使用OSPF

VPN 实例绑定的 OSPF 进程中不使用系统视图下配置的公网 Router ID，因此用户需要在启动进程时手工配置 Router ID，或者所要绑定的 VPN 实例中至少有一个接口配置了 IP 地址。

一个 OSPF 进程只能属于一个 VPN 实例。如果在启动 OSPF 进程时不绑定到 VPN 实例，则该进程属于公网进程。

表1-10 配置 PE-CE 间使用 OSPF

操作	命令	说明
进入系统视图	**system-view**	-
创建PE-CE间的OSPF实例，并进入OSPF视图	**ospf** [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*] *	必选 该配置在PE上进行，CE上配置普通OSPF即可
配置OSPF域标识符	**domain-id** *domain-id* [**secondary**]	可选 缺省情况下，OSPF域标识符为0 该配置在PE上进行，CE上配置普通OSPF即可

操作	命令	说明
配置OSPF扩展团体属性的类型编码	**ext-community-type** { **domain-id** *type-code1* | **router-id** *type-code2* | **route-type** *type-code3* }	可选 缺省情况下，OSPF扩展团体属性Domain ID的类型编码是0x0005，Router ID的类型编码是0x0107，Route Type的类型编码是0x0306 该配置在PE上进行
配置OSPF区域，进入OSPF区域视图	**area** *area-id*	必选 缺省情况下，没有配置OSPF区域
配置区域所包含的网段并在指定网段的接口上使能OSPF	**network** *ip-address wildcard-mask*	必选 缺省情况下，接口不属于任何区域且OSPF功能处于关闭状态

说明

删除 VPN 实例后，相关的所有 OSPF 进程也将全部被删除。

每个 OSPF 进程只能配置一个域 ID，不同进程的域 ID 相互没有影响。不同 VPN 的 PE 的 OSPF 进程域 ID 配置没有限制。但同一 VPN 的所有 OSPF 进程应配置相同的域 ID，以保证路由发布的正确性。

OSPF 进程的域 ID 包含在此进程生成的路由中，在将 OSPF 路由引入 BGP 中时， 域 ID 被附加到 BGP VPN 路由上，作为 BGP 的扩展团体属性传递。

说明

- 配置 OSPF 实例后，需要启动 OSPF，具体配置与普通的 OSPF 相同。
- 有关 OSPF 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“OSPF”。

5. 配置PE-CE间使用IS-IS

一个 IS-IS 进程只能属于一个 VPN 实例。如果在启动 IS-IS 进程时不绑定到 VPN 实例，则该进程属于公网进程。

表1-11 配置 PE-CE 间使用 IS-IS

操作	命令	说明
进入系统视图	**system-view**	-
创建PE-CE间的IS-IS实例，并进入IS-IS视图	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在PE上进行，CE上配置普通IS-IS即可

操作	命令	说明
配置网络实体名称	**network-entity** *net*	必选 缺省情况下，没有配置网络实体名称
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
使能接口IS-IS并指定要关联的IS-IS进程号	**isis enable** [*process-id*]	必选 缺省情况下，接口上没有使能IS-IS

说明

有关 IS-IS 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“IS-IS”。

6. 配置PE-CE间使用EBGP

(1) PE 上的配置

表1-12 PE 上的配置

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	必选
将CE配置为VPN私网EBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
引入本端CE路由	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	必选 PE需要将到本端CE的路由引入VPN路由表中，以发布给对端PE
配置对发布的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct | isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤
配置允许本地AS号在所接收的路由的AS_PATH属性中出现，并可同时配置允许重复的次数	**peer** { *group-name* | *ip-address* } **allow-as-loop** [*number*]	可选 用于Hub&Spoke组网方案

说明

通常情况下，BGP 通过 AS 号检测路由环路。但在 Hub&Spoke 组网方式下，如果在 PE 和 CE 之间运行 EBGP，当 PE 将路由信息通告给 CE 时带上本自治系统的 AS 号，再从 CE 接收路由更新时，路由更新消息中会带有本自治系统的 AS 号，这样 PE 就不能接收这条路由更新信息，此时需要配置允许路由环路。

(2) CE 上的配置

表1-13 CE 上的配置

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
将PE配置为EBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
配置路由引入	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	可选 CE需要将自己所能到达的VPN网段地址发布给接入的PE，通过PE发布给对端CE

说明：

- 配置 BGP VPN 实例后，VPN 实例中的 BGP 路由交换与普通的 BGP 相同。
- BGP 实例视图下配置任务与 BGP 视图下的相同，有关介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“BGP”。
- BGP 的配置中有对等体和对等体组的配置，有关介绍和详细说明请参见“三层技术-IP 路由配置指导”中的“BGP”。本章中不将对等体和对等体组加以区分。

7. 配置PE-CE间使用IBGP

说明

PE 和 CE 之间使用 IBGP 路由协议只适用于普通的 MPLS L3VPN 组网环境，Extranet、跨域 VPN、运营商的运营商、嵌套 VPN 和 HoVPN 等组网中，PE 和 CE 之间不能配置 IBGP。

(1) PE 上的配置

表1-14 PE 上的配置

操作	命令	说明
进入系统视图	**system-view**	-

操作	命令	说明
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	必选
将CE配置为VPN私网IBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
将CE配置为路由反射器的客户端	**peer** { *group-name* | *ip-address* } **reflect-client**	可选 缺省情况下，没有配置路由发射器及其客户端
允许客户到客户的路由反射	**reflect between-clients**	可选 缺省情况下，允许客户到客户的路由反射 如果客户之间是全连接的，则不需要配置路由反射
为路由反射器配置群ID	**reflector cluster-id** { *cluster-id* | *ip-address* }	可选 缺省情况下，每个路由反射器都使用自己的Router ID作为群ID 如果一个群中配置了多个路由反射器，请使用本命令为所有的路由反射器配置相同的群ID，以避免产生路由环路
配置对发布的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct** | **isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤

说明

- 缺省情况下 PE 不会向 IBGP 对等体（包括 VPNv4 的 IBGP 对等体）发送从 IBGP 对等体 CE 学习的路由，只有将 CE 配置为路由反射器的客户端后，才能向其它 IBGP 对等体发送从该 CE 学习的路由。
- 可以在多个视图下执行 **reflect between-clients** 和 **reflector cluster-id** 命令，如 BGP-VPN 实例视图和 BGP-VPNv4 子地址族视图。这两条命令仅对执行此命令的视图下的路由反射器生效，对于其他视图下的路由反射器不生效。
- 配置路由反射器后不会修改路由的下一跳。如果需要修改下一跳，则需在路由的接收端通过入策略进行修改。

(2) CE 上的配置

表1-15 CE 上的配置

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
将PE配置为IBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
配置路由引入	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	可选 CE需要将自己所能到达的VPN网段地址发布给接入的PE，通过PE发布给对端CE

说明：

- 配置 BGP VPN 实例后，VPN 实例中的 BGP 路由交换与普通的 BGP 相同。
- BGP-VPN 实例视图下配置任务与 BGP 视图下的相同，有关介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“BGP”。
- 有关对等体和对等体组的介绍和详细说明请参见“三层技术-IP 路由配置指导”中的“BGP”。本章中不将对等体和对等体组加以区分。

1.3.5 配置PE-PE间的路由交换

以下配置是在 PE 上进行的。

表1-16 配置 PE-PE 间的路由交换

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	必选
将对端PE配置为对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
指定路由更新报文的源接口	**peer** { *group-name* | *ip-address* } **connect-interface** *interface-type interface-number*	必选 缺省情况下，BGP使用最佳路由更新报文的源接口
进入BGP-VPNv4子地址族视图	**ipv4-family vpnv4**	必选
使能对等体交换BGP-VPNv4路由信息	**peer** { *group-name* | *ip-address* } **enable**	必选 缺省情况下，BGP对等体之间只能交换IPv4路由信息

1.3.6 配置BGP-VPNv4 子地址族下的路由特性

BGP-VPNv4 子地址族下的很多路由特性都与 BGP IPv4 单播路由的特性相同。这些特性根据组网的需求来决定是否选择。

1. 配置各种地址族通用路由特性

对于 VPN 的应用，BGP 的地址族包括：BGP-VPNv4 子地址族、BGP-L2VPN 地址族和 VPLS 地址族。对于下面的命令在各地址族中的作用都是相同的，且只对相应地址族中的 BGP 路由起作用。

表1-17 配置各种地址族通用路由特性

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	必选
将对端PE配置为对等体	**peer** *ip-address* **as-number** *as-number*	必选
指定建立TCP连接的接口	**peer** *ip-address* **connect-interface** *interface-type interface-number*	必选
进入地址族视图	**ipv4-family vpnv4**	必选 根据不同的组网情况选择地址族
	l2vpn-family	
	vpls-family	
配置允许本地AS号在所接收的路由的AS_PATH属性中出现，并可同时配置允许重复的次数	**peer** { *group-name* | *ip-address* } **allow-as-loop** [*number*]	可选
在指定的地址族下使能对等体，并使能交换对应地址族的BGP路由信息的能力	**peer** { *group-name* | *ip-address* } **enable**	必选 缺省情况下，BGP对等体之间只能交换IPv4路由信息
将对等体加入已存在的对等体组	**peer** *ip-address* **group** *group-name*	可选
向对等体或对等体组发布路由时，将自身地址作为下一跳	**peer** { *group-name* | *ip-address* } **next-hop-local**	可选 缺省情况下，向EBGP对等体发布路由时会将下一跳改为自己的地址。在跨域VPN OptionC组网中反射器上对多跳EBGP邻居和反射客户需要配置next-hop-invariable以保证私网路由下一跳不会被改变。
配置将本机作为路由反射器，并将对等体或对等体组作为路由反射器的客户	**peer** { *group-name* | *ip-address* } **reflect-client**	可选 缺省情况下，没有配置路由反射器及其客户
配置BGP邻居的ORF（Outbound Route Filtering，输出路由过滤）能力	**peer** { *group-name* | *ip-address* } **capability-advertise orf ip-prefix** { **both** | **receive** | **send** }	可选 缺省情况下，BGP对等体/对等体组不使能邻居的ORF能力

操作	命令	说明
对接收到的相应地址族下BGP路由使能VPN-Target过滤功能	**policy vpn-target**	可选 缺省情况下，对接收到的相应地址族的BGP路由使能VPN-Target过滤功能
允许客户到客户的路由反射	**reflect between-clients**	可选 缺省情况下，允许客户到客户的路由反射
为路由反射器配置群ID	**reflector cluster-id** { *cluster-id* | *ip-address* }	可选 缺省情况下，群ID是群中某个路由反射器的Router ID
创建路由反射器的反射策略	**rr-filter** *extended-community-list-number*	可选 缺省情况下，路由反射器不会对反射的路由进行过滤 执行本命令后，只有与本命令配置的扩展团体属性号匹配的IBGP路由才会被反射 通过在不同的路由反射器上配置不同的反射策略，可以实现路由反射器之间的负载分担

说明

有关 BGP-L2VPN 地址族和 VPLS 地址族的介绍请参见“MPLS 命令参考”中的“MPLS L2VPN”和“VPLS”。

2. 配置BGP-VPNv4 子地址族的特有路由特性

表1-18 配置 BGP-VPNv4 子地址族的特有路由特性

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
将对端PE配置为对等体	**peer** *ip-address* **as-number** *as-number*	必选
指定建立TCP连接的接口	**peer** *ip-address* **connect-interface** *interface-type interface-number*	必选
进入BGP-VPNv4子地址族视图	**ipv4-family vpnv4**	-
配置本地优先级的缺省值	**default local-preference** *value*	可选 缺省情况下，本地优先级的值为100

操作	命令	说明
配置系统MED的缺省值	**default med** *med-value*	可选 缺省情况下，系统MED的缺省值为0
配置对发布的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **export** [**direct | isis** *process-id* | **ospf** *process-id* | **rip** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl-number* | **ip-prefix** *ip-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤
配置将团体属性通告给对等体或对等体组	**peer** { *group-name* | *ip-address* } **advertise-community**	可选 缺省情况下，不将团体属性传给任何对等体或对等体组
为指定对等体或对等体组配置基于AS路径列表的路由过滤	**peer** { *group-name* | *ip-address* } **as-path-acl** *aspath-filter-number* { **import** | **export** }	可选 缺省情况下，对等体及对等体组没有应用AS过滤列表
配置向对等体或对等体组发布指定VPN实例的缺省路由	**peer** { *group-name* | *ip-address* } **default-route-advertise vpn-instance** *vpn-instance-name*	可选 缺省情况下，不向对等体或对等体组发布缺省路由
配置为指定对等体或对等体组应用过滤策略	**peer** { *group-name* | *ip-address* } **filter-policy** *acl-number* { **export** | **import** }	可选 缺省情况下，没有为对等体或对等体组应用过滤策略
为指定对等体或对等体组应用基于IP前缀列表的路由过滤策略	**peer** { *group-name* | *ip-address* } **ip-prefix** *prefix-name* { **export** | **import** }	可选 缺省情况下，对等体或对等体组上不应用基于IP前缀列表的路由过滤策略
配置向EBGP对等体发布路由时不改变下一跳	**peer** { *group-name* | *ip-address* } **next-hop-invariable**	可选 缺省情况下，向EBGP对等体发布路由时会将下一跳改为自己的地址
为从对等体/对等体组接收的路由分配首选值	**peer** { *group-name* | *ip-address* } **preferred-value** *value*	可选 缺省情况下，从对等体/对等体组接收的路由的首选值为0
配置发送BGP更新报文时不携带私有自治系统号	**peer** { *group-name* | *ip-address* } **public-as-only**	可选 缺省情况下，发送BGP更新报文时，携带私有自治系统号
为指定对等体或对等体组配置路由策略	**peer** { *group-name* | *ip-address* } **route-policy** *route-policy-name* { **export** | **import** }	可选 缺省情况下，对等体及对等体组没有应用路由策略

说明

有关 BGP 的路由特性请参见“三层技术-IP 路由配置指导”中的“BGP”。

1.4 配置跨域VPN

如果承载 VPN 路由的 MPLS 骨干网跨越多个 AS，就需要配置跨域 VPN。

跨域VPN有三种解决方案，详细描述请参见“1.1.6 跨域VPN”。

1.4.1 配置准备

在配置跨域 VPN 之前，需完成以下任务：

- 为各 AS 的 MPLS 骨干网分别配置 IGP，实现同一 AS 内骨干网的 IP 连通性
- 为各 AS 的 MPLS 骨干网分别配置 MPLS 基本能力
- 为各 AS 的 MPLS 骨干网分别配置 MPLS LDP，建立 LDP LSP
- 为各 AS 分别进行基本 MPLS L3VPN 配置

在对各 AS 分别进行基本 MPLS L3VPN 配置时，根据所选择的跨域 VPN 方式，PE 和 ASBR-PE 上还需要进行专门的配置。

说明

请根据实际应用的跨域 VPN 方式，选择下述配置任务中的对应项。

1.4.2 配置跨域VPN-OptionA

跨域 VPN-OptionA 的实现比较简单，当 PE 上的 VPN 数量及 VPN 路由数量都比较少时可以采用这种方案。跨域 VPN-OptionA 的配置可以描述为：对各 AS 分别进行基本 MPLS L3VPN 配置，对于 ASBR-PE，将对端 ASBR-PE 看作自己的 CE 配置即可。即：跨域 VPN-OptionA 方式需要在 PE 和 ASBR-PE 上分别配置 VPN 实例，前者用于接入 CE，后者用于接入对端 ASBR-PE。

请参见“1.3 配置基本MPLS L3VPN”。

说明：

在跨域 VPN-OptionA 方式中，对于同一个 VPN，同一 AS 内的 ASBR-PE 和 PE 上配置的 VPN Target 应能匹配，即 VPN Target 的配置应能保证 PE（或 ASBR-PE）发送的 VPN 路由能够被 ASBR-PE（或 PE）接受；不同 AS 的 PE 上配置的 VPN Target 则不需要匹配。

1.4.3 配置跨域VPN-OptionB

以下配置是在作为 ASBR-PE 的路由器上进行的。

表1-19 配置跨域 VPN-OptionB 的 ASBR-PE

操作	命令	说明
进入系统视图	**system-view**	-
进入连接对端ASBR-PE接口的接口视图	**interface** *interface-type interface-number*	-
配置接口IP地址	**ip address** *ip-address* { *mask* | *mask-length* }	必选
退回系统视图	**quit**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPNv4子地址族视图	**ipv4-family vpnv4**	-
不对VPNv4路由进行VPN Target过滤	**undo policy vpn-target**	必选 缺省情况下，PE对收到的VPNv4路由进行VPN Target过滤 通过过滤的路由会被加入到路由表中，没有通过过滤的路由将被丢弃

在跨域 VPN-OptionB 方式中，ASBR-PE 需要保存所有 VPNv4 路由信息，以通告给对端 ASBR-PE。这种情况下，ASBR-PE 应接收所有的 VPNv4 路由信息，不对它们进行 VPN Target 过滤。

说明

在跨域 VPN-OptionB 方式中，对于同一个 VPN，不同 AS 的 PE 之间的 VPN 实例的 VPN Target 需要匹配。

注意

对于跨域 VPN-OptionB，存在两种配置方式:

- 一种是在 ASBR 不改下一跳的方式，在这种方式下 ASBR 间仍然需要配置 MPLS LDP;
- 另一种在 ASBR 上需要修改下一跳，这种组网下 ASBR 间不需要配置 MPLS LDP。

目前实现只支持在 ASBR 上修改下一跳的方式，因此在实现上，默认情况下对于 MP-EBGP 路由，在发往 MP-IBGP 时是修改下一跳的。而对于发往 IBGP 的普通 EBGP 路由，默认情况下是不改下一跳的，如果需要将下一跳改为本地地址，需要配置 **peer** { *ip-address* | *group-name* } **next-hop-local** 命令。具体说明请参见“三层技术-IP 路由配置指导”中的“BGP”。

1.4.4 配置跨域VPN-OptionC

1. 配置PE

同一AS内的PE和ASBR-PE之间建立普通的IBGP对等体关系，不同AS的PE之间建立MP-EBGP对等体关系。

同一 AS 内的 PE 和 ASBR-PE 之间应能够交换标签 IPv4 路由。

表1-20 配置 PE

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
配置本AS的ASBR-PE为IBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
配置与本AS的ASBR-PE之间能够交换带标签的IPv4路由	**peer** { *group-name* | *ip-address* } **label-route-capability**	必选 缺省情况下，不向IPv4对等体/对等体组发送标签路由
将另一AS的PE配置为EBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
进入BGP-VPNv4子地址族视图	**ipv4-family vpnv4**	-
使能EBGP对等体交换BGP-VPNv4路由信息	**peer** { *group-name* | *ip-address* } **enable**	必选
配置向EBGP对等体发送路由时不改变下一跳	**peer** { *group-name* | *ip-address* } **next-hop-invariable**	可选 该步骤用于使用路由反射器通告VPNv4路由的情况：RR之间通告VPNv4路由时，路由的下一跳不能被改变

2. 配置ASBR-PE

在跨域 VPN-OptionC 方式中，需要建立一条跨域的 VPN LSP，相关 PE、ASBR 之间发布公网路由时携带 MPLS 标签信息。

ASBR-PE 与本 AS 的 PE 建立普通 IBGP 对等体关系，与对端 ASBR-PE 建立普通 EBGP 对等体关系。都能够交换标签 IPv4 路由。

携带 MPLS 标签的公网路由通过 MP-BGP 发布。根据 RFC 3107（Carrying Label Information in BGP-4）中的描述，一条路由的标签映射信息可以通过发布这条路由的 BGP Update 消息捎带（piggyback）。这种能力使用 BGP 的扩展属性实现，要求 BGP 对等体能够处理标签 IPv4 路由。

表1-21 配置 ASBR-PE

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
配置本AS的PE为IBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
配置与本AS的PE之间能够交换带标签的IPv4路由	**peer** { *group-name* | *ip-address* } **label-route-capability**	必选 缺省情况下，不向IPv4对等体/对等体组发送标签路由

操作	命令	说明
向本AS的PE发布路由时将下一跳改为自己	**peer** { *group-name* | *ip-address* } **next-hop-local**	必选 缺省情况下，BGP发言者在向IBGP对等体/对等体组发布路由时不会将下一跳改为自己的地址
将对端ASBR-PE配置为EBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
配置与对端ASBR-PE之间能够交换带标签的IPv4路由	**peer** { *group-name* | *ip-address* } **label-route-capability**	必选 缺省情况下，不向IPv4对等体发送标签路由
对对端ASBR-PE发布来的路由应用路由策略	**peer** { *group-name* | *ip-address* } **route-policy** *route-policy-name* **export**	必选 缺省情况下，对等体及对等体组没有应用路由策略

3. 配置路由策略

配置路由策略应用在 ASBR-PE 上：

- 对于从本 AS 的 PE 接收的路由，在向对端 ASBR-PE 发布时，分配 MPLS 标签；
- 对于向本 AS 的 PE 发布的路由，如果是带标签的 IPv4 路由，为其分配新的 MPLS 标签。

对 IPv4 路由分配 MPLS 标签是通过路由策略控制的，只对满足某些条件的路由分配标签，其它路由还是普通 IPv4 路由。

表1-22 配置路由策略

操作	命令	说明
进入系统视图	**system-view**	-
进入路由策略视图	**route-policy** *policy-name* **permit node** *seq-number*	必选
匹配带标签的IPv4路由	**if-match mpls-label**	必选
为IPv4路由分配标签	**apply mpls-label**	必选 缺省情况下，IPv4路由不带标签

说明

有关路由策略的配置请参见“三层技术-IP 路由配置指导”中的“路由策略”。

1.5 配置嵌套VPN

如果网络中 VPN 接入数量比较多，管理者想要管理权限层次化（通过 VPN 划分访问权限），且不想让外部知道用户内部 VPN 的部署情况，可以使用嵌套 VPN 的组网方式。在嵌套 VPN 组网方式中，用户通过较低的管理复杂度和组网成本，就可以实现内部 VPN 网络的层次化管理。

1.5.1 配置准备

配置嵌套VPN之前，需完成MPLS L3VPN的基本配置，请参见“1.3 配置基本MPLS L3VPN”。

1.5.2 配置嵌套VPN

表1-23 配置嵌套 VPN

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	-
配置CE对等体或对等体组	**peer** { *group-name* | *peer-address* } **as-number** *number*	必选
退回BGP视图	**quit**	-
进入BGP-VPNv4子地址族视图	**ipv4-family vpnv4**	-
使能嵌套VPN功能	**nesting-vpn**	必选 缺省情况下，禁止嵌套VPN功能
激活嵌套VPN对等体或对等体组，并使能其交换BGP-VPNv4路由信息的能力	**peer** { *group-name* | *peer-address* } **vpn-instance** *vpn-instance-name* **enable**	必选 缺省情况下，嵌套VPN对等体/对等体组之间只能交换IPv4路由信息，不能交换BGP-VPNv4路由信息
将对等体加入到嵌套VPN对等体组中	**peer** *peer-address* **vpn-instance** *vpn-instance-name* **group** *group-name*	可选 缺省情况下，对等体没有加入任何嵌套VPN对等体组
配置从嵌套VPN对等体或对等体组接收的VPNv4路由应用的路由策略	**peer** { *group-name* | *peer-address* } **vpn-instance** *vpn-instance-name* **route-policy** *route-policy-name* **import**	可选 缺省情况下，对于从嵌套VPN对等体或对等体组接收的VPNv4路由不应用路由策略

说明

- 同一用户 VPN 的不同子 VPN 之间地址空间不能重叠。
- 建议嵌套 VPN 对等体的地址不要与公网中对等体的地址重叠。
- 配置嵌套 VPN 对等体或对等体组之前，需要先配置 BGP-VPN 实例视图下的 CE 对等体或对等体组。
- 目前嵌套 VPN 不支持多跳 EBGP 组网方式，因此运营商 PE 和 CE 之间必须使用直连接口地址建立邻居关系。
- 在某些产品上，如果子 VPN 的 CE 设备直接连接运营商 PE 设备，则需要在该 PE 设备上配置策略路由，使得主 VPN 和子 VPN 可以互访。

1.6 配置HoVPN

对于层次化比较明显的 VPN 网络，可以采用 HoVPN 方案，以降低对 PE 设备的性能要求。

1.6.1 配置准备

在配置 HoVPN 之前，需先在 UPE 和 SPE 上配置基本 MPLS L3VPN。

1.6.2 配置HoVPN

表1-24 配置 HoVPN

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入BGP-VPNv4子地址族视图	**ipv4-family vpnv4**	必选
使能对等体交换BGP-VPNv4路由信息	**peer** { *group-name* | *ip-address* } **enable**	必选
配置BGP对等体或对等体组为UPE	**peer** { *group-name* | *ip-address* } **upe**	必选
向UPE发送指定VPN实例的缺省路由	**peer** { *group-name* | *ip-address* } **default-route-advertise vpn-instance** *vpn-instance-name*	二者必选其一 缺省情况下，不向对等体发布路由
向UPE发送通过路由策略的路由	**peer** { *group-name* | *ip-address* } **upe route-policy** *route-policy-name* **export**	

执行 **peer default-route-advertise vpn-instance** 命令后，不论本地路由表中是否存在缺省路由，SPE 都会向 UPE 发布一条下一跳地址为本地地址的缺省路由。

说明

- 向 BGP 对等体或对等体组发布 VPN 实例缺省路由的前提是：此 BGP 对等体或对等体组必须是 UPE。
- 建议不要同时配置 **peer default-route-advertise vpn-instance** 命令和 **peer upe route-policy** 命令。
- 建议 SPE 不直接同 CE 连接，如必须连接，则 SPE 上 VPN 实例和 UPE 上 VPN 实例必须配置不同的 RD。

1.7 配置OSPF伪连接

在 VPN PE 间配置伪连接 Sham-link 后，Sham-link 将被视为 OSPF 区域内路由。这一特性用于使经过 MPLS VPN 骨干网的路由成为 OSPF 区域内路由，避免 VPN 流量经 backdoor 路由转发。

Sham-link 的源地址和目的地址应使用 32 位掩码的 Loopback 接口地址，且该 loopback 接口需要绑定到 VPN 实例中，并通过 BGP 发布。

1.7.1 配置准备

在配置 OSPF 伪连接之前，需完成以下任务：

- 配置基本 MPLS L3VPN（PE-CE 间使用 OSPF）
- 配置用户 CE 所在局域网的 OSPF

1.7.2 配置Loopback接口

表1-25 配置 Loopback 接口

操作	命令	说明
进入系统视图	**system-view**	-
创建 Loopback 接口并进入 Loopback接口视图	**interface loopback** *interface-number*	必选
将Loopback接口绑定到VPN实例	**ip binding vpn-instance** *vpn-instance-name*	必选 缺省情况下，接口不关联到任何VPN实例，属于公网接口
配置Loopback接口的地址	**ip address** *ip-address* { *mask* | *mask-length* }	必选

1.7.3 配置BGP引入Loopback接口地址和OSPF的路由

表1-26 发布 Loopback 接口的路由

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	必选
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	必选
引入直连路由（将Loopback主机路由引入BGP）	**import-route direct** [**med** *med-value* | **route-policy** *route-policy-name*] *	必选
引入OSPF的私网路由	**import-route ospf** [{ *process-id* | **all-processes** } [**allow-direct** | **med** *med-value* | **route-policy** *route-policy-name*] *]	必选

1.7.4 配置伪连接

表1-27 创建伪连接

操作	命令	说明
进入系统视图	**system-view**	-
进入OSPF视图	**ospf** [*process-id* | **router-id** *router-id* | **vpn-instance** *vpn-instance-name*] *	-
配置VPN引入路由的外部路由标记（External Route Tag）值	**route-tag** *tag-value*	必选
进入OSPF区域视图	**area** *area-id*	必选
配置伪连接	**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*] *	必选 缺省情况下，未配置伪连接

说明

- 启动 OSPF 而不手工配置 Router ID 时，系统会自动选取出一个 Router ID。选举规则请参见“三层技术-IP 路由配置指导”中的“OSPF”。需要注意的是，由于选举规则相同会得到相同的 Router ID。建议用户启动 OSPF 进程时手工配置 Router ID。
- 配置 OSPF 的 VPN 多实例而不配置 route tag 时，系统会根据配置的 BGP 自治系统号生成一个 tag 值，如果没有配置 BGP 则 tag 为 0。需要注意的是，在 BGP 的自治系统号相同的 PE 或同

一台 PE 上，因为计算规则相同会得到相同的 tag 值。建议用户配置 OSPF 的 VPN 多实例时手工配置 tag 值。

1.8 配置BGP的AS号替换和SoO属性

1.8.1 配置准备

在配置 BGP 的 AS 号替换和 SoO 属性之前，需完成以下任务：

- 配置基本 MPLS L3VPN
- 不同 Site 的 CE 具有相同的 AS 号

1.8.2 配置BGP的AS号替换和SoO属性

不同 Site 的 CE 具有相同的 AS 号时，PE 上需要开启 BGP 的 AS 号替换功能，从而避免路由被丢弃。

使能了 BGP 的 AS 号替换功能后，当 PE 向指定对等体中的 CE 发布路由时，如果路由的 AS_PATH 中有与 CE 相同的 AS 号，将被替换成 PE 的 AS 号后再发布。

同时，为了避免 PE 使用不同接口连接同一站点的多个 CE 时产生路由环路，需要在 PE 上对从 CE 学习到的路由应用入方向路由策略，指定为路由添加的 SoO 属性。

表1-28 配置 BGP 的 AS 号替换和 SoO 属性

操作	命令	说明
进入系统视图	**system-view**	-
创建路由策略，并进入路由策略视图	**route-policy** *route-policy-name* **permit node** *node-number*	可选 缺省情况下，不存在任何路由策略
配置路由的SoO属性	**apply extcommunity soo** *site-of-origin* **additive**	可选 缺省情况下，没有指定SoO属性
退回系统视图	**quit**	-
进入BGP视图	**bgp** *as-number*	必选
进入BGP-VPN实例视图	**ipv4-family vpn-instance** *vpn-instance-name*	必选
使能BGP的AS号替换功能	**peer** { *ip-address* | *group-name* } **substitute-as**	必选 缺省情况下，BGP的AS号替换功能是被禁止的
对从对等体学习的路由应用已配置的路由策略，为路由添加SoO属性	**peer** { *ip-address* | *group-name* } **route-policy** *route-policy-name* **import**	可选 缺省情况下，没有对从对等体学习的路由应用路由策略

说明

route-policy 和 **apply extcommunity** 命令的介绍，请参见“三层技术-IP 路由命令参考”中的“路由策略”；**peer substitute-as** 和 **peer route-policy** 命令的介绍，请参见“三层技术-IP 路由命令参考”中的“BGP”。

1.9 MPLS L3VPN显示和维护

1.9.1 复位BGP连接

当 BGP 配置变化后，可以通过软复位或复位 BGP 连接使新的配置生效。软复位需要 BGP 对等体具备路由刷新能力（支持 ROUTE-REFRESH 消息）。

说明

软复位 BGP 连接是指在不断开 BGP 邻居关系的情况下，更新 BGP 路由信息；复位 BGP 连接是指断开并重新建立 BGP 邻居关系的情况下，更新 BGP 路由信息。

请在用户视图下进行下列操作。

表1-29 复位 BGP 连接

操作	命令
软复位指定VPN实例的BGP连接	**refresh bgp vpn-instance** *vpn-instance-name* { *ip-address* | **all** | **external** | **group** *group-name* } { **export** | **import** }
软复位BGP的VPNv4连接	**refresh bgp vpnv4** { *ip-address* | **all** | **external** | **group** *group-name* | **internal** } { **export** | **import** }
复位指定VPN实例的BGP连接	**reset bgp vpn-instance** *vpn-instance-name* { *as-number* | *ip-address* | **all** | **external** | **group** *group-name* }
复位BGP的VPNv4连接	**reset bgp vpnv4** { *as-number* | *ip-address* | **all** | **external** | **internal** | **group** *group-name* }

1.9.2 显示MPLS L3VPN的运行状态

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 MPLS L3VPN 的运行情况，通过查看显示信息验证配置的效果。

在用户视图下执行 **reset** 命令可以清除路由振荡衰减信息。

表1-30 显示 MPLS L3VPN 的运行状态

操作	命令
显示与VPN实例相关联的IP路由表	**display ip routing-table vpn-instance** *vpn-instance-name* [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]

操作	命令
显示指定VPN实例信息	**display ip vpn-instance** [**instance-name** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例的FIB信息	**display fib vpn-instance** *vpn-instance-name* [**acl** *acl-number* | **ip-prefix** *ip-prefix-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例中与指定目的IP地址匹配的FIB信息	**display fib vpn-instance** *vpn-instance-name ip-address* [*mask* | *mask-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示BGP路由表中的标签路由信息	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **routing-table label** [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示BGP VPNv4对等体组信息	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **group** [*group-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示引入的BGP VPNv4路由信息	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **network** [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示BGP VPNv4的AS路径信息	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **paths** [*as-regular-expression* | { **|** { **begin** | **exclude** | **include** } *regular-expression* }]
显示BGP VPNv4对等体信息	**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*]
显示从指定邻居收到的ORF报文中的前缀信息	**display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **peer** *ip-address* **received ip-prefix** [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示所有BGP VPNv4路由信息	**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* | *comm-list-name* } [**whole-match**] | *adv-community-list-number* } | **different-origin-as** | **peer** *ip-address* { **advertised-routes** | **received-routes** } [**statistic**] | **statistic**] [**|** { **begin** | **exclude** | **include** } *regular-expression*] | **regular-expression** *as-regular-expression*]
显示指定路由标识符的BGP VPNv4路由信息	**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* | *comm-list-name* } [**whole-match**] | *adv-community-list-number* } | **different-origin-as**] [**|** { **begin** | **exclude** | **include** } *regular-expression*] | **regular-expression** *as-regular-expression*]

操作	命令
显示指定VPN实例的BGP VPNv4路由信息	**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* | *comm-list-name* } [**whole-match**] | *adv-community-list-number* } | **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*]
显示OSPF伪连接信息	**display ospf** [*process-id*] **sham-link** [**area** *area-id*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示所有或指定隧道策略的信息	**display tunnel-policy** { **all** | **policy-name** *tunnel-policy-name* } [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定LDP实例的信息	**display mpls ldp vpn-instance** *vpn-instance-name* [**|** { **begin** | **exclude** | **include** } *regular-expression*]
清除指定VPN实例的路由振荡衰减信息	**reset bgp vpn-instance** *vpn-instance-name* **dampening** [*network-address* [*mask* | *mask-length*]
清除指定VPN实例的BGP对等体路由振荡历史信息	**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*]

说明

有关查看路由表的命令介绍请参见“三层技术-IP 路由命令参考”中的“IP 路由基础命令”。

1.10 MPLS L3VPN典型配置举例

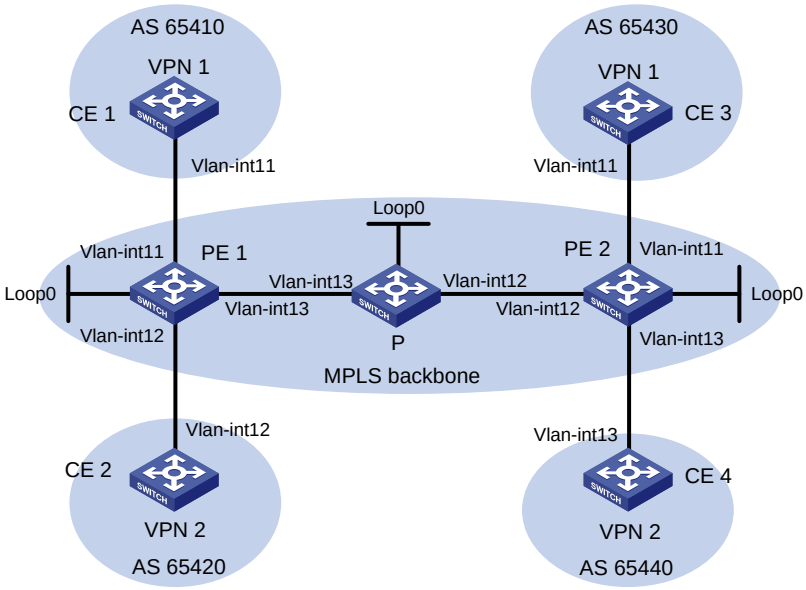
1.10.1 PE-CE间使用EBGP基本MPLS L3VPN配置示例

1. 组网需求

- CE 1、CE 3 属于 VPN 1，CE 2、CE 4 属于 VPN 2；
- VPN 1 使用的 VPN Target 属性为 111:1，VPN 2 使用的 VPN Target 属性为 222:2。不同 VPN 用户之间不能互相访问；
- CE 与 PE 之间配置 EBGP 交换 VPN 路由信息；
- PE 与 PE 之间配置 OSPF 实现 PE 内部的互通、配置 MP-IBGP 交换 VPN 路由信息。

2. 组网图

图1-19 PE-CE 间使用 EBGP 基本 MPLS L3VPN 组网图



设备	接口	IP地址	设备	接口	IP地址
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			

3. 配置步骤

(1) 在 MPLS 骨干网上配置 IGP 协议，实现骨干网 PE 和 P 的互通

配置 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
```

配置 P。

```
<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
```

配置 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
```

配置完成后，PE 1、P、PE 2 之间应能建立 OSPF 邻居，执行 **display ospf peer** 命令可以看到邻居达到 FULL 状态。执行 **display ip routing-table** 命令可以看到 PE 之间学习到对方的 Loopback 路由。

以 PE 1 为例：

```
[PE1] 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   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
172.2.1.0/24      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: 172.1.1.1  BDR: 172.1.1.2  MTU: 0
   Dead timer due in 38  sec
   Neighbor is up for 00:02:44
   Authentication Sequence: [ 0 ]
   Neighbor state change count: 5
```

(2) 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 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
```

配置 P。

```
[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
```

配置 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
```

上述配置完成后，PE 1、P、PE 2 之间应能建立 LDP 会话，执行 **display mpls ldp session** 命令可以看到显示结果中 Status 字段为 Operational。执行 **display mpls ldp lsp** 命令，可以看到 LDP LSP 的建立情况。

以 PE 1 为例：

```
[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     -------/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) 在 PE 设备上配置 VPN 实例，将 CE 接入 PE

配置 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
```

配置 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
```

按 图 1-19 配置各CE的接口IP地址，配置过程略。

配置完成后，在 PE 设备上执行 **display ip vpn-instance** 命令可以看到 VPN 实例的配置情况。各 PE 能 ping 通自己接入的 CE。

以 PE 1 和 CE 1 为例：

```
[PE1] display ip vpn-instance
  Total VPN-Instances configured : 2

  VPN-Instance Name               RD                      Create time
  vpn1                            100:1                   2009/01/22 13:02:21
  vpn2                            100:2                   2009/01/22 13:02:40
[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) 在 PE 与 CE 之间建立 EBGP 对等体，引入 VPN 路由

配置 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
```

说明

另外 3 个 CE 设备（CE 2～CE 4）配置与 CE 1 设备配置类似，配置过程省略。

配置 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
```

说明

PE 2 的配置与 PE 1 类似，配置过程省略。

配置完成后，在 PE 设备上执行 **display bgp vpnv4 vpn-instance peer** 命令，可以看到 PE 与 CE 之间的 BGP 对等体关系已建立，并达到 Established 状态。

以 PE 1 与 CE 1 的对等体关系为例：

```
[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) 在 PE 之间建立 MP-IBGP 对等体

配置 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
```

配置 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
```

配置完成后，在 PE 设备上执行 **display bgp peer** 或 **display bgp vpnv4 all peer** 命令，可以看到 PE 之间的 BGP 对等体关系已建立，并达到 Established 状态。

```
[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) 配置完成后的检验

在 PE 设备上执行 **display ip routing-table vpn-instance** 命令，可以看到去往对端 CE 的路由。

以 PE 1 为例：

```
[PE1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 5        Routes : 5
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
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
[PE1] display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 5        Routes : 5
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
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
```

同一 VPN 的 CE 能够相互 Ping 通，不同 VPN 的 CE 不能相互 Ping 通。

例如：CE 1 能够 Ping 通 CE 3（10.3.1.1），但不能 Ping 通 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
```

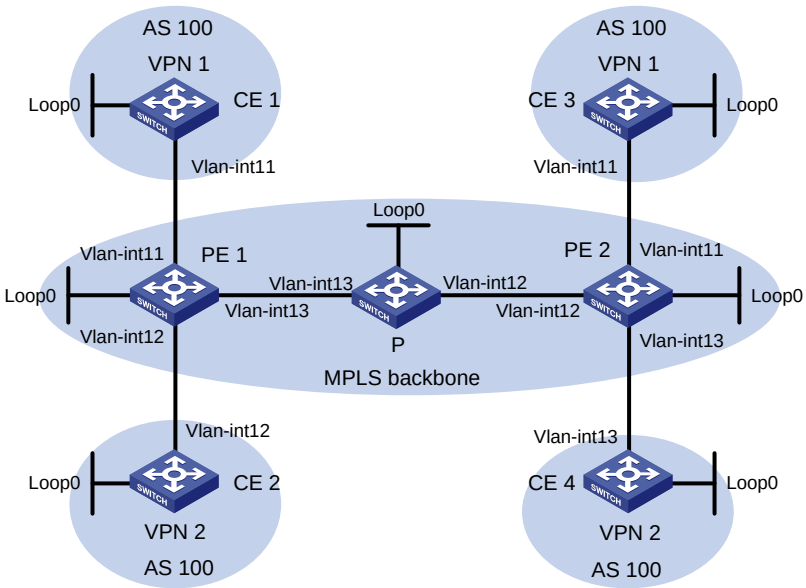
1.10.2 PE-CE间使用IBGP基本MPLS L3VPN配置示例

1. 组网需求

- CE 1、CE 3 属于 VPN 1，CE 2、CE 4 属于 VPN 2；
- VPN 1 使用的 VPN Target 属性为 111:1，VPN 2 使用的 VPN Target 属性为 222:2。不同 VPN 用户之间不能互相访问；
- CE 与 PE 之间配置 IBGP 交换 VPN 路由信息；
- PE 与 PE 之间配置 OSPF 实现 PE 内部的互通、配置 MP-IBGP 交换 VPN 路由信息。

2. 组网图

图1-20 PE-CE 间使用 IBGP 基本 MPLS L3VPN 组网图



设备	接口	IP地址	设备	接口	IP地址
PE 1	Loop0	1.1.1.9/32	PE 2	Loop0	3.3.3.9/32
	Vlan-int11	10.1.1.2/24		Vlan-int12	172.2.1.2/24
	Vlan-int13	172.1.1.1/24		Vlan-int11	10.3.1.2/24
	Vlan-int12	10.2.1.2/24		Vlan-int13	10.4.1.2/24
CE 1	Loop0	4.4.4.9/32	P	Loop0	2.2.2.9/32
	Vlan-int11	10.1.1.1/24		Vlan-int12	172.2.1.1/24
CE 2	Loop0	5.5.5.9/32		Vlan-int13	172.1.1.2/24
	Vlan-int12	10.2.1.1/24	CE 4	Loop0	7.7.7.9/32
CE 3	Loop0	6.6.6.9/32		Vlan-int13	10.4.1.1/24
	Vlan-int11	10.3.1.1/24			

3. 配置步骤

(1) 在 MPLS 骨干网上配置 IGP 协议，实现骨干网 PE 和 P 的互通

配置 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
```

配置 P。

```
<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
```

配置 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
```

配置完成后，PE 1、P、PE 2 之间应能建立 OSPF 邻居，执行 **display ospf peer** 命令可以看到邻居达到 FULL 状态。执行 **display ip routing-table** 命令可以看到 PE 之间学习到对方的 Loopback 路由。

以 PE 1 为例：

```
[PE1] 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   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
172.2.1.0/24      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: 172.1.1.1  BDR: 172.1.1.2  MTU: 0
   Dead timer due in 38  sec
   Neighbor is up for 00:02:44
   Authentication Sequence: [ 0 ]
   Neighbor state change count: 5
```

(2) 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 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
```

配置 P。

```
[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
```

配置 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
```

上述配置完成后，PE 1、P、PE 2 之间应能建立 LDP 会话，执行 **display mpls ldp session** 命令可以看到显示结果中 Status 字段为 Operational。执行 **display mpls ldp lsp** 命令，可以看到 LDP LSP 的建立情况。

以 PE 1 为例：

```
[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     -------/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) 在 PE 设备上配置 VPN 实例，将 CE 接入 PE

配置 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
```

配置 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
```

按 图 1-20 配置各CE的接口IP地址，配置过程略。

配置完成后，在 PE 设备上执行 **display ip vpn-instance** 命令可以看到 VPN 实例的配置情况。各 PE 能 ping 通自己接入的 CE。

以 PE 1 和 CE 1 为例：

```
[PE1] display ip vpn-instance
  Total VPN-Instances configured : 2

  VPN-Instance Name               RD                      Create time
  vpn1                            100:1                   2009/01/22 13:02:21
  vpn2                            100:2                   2009/01/22 13:02:40
[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) 在 PE 与 CE 之间建立 IBGP 对等体，引入 VPN 路由，并配置路由策略修改路由的下一跳

在 CE 1 上配置 IBGP 对等体，并对从 PE 1 接收的路由应用路由策略，将路由下一跳修改为 PE 1 的地址。

```
<CE1> system-view
[CE1] route-policy ce-ibgp permit node 0
```

```
[CE1-route-policy] apply ip-address next-hop 10.1.1.2
[CE1-route-policy] quit
[CE1] bgp 100
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] peer 10.1.1.2 route-policy ce-ibgp import
[CE1-bgp] import-route direct
[CE1-bgp] quit
```

说明

另外 3 个 CE 设备（CE 2～CE 4）配置与 CE 1 设备配置类似，配置过程省略。

在 PE 1 上配置 IBGP 对等体，并配置 PE 1 作为路由反射器反射路由。

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] peer 10.1.1.1 as-number 100
[PE1-bgp-vpn1] peer 10.1.1.1 reflect-client
[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 100
[PE1-bgp-vpn2] peer 10.2.1.1 reflect-client
[PE1-bgp-vpn2] import-route direct
[PE1-bgp-vpn2] quit
[PE1-bgp] quit
```

说明

PE 2 的配置与 PE 1 类似，配置过程省略。

配置完成后，在 PE 设备上执行 **display bgp vpnv4 vpn-instance peer** 命令，可以看到 PE 与 CE 之间的 BGP 对等体关系已建立，并达到 Established 状态。

以 PE 1 与 CE 1 的对等体关系为例：

```
[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               100       26       21    0       2 00:11:08 Established
```

(5) 在 PE 之间建立 MP-IBGP 对等体

在 PE 1 上配置 MP-IBGP 对等体，并对从 PE 2 接收的路由应用路由策略，将路由下一跳修改为 PE 2 的环回接口地址。

```
[PE1] route-policy pe-ibgp permit node 0
```

```
[PE1-route-policy] apply ip-address next-hop 3.3.3.9
[PE1-route-policy] quit
[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] peer 3.3.3.9 route-policy pe-ibgp import
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

在 PE 2 上配置 MP-IBGP 对等体，并对从 PE 1 接收的路由应用路由策略，将路由下一跳修改为 PE 1 的环回接口地址。

```
[PE2] route-policy pe-ibgp permit node 0
[PE2-route-policy] apply ip-address next-hop 1.1.1.9
[PE2-route-policy] quit
[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] peer 1.1.1.9 route-policy pe-ibgp import
[PE2-bgp-af-vpnv4] quit
[PE2-bgp] quit
```

配置完成后，在 PE 设备上执行 **display bgp peer** 或 **display bgp vpnv4 all peer** 命令，可以看到 PE 之间的 BGP 对等体关系已建立，并达到 Established 状态。

```
[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        4        8    0       0 00:00:09 Established
```

(6) 配置完成后的检验

在 PE 设备上执行 **display ip routing-table vpn-instance** 命令，可以看到去往对端 CE 的路由。以 PE 1 为例：

```
[PE1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 7        Routes : 7
Destination/Mask    Proto  Pre  Cost         NextHop         Interface
4.4.4.9/32          BGP    255  0            10.1.1.1        Vlan11
6.6.6.9/32          BGP    255  0            3.3.3.9         NULL0
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
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
[PE1] display ip routing-table vpn-instance vpn2
Routing Tables: vpn2
         Destinations : 5        Routes : 5
Destination/Mask    Proto  Pre  Cost         NextHop         Interface
5.5.5.9/32          BGP    255  0            10.2.1.1        Vlan12
7.7.7.9/32          BGP    255  0            3.3.3.9         NULL0
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
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
```

同一 VPN 的 CE 能够相互 Ping 通，不同 VPN 的 CE 不能相互 Ping 通。

例如：CE 1 能够 Ping 通 CE 3（6.6.6.9），但不能 Ping 通 CE 4（7.7.7.9）。

```
[CE1] ping 6.6.6.9
  PING 6.6.6.9: 56  data bytes, press CTRL_C to break
    Reply from 6.6.6.9: bytes=56 Sequence=1 ttl=253 time=72 ms
    Reply from 6.6.6.9: bytes=56 Sequence=2 ttl=253 time=34 ms
    Reply from 6.6.6.9: bytes=56 Sequence=3 ttl=253 time=50 ms
    Reply from 6.6.6.9: bytes=56 Sequence=4 ttl=253 time=50 ms
    Reply from 6.6.6.9: bytes=56 Sequence=5 ttl=253 time=34 ms
  --- 6.6.6.9 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 7.7.7.9
  PING 7.7.7.9: 56  data bytes, press CTRL_C to break
    Request time out
    Request time out
    Request time out
    Request time out
    Request time out
  --- 7.7.7.9 ping statistics ---
    5 packet(s) transmitted
    0 packet(s) received
    100.00% packet loss
```

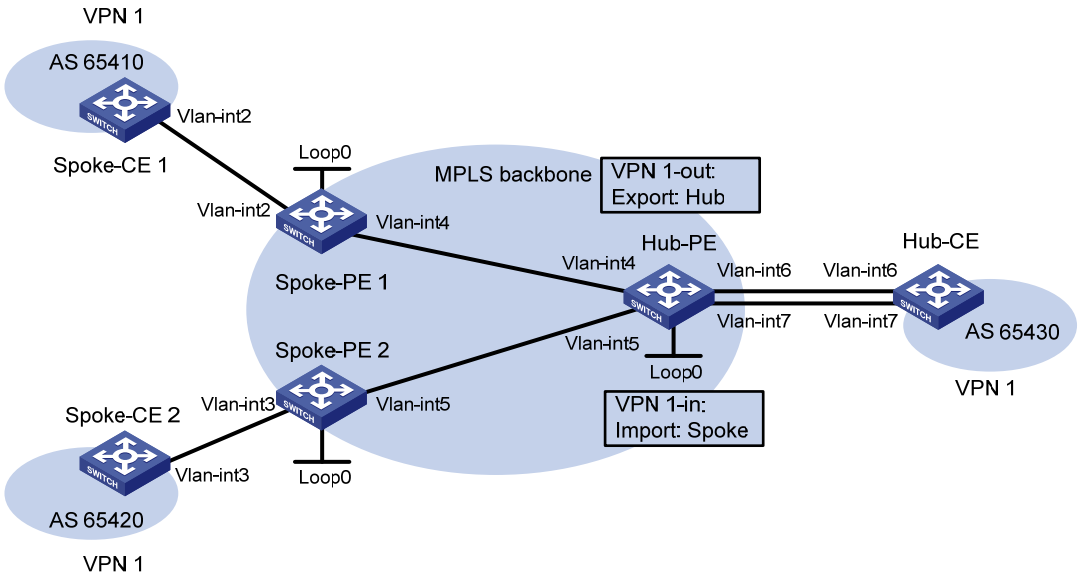
1.10.3 配置Hub&Spoke组网示例

1. 组网需求

- Spoke-CE 之间不能直接通信，只能通过 Hub-CE 转发 Spoke-CE 之间的流量。
- Spoke-CE 与 Spoke-PE 之间、Hub-CE 与 Hub-PE 之间配置 EBGP 交换 VPN 路由信息。
- Spoke-PE 与 Hub-PE 之间配置 OSPF 实现 PE 内部的互通、配置 MP-IBGP 交换 VPN 路由信息。

2. 组网图

图1-21 Hub&Spoke 组网图



设备	接口	IP地址	设备	接口	IP地址
Spoke-CE 1	Vlan-int2	10.1.1.1/24	Hub-CE	Vlan-int6	10.3.1.1/24
Spoke-PE 1	Loop0	1.1.1.9/32		Vlan-int7	10.4.1.1/24
	Vlan-int2	10.1.1.2/24	Hub-PE	Loop0	2.2.2.9/32
	Vlan-int4	172.1.1.1/24		Vlan-int4	172.1.1.2/24
Spoke-CE 2	Vlan-int3	10.2.1.1/24		Vlan-int5	172.2.1.2/24
Spoke-PE 2	Loop0	3.3.3.9/32		Vlan-int6	10.3.1.2/24
	Vlan-int3	10.2.1.2/24		Vlan-int7	10.4.1.2/24
	Vlan-int5	172.2.1.1/24			

3. 配置步骤

(1) 在 MPLS 骨干网上配置 IGP 协议，实现骨干网 Spoke-PE、Hub-PE 之间的互通

配置 Spoke-PE 1。

```
<Spoke-PE1> system-view
[Spoke-PE1] interface loopback 0
[Spoke-PE1-LoopBack0] ip address 1.1.1.9 32
[Spoke-PE1-LoopBack0] quit
[Spoke-PE1] interface vlan-interface 4
[Spoke-PE1-Vlan-interface4] ip address 172.1.1.1 24
[Spoke-PE1-Vlan-interface4] quit
[Spoke-PE1] ospf
[Spoke-PE1-ospf-1] area 0
[Spoke-PE1-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[Spoke-PE1-ospf-1-area-0.0.0.0] network 1.1.1.9 0.0.0.0
[Spoke-PE1-ospf-1-area-0.0.0.0] quit
[Spoke-PE1-ospf-1] quit
```

配置 Spoke-PE 2。

```
<Spoke-PE2> system-view
[Spoke-PE2] interface loopback 0
[Spoke-PE2-LoopBack0] ip address 3.3.3.9 32
```

```
[Spoke-PE2-LoopBack0] quit
[Spoke-PE2] interface vlan-interface 5
[Spoke-PE2-Vlan-interface5] ip address 172.2.1.1 24
[Spoke-PE2-Vlan-interface5] quit
[Spoke-PE2] ospf
[Spoke-PE2-ospf-1] area 0
[Spoke-PE2-ospf-1-area-0.0.0.0] network 172.2.1.0 0.0.0.255
[Spoke-PE2-ospf-1-area-0.0.0.0] network 3.3.3.9 0.0.0.0
[Spoke-PE2-ospf-1-area-0.0.0.0] quit
[Spoke-PE2-ospf-1] quit
```

配置 Hub-PE。

```
<Hub-PE> system-view
[Hub-PE] interface loopback 0
[Hub-PE-LoopBack0] ip address 2.2.2.9 32
[Hub-PE-LoopBack0] quit
[Hub-PE] interface vlan-interface 4
[Hub-PE-Vlan-interface4] ip address 172.1.1.2 24
[Hub-PE-Vlan-interface4] quit
[Hub-PE] interface vlan-interface 5
[Hub-PE-Vlan-interface5] ip address 172.2.1.2 24
[Hub-PE-Vlan-interface5] quit
[Hub-PE] ospf
[Hub-PE-ospf-1] area 0
[Hub-PE-ospf-1-area-0.0.0.0] network 172.1.1.0 0.0.0.255
[Hub-PE-ospf-1-area-0.0.0.0] network 172.2.1.0 0.0.0.255
[Hub-PE-ospf-1-area-0.0.0.0] network 2.2.2.9 0.0.0.0
[Hub-PE-ospf-1-area-0.0.0.0] quit
[Hub-PE-ospf-1] quit
```

配置完成后，Spoke-PE 1、Spoke-PE 2、Hub-PE 之间应能建立 OSPF 邻居，执行 **display ospf peer** 命令可以看到邻居达到 FULL 状态。执行 **display ip routing-table** 命令可以看到 PE 之间学习到对方的 Loopback 路由。

以 Spoke-PE 1 为例：

```
[Spoke-PE1] 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          OSPF   10   1            172.1.1.2       Vlan4
3.3.3.9/32          OSPF   10   2            172.1.1.2       Vlan4
10.1.1.0/24         Direct 0    0            10.1.1.2        Vlan2
10.1.1.2/32         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
172.1.1.0/24        Direct 0    0            172.1.1.1       Vlan4
172.1.1.1/32        Direct 0    0            127.0.0.1       InLoop0
```

```
172.2.1.0/24        OSPF   10   1            172.1.1.2       Vlan4
[Spoke-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-interface4)'s neighbors
 Router ID: 2.2.2.9          Address: 172.1.1.2        GR State: Normal
   State: Full  Mode:Nbr is  Master  Priority: 1
   DR: 172.1.1.1  BDR: 172.1.1.2  MTU: 0
   Dead timer due in 38  sec
   Neighbor is up for 00:02:44
   Authentication Sequence: [ 0 ]
   Neighbor state change count: 5
```

(2) 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 Spoke-PE 1。

```
[Spoke-PE1] mpls lsr-id 1.1.1.9
[Spoke-PE1] mpls
[Spoke-PE1-mpls] quit
[Spoke-PE1] mpls ldp
[Spoke-PE1-mpls-ldp] quit
[Spoke-PE1] interface vlan-interface 4
[Spoke-PE1-Vlan-interface4] mpls
[Spoke-PE1-Vlan-interface4] mpls ldp
[Spoke-PE1-Vlan-interface4] quit
```

配置 Spoke-PE 2。

```
[Spoke-PE2] mpls lsr-id 3.3.3.9
[Spoke-PE2] mpls
[Spoke-PE2-mpls] quit
[Spoke-PE2] mpls ldp
[Spoke-PE2-mpls-ldp] quit
[Spoke-PE2] interface vlan-interface 5
[Spoke-PE2-Vlan-interface5] mpls
[Spoke-PE2-Vlan-interface5] mpls ldp
[Spoke-PE2-Vlan-interface5] quit
```

配置 Hub-PE。

```
[Hub-PE] mpls lsr-id 2.2.2.9
[Hub-PE] mpls
[Hub-PE-mpls] quit
[Hub-PE] mpls ldp
[Hub-PE-mpls-ldp] quit
[Hub-PE] interface vlan-interface 4
[Hub-PE-Vlan-interface4] mpls
[Hub-PE-Vlan-interface4] mpls ldp
[Hub-PE-Vlan-interface4] quit
[Hub-PE] interface vlan-interface 5
[Hub-PE-Vlan-interface5] mpls
[Hub-PE-Vlan-interface5] mpls ldp
[Hub-PE-Vlan-interface5] quit
```

上述配置完成后，Spoke-PE 1、Spoke-PE 2、Hub-PE 之间应能建立 LDP 会话，执行 **display mpls ldp session** 命令可以看到显示结果中 Status 字段为 Operational。执行 **display mpls ldp lsp** 命令，可以看到 LDP LSP 的建立情况。

以 Spoke-PE 1 为例：

```
[Spoke-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
[Spoke-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     -------/InLoop0
 2   2.2.2.9/32         NULL/3       172.1.1.2     -------/Vlan-interface4
 3   3.3.3.9/32         NULL/1024    172.1.1.2     -------/Vlan-interface4
------------------------------------------------------------------------
 A '*' before an LSP means the LSP is not established
 A '*' before a Label means the USCB or DSCB is stale
```

(3) 在 Spoke-PE 和 Hub-PE 设备上配置 VPN 实例，将 CE 接入 PE

配置 Spoke-PE 1。

```
[Spoke-PE1] ip vpn-instance vpn1
[Spoke-PE1-vpn-instance-vpn1] route-distinguisher 100:1
[Spoke-PE1-vpn-instance-vpn1] vpn-target 111:1 import-extcommunity
[Spoke-PE1-vpn-instance-vpn1] vpn-target 222:2 export-extcommunity
[Spoke-PE1-vpn-instance-vpn1] quit
[Spoke-PE1] interface vlan-interface 2
[Spoke-PE1-Vlan-interface2] ip binding vpn-instance vpn1
[Spoke-PE1-Vlan-interface2] ip address 10.1.1.2 24
[Spoke-PE1-Vlan-interface2] quit
```

配置 Spoke-PE 2。

```
[Spoke-PE2] ip vpn-instance vpn1
[Spoke-PE2-vpn-instance-vpn1] route-distinguisher 100:2
[Spoke-PE2-vpn-instance-vpn1] vpn-target 111:1 import-extcommunity
[Spoke-PE2-vpn-instance-vpn1] vpn-target 222:2 export-extcommunity
[Spoke-PE2-vpn-instance-vpn1] quit
[Spoke-PE2] interface vlan-interface 3
[Spoke-PE2-Vlan-interface3] ip binding vpn-instance vpn1
[Spoke-PE2-Vlan-interface3] ip address 10.2.1.2 24
[Spoke-PE2-Vlan-interface3] quit
```

配置 Hub-PE。

```
[Hub-PE] ip vpn-instance vpn1-in
```

```
[Hub-PE-vpn-instance-vpn1-in] route-distinguisher 100:3
[Hub-PE-vpn-instance-vpn1-in] vpn-target 222:2 import-extcommunity
[Hub-PE-vpn-instance-vpn1-in] quit
[Hub-PE] ip vpn-instance vpn1-out
[Hub-PE-vpn-instance-vpn1-out] route-distinguisher 100:4
[Hub-PE-vpn-instance-vpn1-out] vpn-target 111:1 export-extcommunity
[Hub-PE-vpn-instance-vpn1-out] quit
[Hub-PE] interface vlan-interface 6
[Hub-PE-Vlan-interface6] ip binding vpn-instance vpn1-in
[Hub-PE-Vlan-interface6] ip address 10.3.1.2 24
[Hub-PE-Vlan-interface6] quit
[Hub-PE] interface vlan-interface 7
[Hub-PE-Vlan-interface7] ip binding vpn-instance vpn1-out
[Hub-PE-Vlan-interface7] ip address 10.4.1.2 24
[Hub-PE-Vlan-interface7] quit
```

按 图 1-21 配置各CE的接口IP地址，配置过程略。

配置完成后，在 PE 设备上执行 **display ip vpn-instance** 命令可以看到 VPN 实例的配置情况。各 PE 能 ping 通自己接入的 CE。

以 Spoke-PE 1 和 Spoke-CE 1 为例：

```
[Spoke-PE1] display ip vpn-instance
  Total VPN-Instances configured : 1

  VPN-Instance Name               RD                      Create time
  vpn1                            100:1                   2009/04/08 10:55:07
[Spoke-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) 在 PE 与 CE 之间建立 EBGP 对等体，引入 VPN 路由

配置 Spoke-CE 1。

```
<Spoke-CE1> system-view
[Spoke-CE1] bgp 65410
[Spoke-CE1-bgp] peer 10.1.1.2 as-number 100
[Spoke-CE1-bgp] import-route direct
[Spoke-CE1-bgp] quit
```

配置 Spoke-CE 2。

```
<Spoke-CE2> system-view
[Spoke-CE2] bgp 65420
[Spoke-CE2-bgp] peer 10.2.1.2 as-number 100
```

```
[Spoke-CE2-bgp] import-route direct
[Spoke-CE2-bgp] quit
```

配置 Hub-CE。

```
<Hub-CE> system-view
[Hub-CE] bgp 65430
[Hub-CE-bgp] peer 10.3.1.2 as-number 100
[Hub-CE-bgp] peer 10.4.1.2 as-number 100
[Hub-CE-bgp] import-route direct
[Hub-CE-bgp] quit
```

配置 Spoke-PE 1。

```
[Spoke-PE1] bgp 100
[Spoke-PE1-bgp] ipv4-family vpn-instance vpn1
[Spoke-PE1-bgp-vpn1] peer 10.1.1.1 as-number 65410
[Spoke-PE1-bgp-vpn1] import-route direct
[Spoke-PE1-bgp-vpn1] quit
[Spoke-PE1-bgp] quit
```

配置 Spoke-PE 2。

```
[Spoke-PE2] bgp 100
[Spoke-PE2-bgp] ipv4-family vpn-instance vpn1
[Spoke-PE2-bgp-vpn1] peer 10.2.1.1 as-number 65420
[Spoke-PE2-bgp-vpn1] import-route direct
[Spoke-PE2-bgp-vpn1] quit
[Spoke-PE2-bgp] quit
```

配置 Hub-PE。

```
[Hub-PE] bgp 100
[Hub-PE-bgp] ipv4-family vpn-instance vpn1-in
[Hub-PE-bgp-vpn1-in] peer 10.3.1.1 as-number 65430
[Hub-PE-bgp-vpn1-in] import-route direct
[Hub-PE-bgp-vpn1-in] quit
[Hub-PE-bgp] ipv4-family vpn-instance vpn1-out
[Hub-PE-bgp-vpn1-out] peer 10.4.1.1 as-number 65430
[Hub-PE-bgp-vpn1-out] peer 10.4.1.1 allow-as-loop
[Hub-PE-bgp-vpn1-out] import-route direct
[Hub-PE-bgp-vpn1-out] quit
[Hub-PE-bgp] quit
```

配置完成后，在 PE 设备上执行 **display bgp vpnv4 vpn-instance peer** 命令，可以看到 PE 与 CE 之间的 BGP 对等体关系已建立，并达到 Established 状态。

以 Spoke-PE 1 与 Spoke-CE 1 的对等体关系为例：

```
[Spoke-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        6        7    0       2 00:03:16 Established
```

(5) 在 Spoke-PE 和 Hub-PE 之间建立 MP-IBGP 对等体

配置 Spoke-PE 1。

```
[Spoke-PE1] bgp 100
[Spoke-PE1-bgp] peer 2.2.2.9 as-number 100
[Spoke-PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[Spoke-PE1-bgp] ipv4-family vpnv4
[Spoke-PE1-bgp-af-vpnv4] peer 2.2.2.9 enable
[Spoke-PE1-bgp-af-vpnv4] quit
[Spoke-PE1-bgp] quit
```

配置 Spoke-PE 2。

```
[Spoke-PE2] bgp 100
[Spoke-PE2-bgp] peer 2.2.2.9 as-number 100
[Spoke-PE2-bgp] peer 2.2.2.9 connect-interface loopback 0
[Spoke-PE2-bgp] ipv4-family vpnv4
[Spoke-PE2-bgp-af-vpnv4] peer 2.2.2.9 enable
[Spoke-PE2-bgp-af-vpnv4] quit
[Spoke-PE2-bgp] quit
```

配置 Hub-PE。

```
[Hub-PE] bgp 100
[Hub-PE-bgp] peer 1.1.1.9 as-number 100
[Hub-PE-bgp] peer 1.1.1.9 connect-interface loopback 0
[Hub-PE-bgp] peer 3.3.3.9 as-number 100
[Hub-PE-bgp] peer 3.3.3.9 connect-interface loopback 0
[Hub-PE-bgp] ipv4-family vpnv4
[Hub-PE-bgp-af-vpnv4] peer 1.1.1.9 enable
[Hub-PE-bgp-af-vpnv4] peer 3.3.3.9 enable
[Hub-PE-bgp-af-vpnv4] quit
[Hub-PE-bgp] quit
```

配置完成后，在 PE 设备上执行 **display bgp peer** 或 **display bgp vpnv4 all peer** 命令，可以看到 PE 之间的 BGP 对等体关系已建立，并达到 Established 状态。

```
[Spoke-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

  2.2.2.9                100        6        5    0       0 00:00:32 Established
```

(6) 配置完成后的检验

在 PE 设备上执行 **display ip routing-table vpn-instance** 命令，可以看到去往各个 CE 的路由，且 Spoke-PE 上到达对端 Spoke-CE 的路由指向 Hub-PE。以 Spoke-PE 1 为例：

```
[Spoke-PE1] display ip routing-table vpn-instance vpn1
Routing Tables: vpn1
         Destinations : 8       Routes : 8
```

```
Destination/Mask    Proto   Pre  Cost        NextHop         Interface

10.0.0.0/24         BGP     255  0           2.2.2.9         NULL0
10.1.1.0/24         Direct  0    0           10.1.1.2        Vlan2
10.1.1.2/32         Direct  0    0           127.0.0.1       InLoop0
10.2.1.0/24         BGP     255  0           2.2.2.9         NULL0
10.3.1.0/24         BGP     255  0           2.2.2.9         NULL0
10.4.1.0/24         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
```

Spoke-CE 1 和 Spoke-CE 2 之间可以 ping 通。从 TTL 值可以推算出 Spoke-CE 1 到 Spoke-CE 2 经过 6 跳（255-250），即 Spoke-CE 1 和 Spoke-CE 2 之间的流量需要通过 Hub-CE 转发。以 Spoke-CE 1 为例：

```
[Spoke-CE1] ping 10.2.1.1
  PING 10.2.1.1: 56  data bytes, press CTRL_C to break
    Reply from 10.2.1.1: bytes=56 Sequence=1 ttl=250 time=3 ms
    Reply from 10.2.1.1: bytes=56 Sequence=2 ttl=250 time=3 ms
    Reply from 10.2.1.1: bytes=56 Sequence=3 ttl=250 time=2 ms
    Reply from 10.2.1.1: bytes=56 Sequence=4 ttl=250 time=2 ms
    Reply from 10.2.1.1: bytes=56 Sequence=5 ttl=250 time=2 ms
  --- 10.2.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
        round-trip min/avg/max = 2/2/3 ms
```

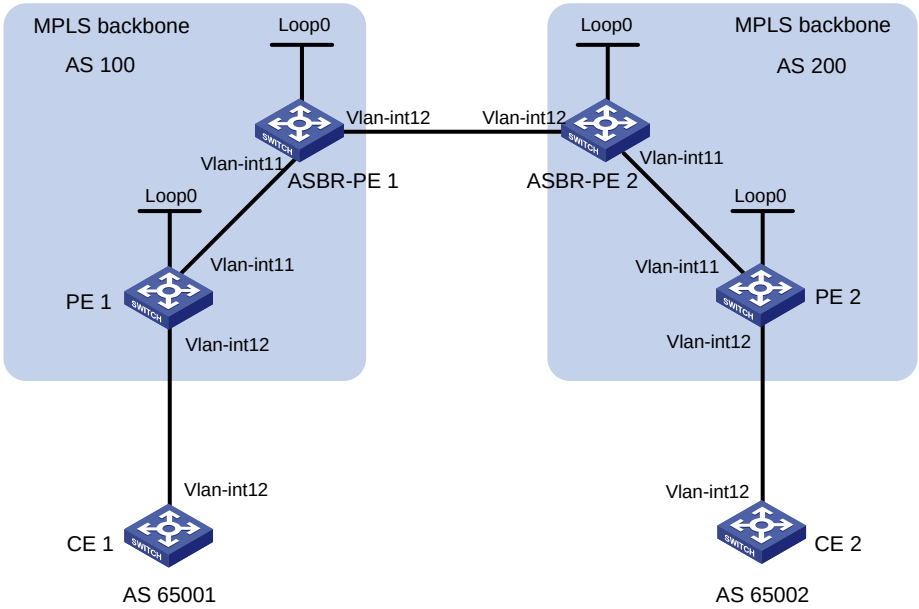
1.10.4 配置跨域VPN-OptionA方式示例

1. 组网需求

- CE 1 和 CE 2 属于同一个 VPN。
- CE 1 通过 AS100 的 PE 1 接入，CE 2 通过 AS200 的 PE 2 接入。
- 采用 OptionA 方式实现跨域的 MPLS L3VPN，即，采用 VRF-to-VRF 方式管理 VPN 路由。
- 同一个 AS 内部的 MPLS 骨干网使用 OSPF 作为 IGP。

2. 组网图

图1-22 配置跨域 VPN-OptionA 方式组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 在 MPLS 骨干网上配置 IGP 协议，实现骨干网内互通

本例中采用 OSPF，具体配置步骤略。

说明

需要将作为 LSR ID 的 LoopBack 接口的 32 位地址通过 OSPF 发布出去。

配置完成后，ASBR-PE 与本 AS 的 PE 之间应能建立 OSPF 邻居，执行 **display ospf peer** 命令可以看到邻居达到 FULL 状态，PE 之间能学习到对方的 Loopback 地址。

ASBR-PE 与本 AS 的 PE 之间能够互相 ping 通。

(2) 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 PE 1 的 MPLS 基本能力，并在与 ASBR-PE 1 相连的接口上使能 LDP。

```
<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
```

配置 ASBR-PE 1 的 MPLS 基本能力，并在与 PE 1 相连的接口上使能 LDP。

```
<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
```

配置 ASBR-PE 2 的 MPLS 基本能力，并在与 PE 2 相连的接口上使能 LDP。

```
<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
```

配置 PE 2 的 MPLS 基本能力，并在与 ASBR-PE 2 相连的接口上使能 LDP。

```
<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
```

上述配置完成后，同一 AS 的 PE 和 ASBR-PE 之间应该建立起 LDP 邻居，在各设备上执行 **display mpls ldp session** 命令可以看到显示结果中 Status 字段为“Operational”。

(3) 在 PE 设备上配置 VPN 实例，将 CE 接入 PE

说明

同一 AS 内的 ASBR-PE 与 PE 的 VPN 实例的 VPN Target 应能匹配，不同 AS 的 PE 的 VPN 实例的 VPN Target 则不需要匹配。

配置 CE 1。

```
<CE1> system-view
[CE1] interface vlan-interface 12
[CE1-Vlan-interface12] ip address 10.1.1.1 24
[CE1-Vlan-interface12] quit
```

配置 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
```

配置 CE 2。

```
<CE2> system-view
[CE2] interface vlan-interface 12
[CE2-Vlan-interface12] ip address 10.2.1.1 24
[CE2-Vlan-interface12] quit
```

配置 PE 2。

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance] route-distinguisher 200:1
[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
```

配置 ASBR-PE 1：创建 VPN 实例，并将此实例绑定到连接 ASBR-PE 2 的接口（ASBR-PE 1 认为 ASBR-PE 2 是自己的 CE）。

```
[ASBR-PE1] ip vpn-instance vpn1
[ASBR-PE1-vpn-instance-vpn1] route-distinguisher 100:2
[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
```

配置 ASBR-PE 2：创建 VPN 实例，并将此实例绑定到连接 ASBR-PE 1 的接口（ASBR-PE 2 认为 ASBR-PE 1 是自己的 CE）。

```
[ASBR-PE2] ip vpn-instance vpn1
[ASBR-PE2-vpn-vpn-vpn1] route-distinguisher 200:2
[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
```

上述配置完成后，在各 PE 设备上执行 **display ip vpn-instance** 命令能正确显示 VPN 实例配置。各 PE 能 ping 通 CE。ASBR-PE 之间也能互相 ping 通。

(4) 在 PE 与 CE 之间建立 EBGP 对等体，引入 VPN 路由

配置 CE 1。

```
[CE1] bgp 65001
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] import-route direct
[CE1-bgp] quit
```

配置 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
```

配置 CE 2。

```
[CE2] bgp 65002
[CE2-bgp] peer 10.2.1.2 as-number 200
[CE2-bgp] import-route direct
[CE2-bgp] quit
```

配置 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) PE 与本 AS 的 ASBR-PE 之间建立 MP-IBGP 对等体，ASBR-PE 之间建立 EBGP 对等体

配置 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
```

配置 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
```

配置 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
```

配置 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) 配置完成后的检验

上述配置完成后，CE 之间能学习到对方的接口路由，CE 1 和 CE 2 能够相互 ping 通。

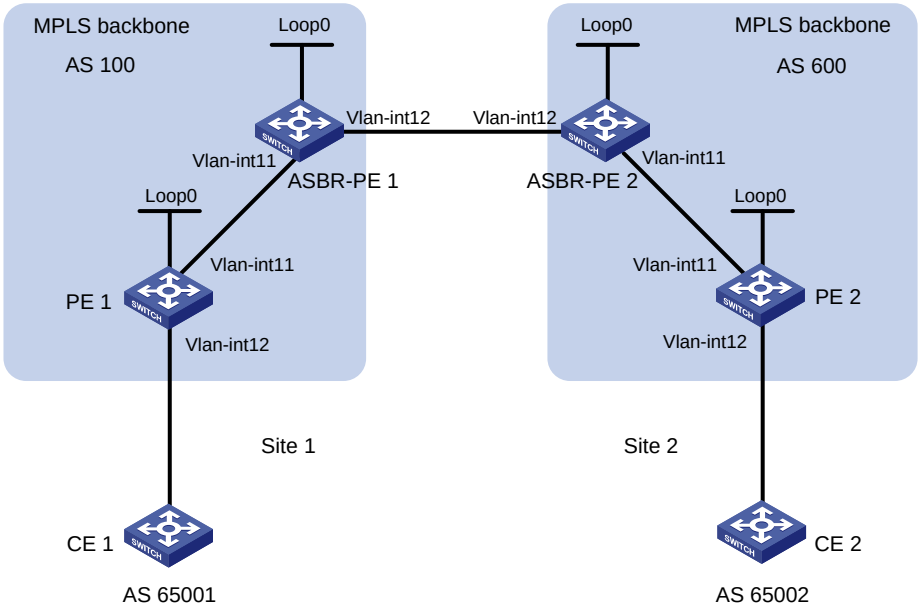
1.10.5 配置跨域VPN-OptionB方式示例

1. 组网需求

- Site 1 和 Site 2 属于同一个 VPN，Site 1 的 CE 1 通过 AS 100 的 PE 1 接入，Site 2 的 CE 2 通过 AS 600 的 PE 2 接入；
- 同一自治系统内的 PE 设备之间运行 IS-IS 作为 IGP；
- PE 1 与 ASBR-PE 1 间通过 MP-IBGP 交换标签 IPv4 路由；
- PE 2 与 ASBR-PE 2 间通过 MP-IBGP 交换标签 IPv4 路由；
- ASBR-PE 1 与 ASBR-PE 2 间通过 MP-EBGP 交换标签 IPv4 路由；
- ASBR 上不对接收的 VPN-IPv4 路由进行 VPN Target 过滤。

2. 组网图

图1-23 配置跨域 VPN-OptionB 方式组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置 PE 1

在 PE 1 上运行 IS-IS。

```
<PE1> system-view
[PE1] isis 1
[PE1-isis-1] network-entity 10.1111.1111.1111.1111.00
[PE1-isis-1] quit
```

配置 LSR ID，使能 MPLS 和 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
```

配置 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```
创建 Loopback0 接口，在接口上运行 IS-IS。
```
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 2.2.2.9 32
[PE1-LoopBack0] isis enable 1
[PE1-LoopBack0] quit
```
创建一个 VPN 实例，名为 vpn1，配置 RD 和 VPN Target 属性。
```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 11:11
[PE1-vpn-instance-vpn1] vpn-target 3:3 import-extcommunity
[PE1-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE1-vpn-instance-vpn1] quit
```
将连接 CE 1 的接口绑定到创建的 VPN 实例。
```
[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
```
在 PE 1 上运行 BGP。
```
[PE1] bgp 100
```
配置 IBGP 对等体 3.3.3.9 为 VPNv4 对等体。
```
[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
```
将直连路由引入 VPN1 的 VPN 路由表。
```
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import-route direct
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```
(2)　配置 ASBR-PE 1

在 ASBR-PE 1 上运行 IS-IS。
```
<ASBR-PE1> system-view
[ASBR-PE1] isis 1
[ASBR-PE1-isis-1] network-entity 10.2222.2222.2222.2222.00
[ASBR-PE1-isis-1] quit
```
配置 LSR ID，使能 MPLS 和 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
```
配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。
```
[ASBR-PE1] interface vlan-interface11
[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
```

配置接口 Vlan-interface12，使能 MPLS。

```
[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
```

创建 Loopback0 接口，并运行 IS-IS。

```
[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
```

在 ASBR-PE 1 上运行 BGP

```
[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
```

不对接收的 VPNv4 路由进行 Import VPN-target 过滤。

```
[ASBR-PE1-bgp] ipv4-family vpnv4
[ASBR-PE1-bgp-af-vpnv4] undo policy vpn-target
```

将 IBGP 对等体 2.2.2.9 和 EBGP 对等体 11.0.0.1 都配置为 VPNv4 对等体。

```
[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) 配置 ASBR-PE 2

在 ASBR-PE 2 上运行 IS-IS。

```
<ASBR-PE2> system-view
[ASBR-PE2] isis 1
[ASBR-PE2-isis-1] network-entity 10.3333.3333.3333.3333.00
[ASBR-PE2-isis-1] quit
```

配置 LSR ID，使能 MPLS 和 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
```

配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```

配置接口 Vlan-interface12，使能 MPLS。

```
[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
```

创建 Loopback0 接口，并运行 IS-IS。

```
[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
```

在 ASBR-PE 2 上运行 BGP。

```
[ASBR-PE2] bgp 600
[ASBR-PE2-bgp] peer 11.0.0.2 as-number 100
[ASBR-PE2-bgp] peer 5.5.5.9 as-number 600
[ASBR-PE2-bgp] peer 5.5.5.9 connect-interface loopback 0
```

不对接收的 VPNv4 路由进行 Import VPN-target 过滤。

```
[ASBR-PE2-bgp] ipv4-family vpnv4
[ASBR-PE2-bgp-af-vpnv4] undo policy vpn-target
```

将 IBGP 对等体 5.5.5.9 和 EBGP 对等体 11.0.0.2 都配置为 VPNv4 对等体。

```
[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) 配置 PE 2

在 PE 2 上运行 IS-IS。

```
<PE2> system-view
[PE2] isis 1
[PE2-isis-1] network-entity 10.4444.4444.4444.4444.00
[PE2-isis-1] quit
```

配置 LSR ID，使能 MPLS 和 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
```

配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```

创建 Loopback0 接口，在接口上运行 IS-IS。

```
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 5.5.5.9 32
```

```
[PE2-LoopBack0] isis enable 1
[PE2-LoopBack0] quit
```

创建一个 VPN 实例，名为 vpn1，配置 RD 和 VPN Target 属性。

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 12:12
[PE2-vpn-instance-vpn1] vpn-target 3:3 import-extcommunity
[PE2-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE2-vpn-instance-vpn1] quit
```

将连接 CE 2 的接口绑定到创建的 VPN 实例。

```
[PE2] interface Vlan-interface12
[PE2-Vlan-interface12] ip binding vpn-instance vpn1
[PE2-Vlan-interface12] ip address 20.0.0.1 8
[PE2-Vlan-interface12] quit
```

在 PE 2 上运行 BGP。

```
[PE2] bgp 600
```

配置 IBGP 对等体 4.4.4.9 为 VPNv4 对等体。

```
[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
```

将直连路由引入 VPN1 的 VPN 路由表。

```
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] import-route direct
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

(5) 配置完成后的检验

配置完成后，在 PE 2 上对 PE 1 进行 Ping 操作，可以 Ping 通：

```
[PE2] ping –vpn-instance vpn1 30.0.0.1
```

在 PE 1 上对 PE 2 进行 Ping 操作，也可以 Ping 通：

```
[PE1] ping –vpn-instance vpn1 20.0.0.1
```

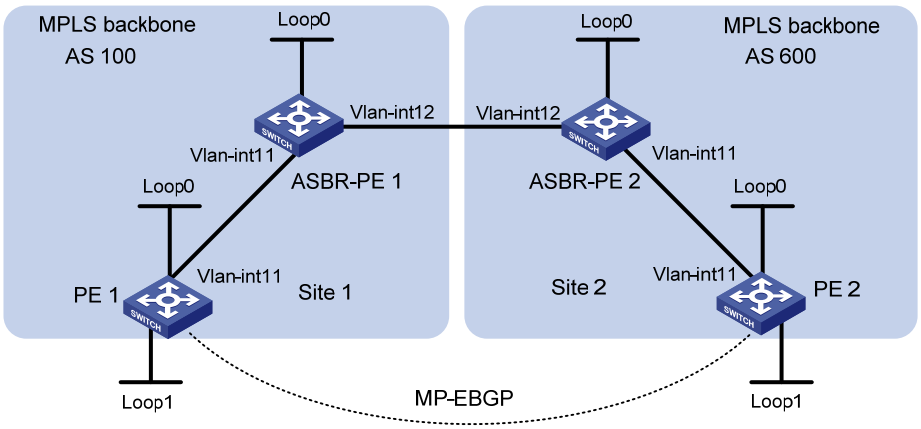
1.10.6 配置跨域VPN-OptionC方式示例

1. 组网需求

- Site 1 和 Site 2 属于同一个 VPN，Site 1 通过 AS 100 的 PE 1 接入，Site 2 通过 AS 600 的 PE 2 接入；
- 同一自治系统内的 PE 设备之间运行 IS-IS 作为 IGP；
- PE 1 与 ASBR-PE 1 间通过 MP-IBGP 交换标签 IPv4 路由；
- PE 2 与 ASBR-PE 2 间通过 MP-IBGP 交换标签 IPv4 路由；
- PE 1 与 PE 2 建立 MP-EBGP 对等体；
- ASBR-PE 1 和 ASBR-PE 2 上分别配置路由策略，对从对方接收的路由压入标签；
- ASBR-PE 1 与 ASBR-PE 2 间通过 MP-EBGP 交换标签 IPv4 路由。

2. 组网图

图1-24 配置跨域 VPN-OptionC 方式组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置 PE 1

在 PE 1 上运行 IS-IS。

```
<PE1> system-view
[PE1] isis 1
[PE1-isis-1] network-entity 10.1111.1111.1111.1111.00
[PE1-isis-1] quit
```

配置 LSR ID，使能 MPLS 和 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
```

配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```

创建 Loopback0 接口，在接口上运行 IS-IS。

```
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 2.2.2.9 32
```

```
[PE1-LoopBack0] isis enable 1
[PE1-LoopBack0] quit
```
创建 VPN 实例，名称为 vpn1，为其配置 RD 和 VPN Target 属性。
```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 11:11
[PE1-vpn-instance-vpn1] vpn-target 3:3 import-extcommunity
[PE1-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE1-vpn-instance-vpn1] quit
```
创建 Loopback1 接口，并将此接口绑定到 vpn1。
```
[PE1] interface loopback 1
[PE1-LoopBack1] ip binding vpn-instance vpn1
[PE1-LoopBack1] ip address 30.0.0.1 32
[PE1-LoopBack1] quit
```
在 PE 1 上运行 BGP。
```
[PE1] bgp 100
```
配置 PE 1 向 IBGP 对等体 3.3.3.9 发布标签路由及从 3.3.3.9 接收标签路由的能力。
```
[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
```
配置 PE 1 到 EBGP 对等体 5.5.5.9 的最大跳数为 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
```
配置对等体 5.5.5.9 作为 VPNv4 对等体。
```
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 5.5.5.9 enable
[PE1-bgp-af-vpnv4] quit
```
将直连路由引入 vpn1 的路由表。
```
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import-route direct
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```
(2)　配置 ASBR-PE1

在 ASBR-PE 1 上运行 IS-IS。
```
<ASBR-PE1> system-view
[ASBR-PE1] isis 1
[ASBR-PE1-isis-1] network-entity 10.2222.2222.2222.2222.00
[ASBR-PE1-isis-1] quit
```
配置 LSR ID，使能 MPLS 和 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
```

配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```

配置接口 Vlan-interface12，并在接口上使能 MPLS。

```
[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
```

创建 Loopback0 接口，在接口上运行 IS-IS。

```
[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
```

创建路由策略。

```
[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
```

在 ASBR-PE 1 上运行 BGP，引入 IS-IS 进程 1 的路由。

```
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] import-route isis 1
```

对向 IBGP 对等体 2.2.2.9 发布的路由应用已配置的路由策略 policy2。

```
[ASBR-PE1-bgp] peer 2.2.2.9 as-number 100
[ASBR-PE1-bgp] peer 2.2.2.9 route-policy policy2 export
```

向 IBGP 对等体 2.2.2.9 发布标签路由及从 2.2.2.9 接收标签路由的能力。

```
[ASBR-PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[ASBR-PE1-bgp] peer 2.2.2.9 label-route-capability
```

对向 EBGP 对等体 11.0.0.1 发布的路由应用已配置的路由策略 policy1。

```
[ASBR-PE1-bgp] peer 11.0.0.1 as-number 600
[ASBR-PE1-bgp] peer 11.0.0.1 route-policy policy1 export
```

向 EBGP 对等体 11.0.0.1 发布标签路由及从 11.0.0.1 接收标签路由的能力。

```
[ASBR-PE1-bgp] peer 11.0.0.1 label-route-capability
[ASBR-PE1-bgp] quit
```

(3) 配置 ASBR-PE 2

在 ASBR-PE 2 上运行 IS-IS。

```
<ASBR-PE2> system-view
[ASBR-PE2] isis 1
[ASBR-PE2-isis-1] network-entity 10.3333.3333.3333.3333.00
```

```
[ASBR-PE2-isis-1] quit
```
配置 LSR ID，使能 MPLS 和 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
```
配置接口 Vlan-interface11，在接口上运行 IS-IS，并在接口上使能 MPLS 和 LDP。
```
[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
```
创建 Loopback0 接口，在接口上运行 IS-IS。
```
[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
```
配置接口 Vlan-interface12，在接口上使能 MPLS。
```
[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
```
创建路由策略。
```
[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
```
在 ASBR-PE 2 上运行 BGP，引入 IS-IS 进程 1 的路由。
```
[ASBR-PE2] bgp 600
[ASBR-PE2-bgp] import-route isis 1
```
向 IBGP 对等体 5.5.5.9 发布标签路由及从 5.5.5.9 接收标签路由的能力。
```
[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
```
对向 IBGP 对等体 5.5.5.9 发布的路由应用已配置的路由策略 policy2。
```
[ASBR-PE2-bgp] peer 5.5.5.9 route-policy policy2 export
```
对向 EBGP 对等体 11.0.0.2 发布的路由应用已配置的路由策略 policy1。
```
[ASBR-PE2-bgp] peer 11.0.0.2 as-number 100
[ASBR-PE2-bgp] peer 11.0.0.2 route-policy policy1 export
```

向 EBGP 对等体 11.0.0.2 发布标签路由及从 11.0.0.2 接收标签路由的能力。

```
[ASBR-PE2-bgp] peer 11.0.0.2 label-route-capability
[ASBR-PE2-bgp] quit
```

(4) 配置 PE 2

在 PE 2 上运行 IS-IS。

```
<PE2> system-view
[PE2] isis 1
[PE2-isis-1] network-entity 10.4444.4444.4444.4444.00
[PE2-isis-1] quit
```

配置 LSR ID，使能 MPLS 和 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
```

配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```

创建 Loopback0 接口，在接口上运行 IS-IS。

```
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 5.5.5.9 32
[PE2-LoopBack0] isis enable 1
[PE2-LoopBack0] quit
```

创建 VPN 实例，名称为 vpn1，为其配置 RD 和 VPN Target 属性。

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 11:11
[PE2-vpn-instance-vpn1] vpn-target 3:3 import-extcommunity
[PE2-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE2-vpn-instance-vpn1] quit
```

创建 Loopback1 接口，并将此接口绑定到 vpn1。

```
[PE2] interface loopback 1
[PE2-LoopBack1] ip binding vpn-instance vpn1
[PE2-LoopBack1] ip address 20.0.0.1 32
[PE2-LoopBack1] quit
```

在 PE 2 上运行 BGP。

```
[PE2] bgp 600
```

配置 PE 2 向 IBGP 对等体 4.4.4.9 发布标签路由及从 4.4.4.9 接收标签路由的能力。

```
[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
```

配置 PE 2 到 EBGP 对等体 2.2.2.9 的最大跳数为 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
```

配置对等体 2.2.2.9 作为 VPNv4 对等体。

```
[PE2-bgp] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 2.2.2.9 enable
[PE2-bgp-af-vpnv4] quit
```

将直连路由引入 vpn1 的路由表。

```
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] import-route direct
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

配置完成后，在 PE2 上对 PE1 进行 Ping 操作，可以 Ping 通：

```
[PE2] ping –vpn-instance vpn1 30.0.0.1
```

在 PE1 上对 PE2 进行 Ping 操作，也可以 Ping 通：

```
[PE1] ping –vpn-instance vpn1 20.0.0.1
```

1.10.7 配置运营商的运营商示例

1. 组网需求

在 图 1-25 中：

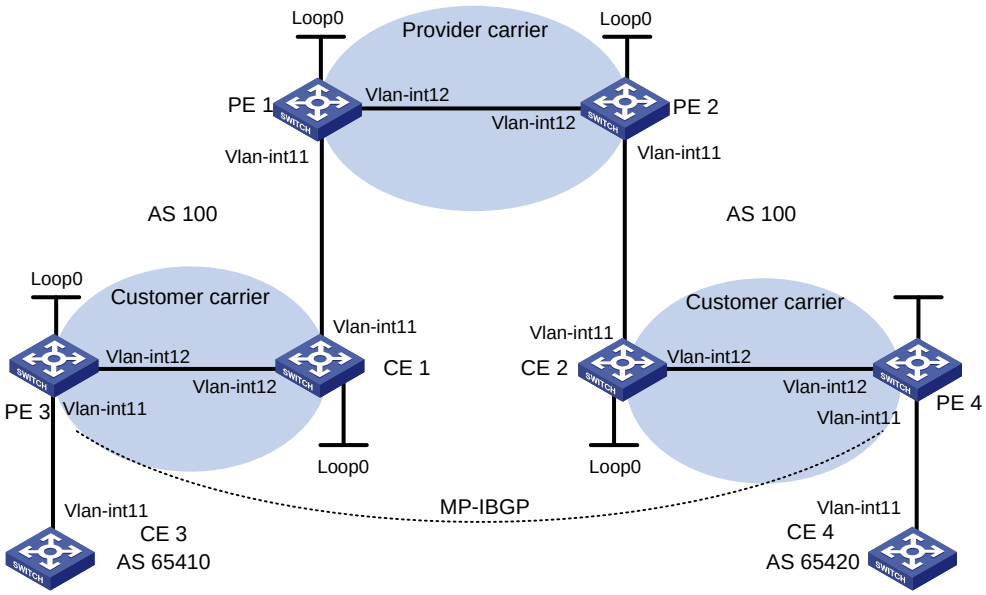
- PE 1 和 PE 2 是一级运营商骨干网的 PE 设备，为二级运营商提供 VPN 服务；
- CE 1 和 CE 2 是同一个二级运营商的设备，作为 CE 接入一级运营商的骨干网；
- PE 3 和 PE 4 是二级运营商的 PE 设备，为二级运营商的客户提供 MPLS L3VPN 服务；
- CE 3 和 CE 4 是二级运营商的客户。

配置运营商的运营商关键在于配置两类路由的交换：

- 二级运营商 VPN 内部路由在一级运营商骨干网上的交换：一级运营商将二级运营商作为自己的 CE 接入；
- 二级运营商本身客户的 VPN 路由在二级运营商 PE 设备间的交换：需要在二级运营商 PE 设备（PE 3 和 PE 4）间建立 MP-IBGP 对等体关系。

2. 组网图

图1-25 配置 Carriers' carriers 配置组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置一级运营商骨干网的 MPLS L3VPN，使用 IS-IS 作为骨干网的 IGP 协议，PE 1 和 PE 2 之间使能 LDP，并建立 MP-IBGP 对等体关系

配置 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] mpls ldp transport-address interface
[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
```

说明

PE 2 的配置与 PE 1 类似，配置过程省略。

配置完成后，在 PE 1 或 PE 2 上执行 **display mpls ldp session** 命令可以看到 LDP 会话建立成功；执行 **display bgp peer** 命令可以看到 BGP 对等体关系已建立，并达到 Established 状态；执行 **display isis peer** 命令可以看到 IS-IS 邻居关系已建立，状态为 up。

以 PE1 为例：

```
[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) 配置二级运营商网络：使用 IS-IS 作为 IGP 协议，PE 3 和 CE 1、PE 4 和 CE 2 之间分别使能 LDP

配置 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
```

配置 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
```

配置完成后，PE 3 和 CE 1 之间应能建立 LDP 和 IS-IS 邻居关系。

说明

PE 4 和 CE 2 之间的配置与 PE 3 和 CE 1 之间的配置类似，配置过程省略。

(3) 配置二级运营商 CE 接入到一级运营商的 PE，并在 PE 上配置 IS-IS 引入 BGP 路由、BGP 引入 IS-IS 路由

配置 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
```

配置 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
```

配置完成后，PE 1 和 CE 1 之间应能建立 LDP 和 IS-IS 邻居关系。

说明

PE 2 和 CE 2 之间的配置与 PE 1 和 CE 1 之间的配置类似，配置过程省略。

(4) 配置二级运营商的客户接入 PE

配置 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 100
[CE3-bgp] import-route direct
[CE3-bgp] quit
```

配置 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] 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
```

说明

PE 4 和 CE 4 之间的配置与 PE 3 和 CE 3 之间的配置类似，配置过程省略。

(5) 在二级运营商的 PE 之间建立 MP-IBGP 对等体关系，交换二级运营商的客户的 VPN 路由

配置 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
```

说明

PE 4 的配置与 PE 3 类似，配置过程省略。

(6) 配置完成后的检验

所有配置完成后，在 PE 1 和 PE 2 上执行 **display ip routing-table** 命令，可以看到 PE 1 和 PE 2 的公网路由表中只有一级运营商网络的路由。以 PE 1 为例：

```
[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
```

在 PE 1 和 PE 2 上执行 **display ip routing-table vpn-instance** 命令，可以看到 VPN 路由表中有二级运营商网络的内部路由，但没有二级运营商维护的 VPN 路由。以 PE 1 为例：

```
[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
```

在 CE 1 和 CE 2 上执行 **display ip routing-table** 命令，可以看到公网路由表中有二级运营商网络的内部路由，但没有二级运营商维护的 VPN 路由。以 CE 1 为例：

```
[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
```

在 PE 3 和 PE 4 上执行 **display ip routing-table** 命令，可以看到公网路由表中有二级运营商网络的内部路由。以 PE 3 为例：

```
[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
```

在 PE 3 和 PE 4 上执行 **display ip routing-table vpn-instance** 命令，可以看到 VPN 路由表中有远端 VPN 客户的路由。以 PE 3 为例：

```
[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 和 PE 4 可以相互 Ping 通：

```
[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 和 CE 4 可以互相 Ping 通。

```
[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
```

1.10.8 配置嵌套VPN示例

1. 组网需求

运营商向客户提供嵌套VPN服务。如 图 1-26 所示：

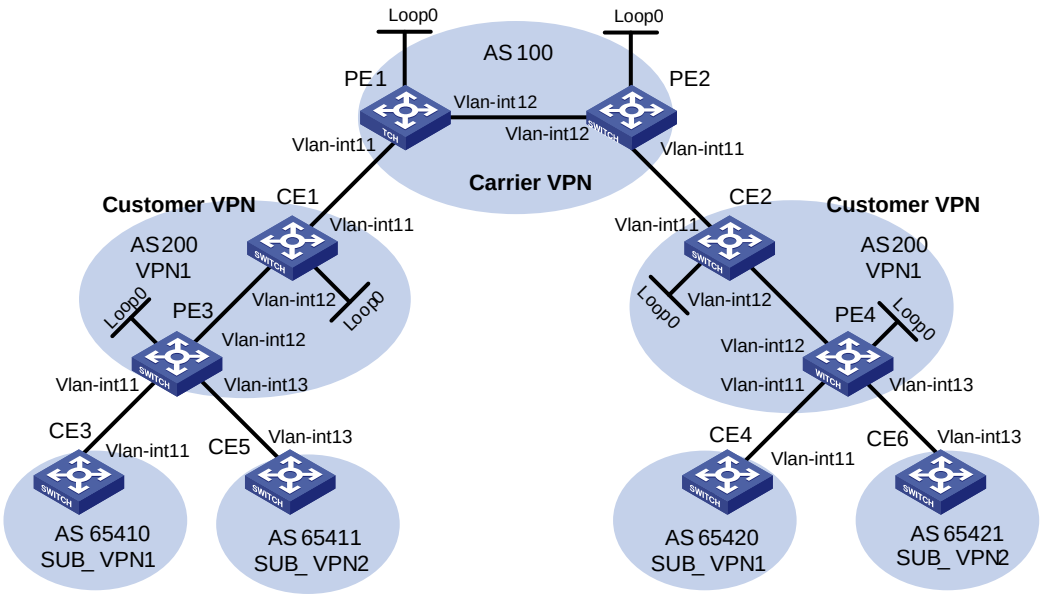
- PE 1 和 PE 2 是运营商骨干网的 PE 设备，支持嵌套 VPN 功能；
- CE 1 和 CE 2 是 VPN 用户的 CE 设备，接入运营商的骨干网，该 CE 设备支持发送 VPNv4 路由；
- PE 3 和 PE 4 是用户网络内部的 PE 设备，支持 MPLS L3VPN；
- CE 3、CE 4、CE 5 和 CE 6 是用户网络内部的子 VPN 的 CE 设备；

配置嵌套 VPN 的关键在于理解子 VPN 路由在运营商 PE 设备上的处理过程：

- 运营商 PE（PE 1 和 PE 2）收到用户 CE（CE 1 和 CE 2）发送来的 VPNv4 路由时，需要将该 VPNv4 路由的 RD 更换为用户 CE 所处 VPN 的 RD，同时将用户 CE 所处 VPN 的 ERT 添加到路由的扩展团体属性列表中，然后再按照一般的 VPNv4 路由发送出去；
- 为了实现用户网络内部子 VPN 的路由在用户 PE 和运营商 PE 间交换，需要在运营商 PE 和用户 CE 间建立 MP-EBGP 对等体关系。

2. 组网图

图1-26 嵌套 VPN 组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置运营商骨干网的 MPLS L3VPN，使用 IS-IS 作为骨干网的 IGP 协议，PE 1 和 PE 2 之间使能 LDP，并建立 MP-IBGP 对等体关系

配置 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
```

说明

PE 2 的配置与 PE 1 类似，配置过程省略。

配置完成后，在 PE 1 或 PE 2 上执行 **display mpls ldp session** 命令可以看到 LDP 会话建立成功；执行 **display bgp peer** 命令可以看到 BGP 对等体关系已建立，并达到 Established 状态；执行 **display isis peer** 命令可以看到 IS-IS 邻居关系已建立，状态为 up。

以 PE 1 为例：

```
[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) 配置用户网络：使用 IS-IS 作为 IGP 协议，PE 3 和 CE 1、PE 4 和 CE 2 之间分别使能 LDP

配置 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
```

配置 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
```

配置完成后，PE 3 和 CE 1 之间可以建立 LDP 和 IS-IS 邻居关系。

说明

PE 4 和 CE 2 之间的配置与 PE 3 和 CE 1 之间的配置类似，配置过程省略。

(3) 配置用户 CE 接入到运营商的 PE

配置 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
```

配置 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
```

说明

PE 2 和 CE 2 之间的配置与 PE 1 和 CE 1 之间的配置类似，配置过程省略。

(4) 配置子 VPN 的 CE 接入用户网络的 PE

配置 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 200
[CE3-bgp] import-route direct
[CE3-bgp] quit
```

配置 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
```

配置 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
```

说明

PE 4 和 CE 4，CE 6 之间的配置与 PE 3 和 CE 3，CE 5 之间的配置类似，配置过程省略。

(5) 在运营商的 PE 和用户的 CE 之间建立 MP-EBGP 对等体关系，交换用户的 VPNv4 路由

配置 PE 1，使能嵌套 VPN 特性。

```
[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
```

配置 CE 1，使能 VPNv4 能力，使之与 PE 1 建立 VPNv4 邻居关系。

```
[CE1] bgp 200
[CE1-bgp] ipv4-family vpnv4
[CE1-bgp-af-vpnv4] peer 11.1.1.2 enable
```

配置允许本地 AS 号在所接收的路由的 AS_PATH 属性中出现。

```
[CE1-bgp-af-vpnv4] peer 11.1.1.2 allow-as-loop 2
```

配置接收所有 VPNv4 路由。

```
[CE1-bgp-af-vpnv4] undo policy vpn-target
[CE1-bgp-af-vpnv4] quit
[CE1-bgp] quit
```

说明

PE 2 和 CE 2 之间的配置与 PE 1 和 CE 1 之间的配置类似，配置过程省略。

(6) 在用户网络内部的 PE 和 CE 之间建立 MP-IBGP 对等体关系，交换用户内部子 VPN 的 VPNv4 路由

配置 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
```

配置允许本地 AS 号在所接收的路由的 AS_PATH 属性中出现。

```
[PE3-bgp-af-vpnv4] peer 2.2.2.9 allow-as-loop 2
[PE3-bgp-af-vpnv4] quit
[PE3-bgp] quit
```

配置 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
```

说明

PE 4 和 CE 2 之间的配置与 PE 3 和 CE 1 之间的配置类似，配置过程省略。

(7) 配置完成后的检验

所有配置完成后，在 PE 1 和 PE 2 上执行 **display ip routing-table** 命令，可以看到 PE 1 和 PE 2 的公网路由表中只有运营商网络的路由。以 PE 1 为例：

```
[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
```

在 PE 1 和 PE 2 上执行 **display ip routing-table vpn-instance** 命令，可以看到 VPN 路由表中有用户子 VPN 网络的路由。以 PE 1 为例：

```
[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
```

在 CE 1 和 CE 2 上执行 **display bgp vpnv4 all routing-table** 命令，可以看到用户网络中 VPNv4 路由表中有子 VPN 网络的内部路由。以 CE 1 为例：

```
[CE1] display bgp vpnv4 all routing-table
BGP Local router ID is 11.11.11.11
 Status codes: * - valid, ^ - VPN 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
```

在 PE 3 和 PE 4 上执行 **display ip routing-table vpn-instance SUB_VPN1** 命令，可以看到私网路由表中有从运营商 PE 发布到用户网络子 VPN 内部的路由。以 PE 3 为例：

```
[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
```

在 CE 3 和 CE 4 上执行 **display ip routing-table** 命令，可以看到路由表中有远端子 VPN 的路由。以 CE 3 为例：

```
[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       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
```

在 CE 5 和 CE 6 上执行 display ip routing-table 命令，可以看到路由表中有远端子 VPN 的路由。以 CE 5 为例：

```
[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 和 CE 4 可以互相 Ping 通。

```
[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
```

CE 5 和 CE 6 可以互相 Ping 通。

```
[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 和 CE 6 不能互相 Ping 通。

```
[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
```

1.10.9 配置HoVPN示例

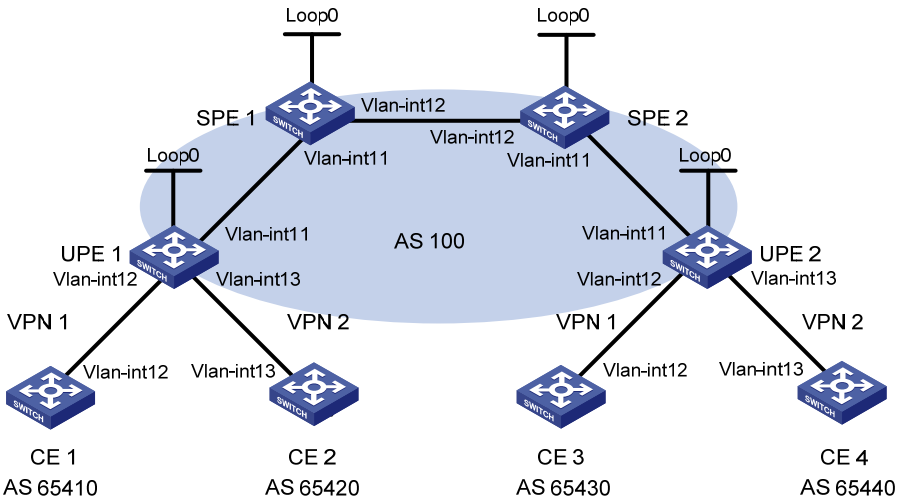
1. 组网需求

以一个包括省骨干和地市的 MPLS VPN 网络为例：

- SPE 作为省网的 PE 设备，接入地市的 MPLS VPN 网络；
- UPE 作为下层地市网络的 PE 设备，最终接入 VPN 客户。对 UPE 的性能要求低于对 SPE 的性能要求。
- SPE 将通过路由策略的路由发送给 UPE，限制不同 Site 之间的互相访问权限，使得 VPN 1 内的 CE 1 和 CE 3 可以互相访问，VPN 2 内的 CE 2 和 CE 4 不能互相访问。

2. 组网图

图1-27 配置 HoVPN 组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置 UPE 1

配置 MPLS 基本能力和 MPLS LDP 能力，建立 LDP LSP。

```
<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
```

配置 IGP 协议，以 OSPF 为例。

```
[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
```

配置 VPN 实例 vpn1 和 vpn2，将 CE 1 和 CE 2 接入 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
```

配置 UPE 1 与 SPE 1 建立 MP-IBGP 对等体，并引入 VPN 路由。

```
[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) 配置 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) 配置 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) 配置 UPE 2

配置 MPLS 基本能力和 MPLS LDP 能力，建立 LDP LSP。

```
<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
```

配置 IGP 协议，以 OSPF 为例。

```
[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
```

配置 VPN 实例 vpn1 和 vpn2，将 CE 3 和 CE 4 接入 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
```

配置 UPE 2 与 SPE 2 建立 MP-IBGP 对等体，并引入 VPN 路由。

```
[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) 配置 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) 配置 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) 配置 SPE 1

配置 MPLS 基本能力和 MPLS LDP 能力，建立 LDP LSP。

```
<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
```

配置 IGP 协议，以 OSPF 为例。

```
[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
```

配置 VPN 实例 vpn1 和 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
```

配置 SPE 1 与 UPE 1 建立 MP-IBGP 对等体，指定 UPE 1，并引入 VPN 路由。

```
[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
```

配置 SPE 1 向 UPE 1 发送通过策略的路由信息，允许 CE 3 的路由发送给 UPE 1。

```
[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) 配置 SPE 2

配置 MPLS 基本能力和 MPLS LDP 能力，建立 LDP LSP。

```
<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
```

配置 IGP 协议，以 OSPF 为例。

```
[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
```

配置 VPN 实例 vpn1 和 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
```

配置 SPE 2 与 UPE 2 建立 MP-IBGP 对等体，指定 UPE 2，并引入 VPN 路由。

```
[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
```

配置 SPE 2 向 UPE 2 发送通过策略的路由信息，允许 CE 1 的路由发送给 UPE 2。

```
[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
```

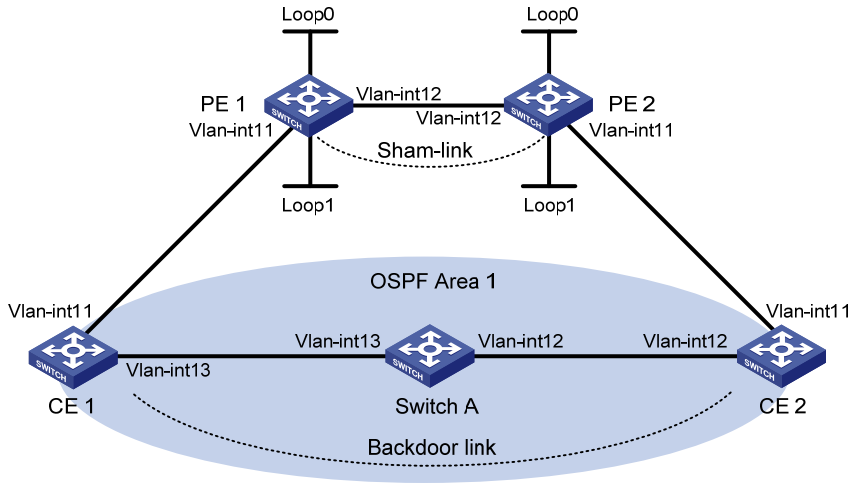
1.10.10 配置OSPF伪连接

1. 组网需求

- CE 1 和 CE 2 都属于 VPN 1，它们分别接入 PE 1 和 PE 2；
- CE 1 和 CE 2 在同一个 OSPF 区域中；
- CE 1 与 CE 2 之间的 VPN 流量通过 MPLS 骨干网转发，不使用 OSPF 的区域内路由。

2. 组网图

图1-28 OSPF 伪连接配置组网图



设备	接口	IP地址	设备	接口	IP地址
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			

3. 配置步骤

(1) 配置用户网络上的 OSPF

在CE 1、Switch A、CE 2 上配置普通OSPF，发布 图 1-28 中所示各接口的网段地址。具体配置过程略。

配置完成后，CE 1 和 CE 2 应学到到对端 Vlan-interface1 接口的 OSPF 路由。

以 CE 1 为例：

```
<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) 在骨干网上配置 MPLS L3VPN

配置 PE 1。

配置 PE 1 的 MPLS 基本能力和 MPLS LDP 能力，建立 LDP LSP。

```
<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
```

配置 PE 1 的 MP-IBGP 对等体为 PE2。

```
[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
```

配置 PE 1 的 OSPF。

```
[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
```

配置 PE 2 的 MPLS 基本能力和 MPLS LDP 能力，建立 LDP LSP。

```
<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
```

配置 PE 2 的 MP-IBGP 对等体为 PE1。

```
[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
```

配置 PE 2 的 OSPF。

```
[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) 配置 PE 接入 CE

配置 PE 1 接入 CE 1。

```
[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
```

配置 PE 2 接入 CE 2。

```
[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
```

完成上述配置后，在 PE 设备上执行 **display ip routing-table vpn-instance** 命令，可以看到去往对端 CE 的路由是通过用户网络的 OSPF 路由，不是通过骨干网的 BGP 路由。

以 PE 1 为例：

```
[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) 配置 Sham-link

配置 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
```

配置 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
```

完成上述配置后，在 PE 设备上再次执行 **display ip routing-table vpn-instance** 命令，可以看到去往对端 CE 的路由变成了通过骨干网的 BGP 路由，并且有去往 Sham-link 目的地址的路由。

以 PE 1 为例：

```
[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
```

在 CE 设备上执行 **display ip routing-table** 命令，可以看到去往对端 CE 的 OSPF 路由开销变为 10（为 Sham-link 配置的开销），下一跳变为接入 PE 的 Vlan-interface11 接口，即去往对端的 VPN 流量将通过骨干网转发。

以 CE 1 为例：

```
[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
```

在 PE 上执行 **display ospf sham-link** 命令可以看到 Sham-link 的建立情况。

以 PE 1 为例：

```
[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
```

执行 **display ospf sham-link area** 命令可以看到对端状态为 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
```

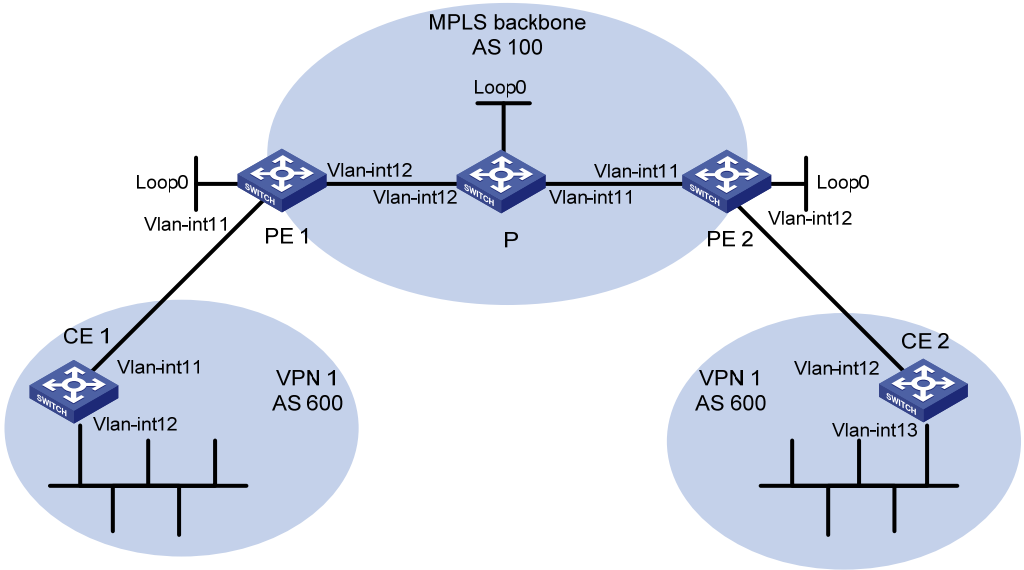
1.10.11 配置BGP的AS号替换

1. 组网需求

如图所示，CE 1 和 CE 2 同属于 VPN 1，分别接入 PE 1 和 PE 2，并且 CE 1 和 CE 2 复用 AS 号 600。

2. 组网图

图1-29 BGP 的 AS 号替换组网图



设备	接口	IP地址	设备	接口	IP地址
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			

3. 配置步骤

(1) 配置基本 MPLS L3VPN

- 在 MPLS 骨干网上配置 OSPF，PE 和 P 之间能够学到对方 Loopback 接口的路由；
- 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP；
- PE 之间建立 MP-IBGP 对等体关系，发布 VPN-IPv4 路由；
- 在 PE 2 上配置 VPN 1 的 VPN 实例，接入 CE 2；
- 在 PE 1 上配置 VPN 1 的 VPN 实例，接入 CE 1；
- PE 1 和 CE 1、PE 2 和 CE 2 之间配置 BGP，将 CE 的路由引入 PE。

完成上述配置后，在 CE 2 上执行 **display ip routing-table** 命令，可以看到 CE 2 能够学到 CE 1 接入 PE 1 的接口所在网段（10.1.1.0/24）的路由，但没有到达 CE 1 内部 VPN（100.1.1.0/24）的路由。CE 1 上也存在同样的现象。

```
<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
```

在 PE 上执行 **display ip routing-table vpn-instance** 命令，可以看到 PE 的 VPN 实例中有到达对端 CE 内部 VPN 的路由。

以 PE 2 为例：

```
<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
```

在 PE 2 上打开 BGP 的 Update 报文调试信息开关，可以看到 PE 2 发布了去往 100.1.1.1/32 的路由，AS 路径信息为“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,
```

在 CE 2 上执行 **display bgp routing-table peer received-routes** 命令，可以看到 CE 2 没有接收 100.1.1.1/32 的路由。

```
<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, ^ - VPN 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) 配置 BGP 的 AS 号替换功能

在 PE 2 上配置 BGP 的 AS 号替换功能。

```
<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
```

可以看到 PE 2 向 CE2 发布的路由中，100.1.1.1/32 的 AS 路径信息由“100 600”变为“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
```

再次查看 CE 2 接收的路由信息和路由表：

```
<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, ^ - VPN 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
```

在 PE 1 上也配置 BGP 的 AS 号替换功能后，CE 1 和 CE 2 的 VLAN 接口能够相互 Ping 通。

```
<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
```

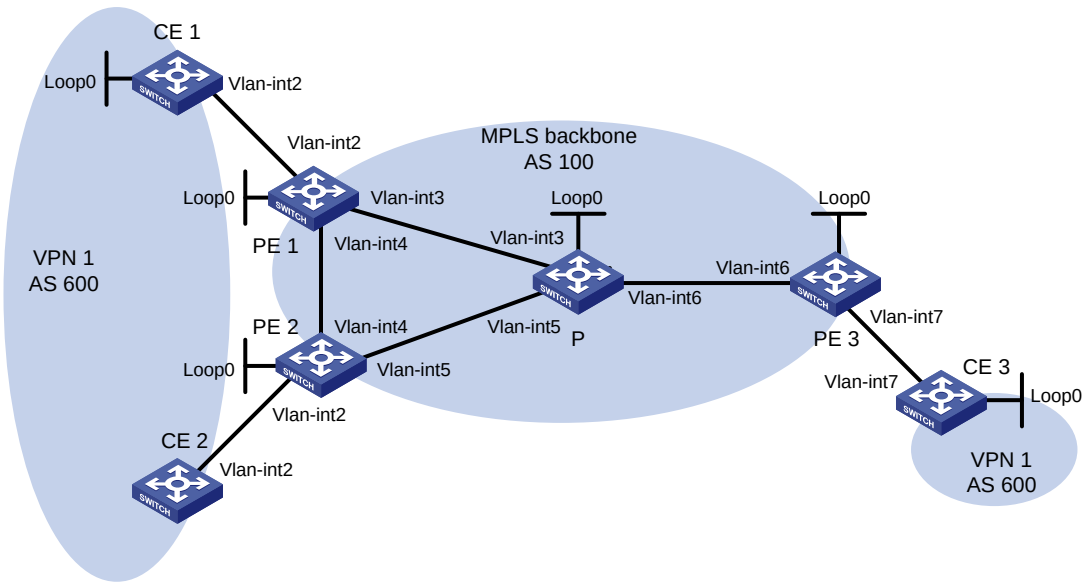
1.10.12 配置BGP的AS号替换和SoO属性

1. 组网需求

- CE 1、CE 2 和 CE 3 同属于 VPN 1，分别接入 PE 1、PE 2 和 PE 3。
- CE 1 和 CE 2 位于同一个站点。
- CE 1、CE 2 和 CE 3 复用 AS 号 600。
- 为了避免路由丢失，在 PE 上配置 AS 号替换；为了避免路由在 CE 1 和 CE 2 之间产生环路，在 PE 1 和 PE 2 上分别通过路由策略为从 CE 1 和 CE 2 学习到的路由添加 SoO 属性。

2. 组网图

图1-30 BGP 的 AS 号替换和 SoO 属性组网图



设备	接口	IP地址	设备	接口	IP地址
CE 1	Loop0	100.1.1.1/32	CE 3	Loop0	200.1.1.1/32
	Vlan-int2	10.1.1.1/24		Vlan-int7	10.3.1.1/24

CE 2	Vlan-int2	10.2.1.1/24	PE 2	Loop0	2.2.2.9/32
PE 1	Loop0	1.1.1.9/32		Vlan-int2	10.2.1.2/24
	Vlan-int2	10.1.1.2/24		Vlan-int4	20.1.1.2/24
	Vlan-int3	30.1.1.1/24		Vlan-int5	40.1.1.1/24
	Vlan-int4	20.1.1.1/24	P	Loop0	3.3.3.9/32
PE 3	Loop0	4.4.4.9/32		Vlan-int3	30.1.1.2/24
	Vlan-int6	50.1.1.2/24		Vlan-int5	40.1.1.2/24
	Vlan-int7	10.3.1.2/24		Vlan-int6	50.1.1.1/24

3. 配置步骤

(1) 配置基本 MPLS L3VPN

- 在 MPLS 骨干网上配置 OSPF，PE 和 P 之间能够学到对方 Loopback 接口的路由；（配置过程略）
- 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP；（配置过程略）
- PE 之间建立 MP-IBGP 对等体关系，发布 VPN-IPv4 路由；（配置过程略）
- 在 PE 1 上配置 VPN 1 的 VPN 实例，接入 CE 1；（配置过程略）
- 在 PE 2 上配置 VPN 1 的 VPN 实例，接入 CE 2；（配置过程略）
- 在 PE 3 上配置 VPN 1 的 VPN 实例，接入 CE 3；（配置过程略）
- PE 1 和 CE 1、PE 2 和 CE 2、PE 3 和 CE 3 之间配置 BGP，将 CE 的路由引入 PE。（配置过程略）

(2) 配置 BGP 的 AS 号替换功能

在PE 1、PE 2 和PE 3 上配置BGP的AS号替换功能，具体配置参见“1.10.11 配置BGP的AS号替换”。

查看 CE 2 接收的路由信息，可以看到 CE 1 发来的路由 100.1.1.1/32。可见，由于 CE 1 和 CE 2 位于同一站点，造成了路由环路。

```
<CE2> display bgp routing-table peer 10.2.1.2 received-routes
 Total Number of Routes: 8
 BGP Local router ID is 10.2.1.1
 Status codes: * - valid, ^ - VPN 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?
 *    10.3.1.0/24      10.2.1.2                             0      100?
 *    10.3.1.1/32      10.2.1.2                             0      100?
 *>   100.1.1.1/32     10.2.1.2                             0      100 100?
 *>   200.1.1.1/32     10.2.1.2                             0      100 100?
<CE2> display ip routing-table
Routing Tables: Public
         Destinations : 10        Routes : 10
Destination/Mask    Proto  Pre  Cost       NextHop         Interface
10.1.1.0/24         BGP    255  0          10.2.1.2        Vlan2
10.1.1.1/32         BGP    255  0          10.2.1.2        Vlan2
```

```
10.2.1.0/24         Direct 0    0          10.2.1.1        Vlan2
10.2.1.1/32         Direct 0    0          127.0.0.1       InLoop0
10.3.1.0/24         BGP    255  0          10.2.1.2        Vlan2
10.3.1.1/32         BGP    255  0          10.2.1.2        Vlan2
100.1.1.1/32        BGP    255  0          10.2.1.2        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
200.1.1.1/32        BGP    255  0          10.2.1.2        Vlan2
```

(3) 配置 BGP 的 SoO 属性

在 PE 1 上配置路由策略 soo，为路由添加 SoO 属性 1:100。

```
<PE1> system-view
[PE1] route-policy soo permit node 10
[PE1-route-policy] apply extcommunity soo 1:100 additive
[PE1-route-policy] quit
```

在 PE 1 上为从 CE 1 学习的路由应用路由策略 soo。

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] peer 10.1.1.1 route-policy soo import
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

在 PE 2 上配置路由策略 soo，为路由添加 SoO 属性 1:100。

```
<PE2> system-view
[PE2] route-policy soo permit node 10
[PE2-route-policy] apply extcommunity soo 1:100 additive
[PE2-route-policy] quit
```

在 PE 2 上为从 CE 2 学习的路由应用路由策略 soo。

```
[PE2] bgp 100
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] peer 10.2.1.1 route-policy soo import
[PE2-bgp-vpn1] quit
[PE2-bgp] quit
```

由于配置的 SoO 属性相同，PE 2 不会将 CE 1 发过来的路由发布给 CE 2。查看 CE 2 路由表，不会再看到 100.1.1.1/32 路由。

```
<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        Vlan2
10.1.1.1/32         BGP    255  0          10.2.1.2        Vlan2
10.2.1.0/24         Direct 0    0          10.2.1.1        Vlan2
10.2.1.1/32         Direct 0    0          127.0.0.1       InLoop0
10.3.1.0/24         BGP    255  0          10.2.1.2        Vlan2
10.3.1.1/32         BGP    255  0          10.2.1.2        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
200.1.1.1/32        BGP    255  0          10.2.1.2        Vlan2
```

2 IPv6 MPLS L3VPN配置

说明

本系列交换机中 S5500-28SC-HI 和 S5500-52SC-HI 不支持 IPv6 MPLS L3VPN 功能。

2.1 IPv6 MPLS L3VPN简介

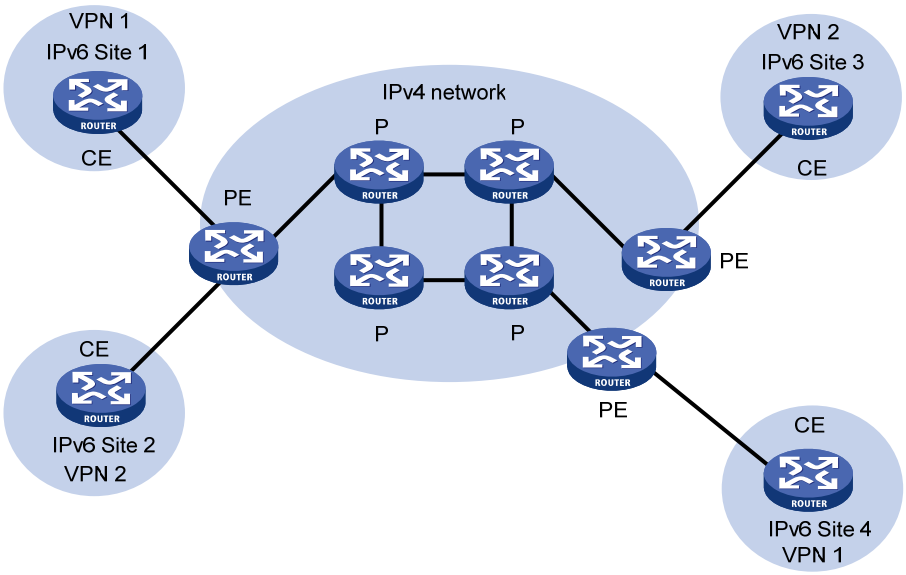
2.1.1 IPv6 MPLS L3VPN概述

MPLS L3VPN 应用于 IPv4 组网环境，利用 BGP 在服务提供商骨干网上发布 VPN 的 IPv4 路由，利用 MPLS 在服务提供商骨干网上转发 VPN 的 IPv4 报文。

IPv6 MPLS L3VPN 的原理与 MPLS L3VPN 相同，所不同的是 IPv6 MPLS L3VPN 利用 BGP 在服务提供商骨干网上发布 VPN 的 IPv6 路由，利用 MPLS 在服务提供商骨干网上转发 VPN 的 IPv6 报文。

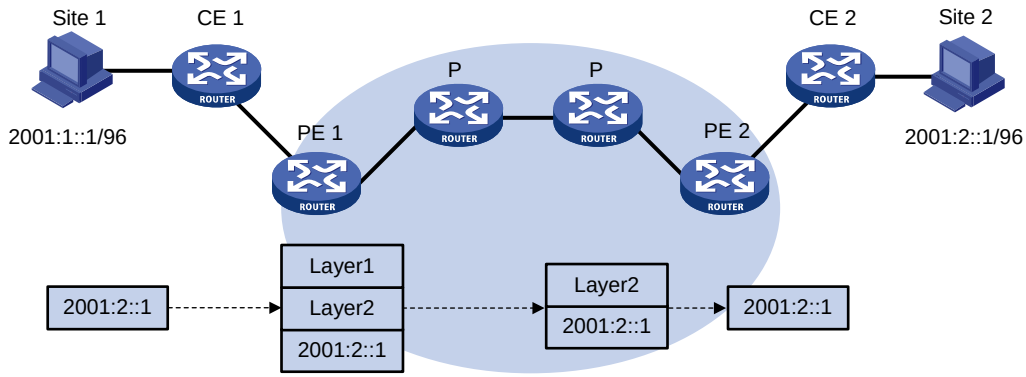
IPv6 MPLS L3VPN的典型组网环境如 图 2-1 所示。目前，IPv6 MPLS L3VPN组网中服务提供商骨干网应为IPv4 网络。VPN内部及CE和PE之间运行IPv6 协议，骨干网中PE和P设备之间运行IPv4 协议。PE需要同时支持IPv4 和IPv6 协议，连接CE的接口上使用IPv6 协议，连接骨干网的接口上使用IPv4 协议。

图2-1 IPv6 MPLS L3VPN 应用组网图



2.1.2 IPv6 MPLS L3VPN的报文转发

图2-2 IPv6 MPLS L3VPN 报文转发示意图



如 图 2-2 所示，IPv6 MPLS L3VPN的报文转发过程为：

(1) Site 1 发出一个目的地址为 2001:2::1 的 IPv6 报文，由 CE 1 将报文发送至 PE 1。

(2) PE 1 根据报文到达的接口及目的地址查找 VPN 实例的路由表项，匹配后将报文转发出去，同时打上内层和外层两个标签。

(3) MPLS 网络利用报文的外层标签，将报文传送到 PE 2。（报文在到达 PE 2 前一跳时已经被剥离外层标签，到达 PE 2 时仅含内层标签）

(4) PE 2 根据内层标签和目的地址查找 VPN 实例的路由表项，确定报文的出接口，将报文转发至 CE 2。

(5) CE 2 根据正常的 IPv6 转发过程将报文传送到目的地。

2.1.3 IPv6 MPLS L3VPN的路由发布

VPN 路由信息的发布过程包括三部分：本地 CE 到入口 PE、入口 PE 到出口 PE、出口 PE 到远端 CE。完成这三部分后，本地 CE 与远端 CE 之间将建立可达路由，VPN 私网路由信息能够在骨干网上发布。

1. 本地CE到入口PE的路由信息交换

CE 与直接相连的 PE 建立邻接关系后，把本站点的 VPN IPv6 路由发布给 PE。

CE 与 PE 之间可以使用 IPv6 静态路由、RIPng、OSPFv3、IPv6 IS-IS、EBGP 路由协议。无论使用哪种路由协议，CE 发布给 PE 的都是标准的 IPv6 路由。

2. 入口PE到出口PE的路由信息交换

PE 从 CE 学到 VPN 的 IPv6 路由信息后，为这些标准 IPv6 路由增加 RD 和 VPN Target 属性，形成 VPN-IPv6 路由，存放到为 CE 创建的 VPN 实例的路由表中，并触发 MPLS 为其分配私网标签。

入口 PE 通过 MP-BGP 把 VPN-IPv6 路由发布给出口 PE。出口 PE 根据 VPN-IPv6 路由的 Export Target 属性与自己维护的 VPN 实例的 Import Target 属性，决定是否将该路由加入到 VPN 实例的路由表。

PE 之间通过 IGP 来保证内部的连通性。

3. 出口PE到远端CE的路由信息交换

与本地 CE 到入口 PE 的路由信息交换相同，远端 CE 有多种方式可以从出口 PE 学习 VPN 路由，包括 IPv6 静态路由、RIPng、OSPFv3、IPv6 IS-IS、EBGP 路由协议。

2.1.4 IPv6 MPLS L3VPN支持的组网方案及功能

目前，IPv6 MPLS L3VPN 支持如下组网方案及功能：

- 基本的 VPN 组网方案
- 跨域 VPN-OptionA
- 跨域 VPN-OptionC
- 运营商的运营商

2.2 IPv6 MPLS L3VPN配置任务简介

表2-1 IPv6 MPLS L3VPN 配置任务简介

配置任务	详细配置	说明
配置基本IPv6 MPLS L3VPN	2.3	完成基本IPv6 MPLS L3VPN配置后，即可实现通过MPLS技术构建简单的IPv6 VPN网络 如果部署一些特殊的IPv6 MPLS L3VPN组网方案，例如跨域VPN，除了配置基本IPv6 MPLS L3VPN外，还需要根据相应章节进行其他的配置
配置IPv6跨域VPN	2.4	

2.3 配置基本IPv6 MPLS L3VPN

2.3.1 基本IPv6 MPLS L3VPN配置任务简介

配置 IPv6 MPLS L3VPN 的关键在于管理 VPN 私网路由在 MPLS 骨干网上的发布，包括 PE-CE 间的路由发布管理以及 PE-PE 间的路由发布管理。

表2-2 基本 IPv6 MPLS L3VPN 配置任务简介

配置任务		说明	详细配置
配置VPN实例	创建VPN实例	必选	2.3.3 1.
	配置VPN实例与接口关联	必选	2.3.3 2.
	配置VPN实例的路由相关属性	可选	2.3.3 3.
	配置VPN实例的隧道策略	可选	2.3.3 4.
	配置LDP实例	可选	2.3.3 5.
配置PE-CE间的路由交换		必选	2.3.4
配置PE-PE间的路由交换		必选	2.3.5
配置BGP-VPNv6子地址族下的路由特性		可选	2.3.6

2.3.2 配置准备

在配置基本 IPv6 MPLS L3VPN 之前，需完成以下任务：

- 对 MPLS 骨干网（PE、P）配置 IGP，实现骨干网的 IP 连通性
- 对 MPLS 骨干网（PE、P）配置 MPLS 基本能力
- 对 MPLS 骨干网（PE、P）配置 MPLS LDP，建立 LDP LSP

2.3.3 配置VPN实例

VPN 实例不仅可以隔离 VPN 私网路由与公网路由，还可以隔离不同 VPN 实例的路由，这一特点使得 VPN 实例的使用不限于 MPLS L3VPN。

配置 VPN 实例的操作是在 PE 或 MCE 设备上进行的。

1. 创建VPN实例

VPN 实例在实现中与 Site 关联。VPN 实例不是直接对应于 VPN，一个 VPN 实例综合了和它所对应 Site 的 VPN 成员关系和路由规则。

表2-3 创建 VPN 实例

操作	命令	说明
进入系统视图	**system-view**	-
创建VPN实例，并进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	必选
配置VPN实例的RD	**route-distinguisher** *route-distinguisher*	必选 一个VPN实例只有配置了RD后才生效
配置VPN实例的描述信息	**description** *text*	可选 描述信息用于描述VPN实例，可以用来记录VPN实例与某个VPN的关系等信息

2. 配置VPN实例与接口关联

VPN 实例配置完成后，还需要与连接 CE 的接口进行关联。支持 MPLS LDP 能力的接口类型都可以与 VPN 实例相关联。具体接口类型请参见“MPLS 配置指导”中的“MPLS 基本配置”。

说明

- 配置三层聚合接口与 VPN 实例关联时，还需要同时配置其所有成员端口与相同的 VPN 实例关联，否则会影响流量的正常转发。
- 执行 **ip binding vpn-instance** 命令将删除接口上已经配置的 IPv6 地址，因此需要重新配置接口的 IPv6 地址。

表2-4 配置 VPN 实例与接口关联

操作	命令	说明
进入系统视图	**system-view**	-
进入要关联接口的接口视图	**interface** *interface-type interface-number*	-
将当前接口与VPN实例关联	**ip binding vpn-instance** *vpn-instance-name*	必选 缺省情况下，接口不关联任何VPN实例

3. 配置VPN实例的路由相关属性

VPN 路由的发布控制过程如下：

- 当从 CE 学习到的一条 VPN 路由引入 BGP 时，BGP 为它关联一个 VPN Target 扩展团体属性列表，通常这个列表是与 CE 相关联的 VPN 实例的输出路由属性列表。
- VPN 实例根据 VPN Target 中 **import-extcommunity** 确定可被接受并引入此 VPN 实例的路由。
- VPN 实例根据 VPN Target 中的 **export-extcommunity** 对向外发布的路由进行 VPN Target 属性的修改。

表2-5 配置 VPN 实例的路由相关属性

操作	命令	说明
进入系统视图	**system-view**	-
进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	-
进入IPv6 VPN视图	**ipv6-family**	可选
配置VPN Target	**vpn-target** *vpn-target*&<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]	必选
配置支持的最大路由数	**routing-table limit** *number* { *warn-threshold* | **simply-alert** }	可选
应用入方向路由策略	**import route-policy** *route-policy*	可选 缺省情况下，接收所有VPN Target属性匹配的路由
应用出方向路由策略	**export route-policy** *route-policy*	可选 缺省情况下，不对发布的路由进行过滤

说明

- VPN 实例视图下配置的路由相关属性既可以用于 IPv4 VPN，也可以用于 IPv6 VPN。
- 既可以在 VPN 实例视图下，也可以在 IPv6 VPN 视图下，配置 IPv6 VPN 的路由相关属性。如果同时在两个视图下配置了路由相关属性，则 IPv6 VPN 采用 IPv6 VPN 视图下配置的路由相关属性。
- 命令 **vpn-target** 最多可以配置 8 个 VPN Target；一个 VPN 实例最多可以配置 64 个 VPN Target。
- 可以配置一个 VPN 实例支持的最大路由数，以防止 PE 路由器的入接口有过多的路由。
- 为 VPN 实例配置路由策略之前必须已经创建了路由策略，否则无法对接收和发布的路由进行过滤。

4. 配置VPN实例的隧道策略

在网络中存在多条、多种类型的隧道时，可以通过配置隧道策略对选择的隧道种类和隧道数目进行限制。

通过 **tunnel select-seq** 命令，可以配置隧道的优先级顺序和负载分担条数。

通过 **preferred-path** 命令可以配置优选路径，每条优选路径对应一个隧道接口，指定通过哪个隧道接口转发流量。

应用隧道策略后，隧道的选择方式为：

(1) 从编号最小的优选路径开始匹配，如果对端 PE 地址和任何优选路径的隧道目的地址都不同，则根据 **tunnel select-seq** 命令配置的隧道优先级顺序和负载分担条数选择隧道。如果没有配置 **tunnel select-seq** 命令，则根据缺省的隧道策略选择隧道，即按照 LSP 隧道－>CR-LSP 隧道的优先级顺序选择隧道，负载分担条数为 1。

(2) 如果对端 PE 地址和某条优选路径的隧道目的地址相同，则匹配成功。此优选路径指定的隧道可用时，不再匹配后续的隧道策略规则，使用该优选路径指定的隧道接口转发流量。

(3) 匹配的优选路径指定的隧道不可用时（例如，隧道接口 down、隧道下配置的 ACL 规则不允许转发该流量等原因导致隧道不可用），如果该优选路径中没有指定 **disable-fallback** 参数，则继续匹配后续的隧道策略规则；如果该优选路径中指定了 **disable-fallback** 参数，则不再匹配后续的隧道策略规则，隧道查找失败。

表2-6 配置 VPN 实例的隧道策略

操作	命令	说明
进入系统视图	**system-view**	-
创建隧道策略，并进入隧道策略视图	**tunnel-policy** *tunnel-policy-name*	必选
配置通过隧道接口转发报文的优选路径	**preferred-path** *number* **interface tunnel** *tunnel-number* [**disable-fallback**]	可选 缺省情况下，不存在任何优选路径

操作	命令	说明
配置隧道的优先级顺序和负载分担条数	**tunnel select-seq** { **cr-lsp** | **lsp** } * **load-balance-number** *number*	可选 缺省情况下，按照LSP隧道->CR-LSP隧道的优先级顺序选择隧道，负载分担条数为1
退回系统视图	**quit**	-
进入VPN实例视图	**ip vpn-instance** *vpn-instance-name*	必选
进入IPv6 VPN视图	**ipv6-family**	可选
配置VPN实例的隧道策略	**tnl-policy** *tunnel-policy-name*	必选 缺省情况下，按照LSP隧道->CR-LSP隧道的优先级顺序选择隧道，负载分担条数为1

说明：

- 配置的隧道类型不止一种，且负载分担的条数多于单个类型包含的隧道个数时，会使用不同类型的隧道一起进行负载分担。
- 一个隧道策略下最多可以配置 64 条优选路径。不同优选路径对应的隧道接口的目的地址可以相同。隧道接口的隧道类型必须为 MPLS TE。
- 在配置隧道优先级顺序时，隧道类型越靠近关键字 **select-seq** 的，其优先级越高。例如：如果配置了命令 **tunnel select-seq lsp cr-lsp load-balance-number** 1，在没有 LSP 的情况下，VPN 将使用 CR-LSP 隧道，当 LSP 建立后，VPN 将选用 LSP。
- VPN 实例视图下配置的隧道策略既可以用于 IPv4 VPN，也可以用于 IPv6 VPN。
- 既可以在 VPN 实例视图下，也可以在 IPv6 VPN 视图下，配置 IPv6 VPN 的隧道策略。如果同时在两个视图下配置了隧道策略，则 IPv6 VPN 采用 IPv6 VPN 视图下配置的隧道策略。
- 为 VPN 实例配置隧道策略之前必须先创建隧道策略，否则将采用缺省策略，即按照 LSP 隧道->CR-LSP 隧道的优先级顺序选择隧道，负载分担条数为 1。

5. 配置LDP实例

LDP 实例应用于运营商的运营商（Carriers' carriers）组网模型中。

本配置任务用来使能 VPN 实例的 LDP 能力，创建 LDP 实例，并在 MPLS-LDP-VPN 实例视图下为 VPN 实例配置 LDP 参数。

LDP实例的配置方法，请参见“1.3.3 5. 配置LDP实例”。

2.3.4 配置PE-CE间的路由交换

PE-CE 间路由交换可以使用 IPv6 静态路由、RIPng、OSPFv3、IPv6 IS-IS 和 EBGP 路由协议。根据实际组网情况选择一种进行配置即可。

1. 配置准备

在配置 PE-CE 间路由交换之前，需要先完成以下任务：

- 在 CE 上配置连接 PE 接口的 IPv6 地址
- 在 PE 上配置连接 CE 接口的 IPv6 地址

2. 配置PE-CE间使用IPv6 静态路由

表2-7 配置 PE-CE 间使用 IPv6 静态路由

操作	命令	说明
进入系统视图	**system-view**	-
为指定VPN实例配置IPv6静态路由	**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*]	二者必选其一 该配置在PE上进行，CE上的配置方法与普通IPv6静态路由相同
	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*]	

说明

有关 IPv6 静态路由的配置请参见“三层技术-IP 路由配置指导”中的“IPv6 静态路由”。

3. 配置PE-CE间使用RIPng

一个 RIPng 进程只能属于一个 VPN 实例。如果在启动 RIPng 进程时不绑定到 VPN 实例，则该进程属于公网进程。

表2-8 配置 PE-CE 间使用 RIPng

操作	命令	说明
进入系统视图	**system-view**	-
创建PE-CE间的RIPng实例，并进入RIPng视图	**ripng** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在PE上进行，CE上配置普通RIPng即可
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
在指定的网络接口上使能RIPng	**ripng** *process-id* **enable**	必选 缺省情况下，接口禁用RIPng

说明

有关 RIPng 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“RIPng”。

4. 配置PE-CE间使用OSPFv3

一个 OSPFv3 进程只能属于一个 VPN 实例。如果在启动 OSPFv3 进程时不绑定到 VPN 实例，则该进程属于公网进程。

表2-9 配置 PE-CE 间使用 OSPFv3

操作	命令	说明
进入系统视图	**system-view**	-
创建PE-CE间的OSPFv3实例，并进入OSPFv3视图	**ospfv3** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在PE上进行，CE上配置普通OSPFv3即可
配置Router ID	**router-id** *router-id*	必选
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
在接口上使能OSPFv3	**ospfv3** *process-id* **area** *area-id* [**instance** *instance-id*]	必选 缺省情况下，接口上没有使能OSPFv3 该配置在PE上进行

说明

有关 OSPFv3 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“OSPFv3”。

5. 配置PE-CE间使用IPv6 IS-IS

一个 IPv6 IS-IS 进程只能属于一个 VPN 实例。如果在启动 IPv6 IS-IS 进程时不绑定到 VPN 实例，则该进程属于公网进程。

表2-10 配置 PE-CE 间使用 IPv6 IS-IS

操作	命令	说明
进入系统视图	**system-view**	-
创建PE-CE间的IPv6 IS-IS实例，并进入IS-IS视图	**isis** [*process-id*] **vpn-instance** *vpn-instance-name*	必选 该配置在PE上进行，CE上配置普通IPv6 IS-IS即可

操作	命令	说明
配置网络实体名称	**network-entity** *net*	必选 缺省情况下，没有配置网络实体名称
使能IS-IS进程的IPv6能力	**ipv6 enable**	必选 缺省情况下，没有使能IS-IS路由进程的IPv6能力
退回系统视图	**quit**	-
进入接口视图	**interface** *interface-type interface-number*	-
使能接口IS-IS路由进程的IPv6能力，并指定要关联的IS-IS进程号	**isis ipv6 enable** [*process-id*]	必选 缺省情况下，接口上没有使能IS-IS路由进程的IPv6能力

说明

有关 IPv6 IS-IS 的介绍和详细配置，请参见“三层技术-IP 路由配置指导”中的“IPv6 IS-IS”。

6. 配置PE-CE间使用EBGP

(1) PE 上的配置

表2-11 PE 上的配置

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入IPv6 BGP-VPN实例视图	**ipv6-family vpn-instance** *vpn-instance-name*	必选
将CE配置为VPN私网EBGP对等体	**peer** *ipv6-address* **as-number** *as-number*	必选
引入本端CE路由	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	必选 PE需要将到本端CE的路由引入VPN路由表中，以发布给对端PE
配置对发布的路由信息进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **export** [**direct** | **isisv6** *process-id* | **ripng** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤

(2) CE 上的配置

表2-12 CE 上的配置

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
进入IPv6 BGP子地址族视图	**ipv6-family**	必选
将PE配置为EBGP对等体	**peer** *ipv6-address* **as-number** *as-number*	必选
配置路由引入	**import-route** *protocol* [*process-id*] [**med** *med-value* | **route-policy** *route-policy-name*] *	可选 CE需要将自己所能到达的VPN网段地址发布给接入的PE，通过PE发布给对端CE

说明

- 配置 IPv6 BGP-VPN 实例后，VPN 实例中的 BGP 路由交换与普通的 BGP 相同。
- IPv6 BGP-VPN 实例视图下的配置命令与 IPv6 BGP 子地址族视图下的配置命令相同。上文只列举了部分配置命令，更多的配置命令请参见“三层技术-IP 路由配置指导”中的“IPv6 BGP”。

2.3.5 配置PE-PE间的路由交换

表2-13 配置 PE-PE 间的路由交换

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	必选
将对端PE配置为对等体	**peer** *ip-address* **as-number** *as-number*	必选
指定路由更新报文的源接口	**peer** { *group-name* | *ip-address* } **connect-interface** *interface-type interface-number*	必选 缺省情况下，BGP使用最佳路由更新报文的源接口
进入BGP-VPNv6子地址族视图	**ipv6-family vpnv6**	必选
使能对等体交换BGP-VPNv6路由信息	**peer** *ip-address* **enable**	必选 缺省情况下，BGP对等体之间只能交换IPv4路由信息

2.3.6 配置BGP-VPNv6 子地址族下的路由特性

BGP-VPNv6 子地址族下的很多路由特性都与 BGP IPv6 单播路由的特性相同。这些特性根据组网的需求来决定是否选择。

表2-14 配置 BGP-VPNv6 子地址族的路由特性

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
将对端PE配置为对等体	**peer** *ip-address* **as-number** *as-number*	必选
指定建立TCP连接的接口	**peer** *ip-address* **connect-interface** *interface-type interface-number*	必选
进入BGP-VPNv6子地址族视图	**ipv6-family vpnv6**	-
配置本地优先级的缺省值	**default local-preference** *value*	可选 缺省情况下，本地优先级的值为100
配置系统MED的缺省值	**default med** *med-value*	可选 缺省情况下，系统MED的缺省值为0
配置对发布的路由信息进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **export** [**direct** | **isisv6** *process-id* | **ripng** *process-id* | **static**]	可选 缺省情况下，不对发布的路由信息进行过滤
配置对接收的路由信息进行过滤	**filter-policy** { *acl6-number* | **ipv6-prefix** *ipv6-prefix-name* } **import**	可选 缺省情况下，不对接收的路由信息进行过滤
配置为指定对等体应用过滤策略	**peer** *ip-address* **filter-policy** *acl6-number* { **export** | **import** }	可选 缺省情况下，没有为对等体应用过滤策略
为指定对等体应用基于IPv6地址前缀列表的路由过滤策略	**peer** *ip-address* **ipv6-prefix** *prefix-name* { **export** | **import** }	可选 缺省情况下，对等体上不应用基于IPv6地址前缀列表的路由过滤策略
为从对等体接收的路由分配首选值	**peer** *ip-address* **preferred-value** *value*	可选 缺省情况下，从对等体接收的路由的首选值为0
配置发送BGP更新报文时不携带私有自治系统号	**peer** *ip-address* **public-as-only**	可选 缺省情况下，发送BGP更新报文时，携带私有自治系统号
为指定对等体配置路由策略	**peer** *ip-address* **route-policy** *route-policy-name* { **export** | **import** }	可选 缺省情况下，对等体没有应用路由策略
对接收到的BGP-VPNv6子地址族的BGP路由使能VPN-Target过滤功能	**policy vpn-target**	可选 缺省情况下，对接收到的BGP-VPNv6子地址族的BGP路由使能VPN-Target过滤功能

操作	命令	说明
配置将本机作为路由反射器，并将对等体作为路由反射器的客户	**peer** *ip-address* **reflect-client**	可选 缺省情况下，没有配置路由反射器及其客户
允许客户到客户的路由反射	**reflect between-clients**	可选 缺省情况下，允许客户到客户的路由反射
为路由反射器配置群ID	**reflector cluster-id** { *cluster-id* | *ip-address* }	可选 缺省情况下，每个路由反射器都使用自己的Router ID作为群ID 如果一个群中配置了多个路由反射器，请使用本命令为所有的路由反射器配置相同的群ID，以避免产生路由环路
创建路由反射器的反射策略	**rr-filter** *extended-community-list-number*	可选 缺省情况下，路由反射器不会对反射的路由进行过滤 执行本命令后，只有与本命令配置的扩展团体属性号匹配的IBGP路由才会被反射 通过在不同的路由反射器上配置不同的反射策略，可以实现路由反射器之间的负载分担

说明

有关 IPv6 BGP 的路由特性请参见“三层技术-IP 路由配置指导”中的“IPv6 BGP”。

2.4 配置IPv6跨域VPN

如果承载 IPv6 VPN 路由的 MPLS 骨干网跨越多个 AS，就需要配置跨域 VPN。

跨域VPN有三种解决方案，具体介绍请参见“1.1.6 跨域VPN”。目前，IPv6 MPLS L3VPN只支持VPN-OptionA和VPN-OptionC。

2.4.1 配置准备

在配置跨域 VPN 之前，需完成以下任务：

- 为各 AS 的 MPLS 骨干网分别配置 IGP，实现同一 AS 内骨干网的 IP 连通性
- 为各 AS 的 MPLS 骨干网分别配置 MPLS 基本能力
- 为各 AS 的 MPLS 骨干网分别配置 MPLS LDP，建立 LDP LSP

说明

请根据实际应用的 IPv6 跨域 VPN 方式，选择下述配置任务中的对应项。

2.4.2 配置IPv6 跨域VPN-OptionA

IPv6 跨域 VPN-OptionA 的实现比较简单，当 PE 上的 VPN 数量及 VPN 路由数量都比较少时可以采用这种方案。

IPv6 跨域 VPN-OptionA 的配置可以描述为：

- 对各 AS 分别进行基本 IPv6 MPLS L3VPN 配置。
- 对于 ASBR-PE，将对端 ASBR-PE 看作自己的 CE 配置即可。即：IPv6 跨域 VPN-OptionA 方式需要在 PE 和 ASBR-PE 上分别配置 IPv6 VPN 实例，前者用于接入 CE，后者用于接入对端 ASBR-PE。

具体配置方法，请参见“2.3 配置基本IPv6 MPLS L3VPN”。

说明

在 IPv6 跨域 VPN-OptionA 方式中，对于同一个 IPv6 VPN，同一 AS 内的 ASBR-PE 和 PE 上配置的 VPN Target 应能匹配，即 VPN Target 的配置应能保证 PE（或 ASBR-PE）发送的 VPN 路由能够被 ASBR-PE（或 PE）接受；不同 AS 的 PE 上配置的 VPN Target 则不需要匹配。

2.4.3 配置IPv6 跨域VPN-OptionC

1. 配置PE

同一AS内的PE和ASBR-PE之间建立普通的IBGP对等体关系，不同AS的PE之间建立MP-EBGP对等体关系。

同一 AS 内的 PE 和 ASBR-PE 之间应能够交换带标签的路由。

表2-15 配置 PE

操作	命令	说明
进入系统视图	**system-view**	-
进入BGP视图	**bgp** *as-number*	-
配置本AS的ASBR-PE为IBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选
配置与本AS的ASBR-PE之间能够交换带标签的路由	**peer** { *group-name* | *ip-address* } **label-route-capability**	必选 缺省情况下，不向IPv4对等体/对等体组发送标签路由
将另一AS的PE配置为EBGP对等体	**peer** { *group-name* | *ip-address* } **as-number** *as-number*	必选

操作	命令	说明
进入BGP-VPNv6子地址族视图	**ipv6-family vpnv6**	-
使能EBGP对等体交换BGP-VPNv6路由信息	**peer** *ip-address* **enable**	必选

2. 配置ASBR-PE

在跨域VPN-OptionC方式中，需要建立一条跨域的VPN LSP，相关PE、ASBR之间发布公网路由时携带MPLS标签信息。此跨域VPN LSP的建立同IPv4 MPLS VPN相同，配置方式参见“1.4.4 配置跨域VPN-OptionC”。

3. 配置路由策略

配置在 ASBR-PE 上应用路由策略：

- 对于从本 AS 的 PE 接收的路由，在向对端 ASBR-PE 发布时，分配 MPLS 标签；
- 对于向本 AS 的 PE 发布的路由，如果是带标签的路由，为其分配新的 MPLS 标签。

IPv6 跨域VPN路由策略的配置与IPv4 MPLS VPN相同，配置方式参见“1.4.4 配置跨域VPN-OptionC”。

2.5 IPv6 MPLS L3VPN显示和维护

2.5.1 复位BGP连接

说明

软复位 BGP 连接是指在不断开 BGP 邻居关系的情况下，更新 BGP 路由信息；复位 BGP 连接是指断开并重新建立 BGP 邻居关系的情况下，更新 BGP 路由信息。

当 BGP 配置变化后，可以通过软复位或复位 BGP 连接使新的配置生效。软复位需要 BGP 对等体具备路由刷新能力（支持 ROUTE-REFRESH 消息）。

请在用户视图下进行下列操作。

表2-16 复位 BGP 连接

操作	命令
软复位指定VPN实例的IPv6 BGP连接	**refresh bgp ipv6 vpn-instance** *vpn-instance-name* { *ipv6-address* | **all** | **external** } { **export** | **import** }
软复位BGP的VPNv6连接	**refresh bgp vpnv6** { *ip-address* | **all** | **external** | **internal** } { **export** | **import** }
复位指定VPN实例的IPv6 BGP连接	**reset bgp ipv6 vpn-instance** *vpn-instance-name* { *as-number* | *ipv6-address* | **all** | **external** }
复位BGP的VPNv6连接	**reset bgp vpnv6** { *as-number* | *ip-address* | **all** | **external** | **internal** }

2.5.2 显示IPv6 MPLS L3VPN的运行状态

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 IPv6 MPLS L3VPN 的运行情况，通过查看显示信息验证配置的效果。

表2-17 显示 IPv6 MPLS L3VPN 的运行状态

操作	命令
显示与VPN实例相关联的IPv6路由表	**display ipv6 routing-table vpn-instance** *vpn-instance-name* [**verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例信息	**display ip vpn-instance** [**instance-name** *vpn-instance-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例的IPv6 FIB信息	**display ipv6 fib vpn-instance** *vpn-instance-name* [**acl6** *acl6-number* | **ipv6-prefix** *ipv6-prefix-name*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例中与指定目的IPv6地址匹配的IPv6 FIB信息	**display ipv6 fib vpn-instance** *vpn-instance-name ipv6-address* [*prefix-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示PE与PE之间建立的BGP VPNv6对等体的信息	**display bgp vpnv6 all peer** [*ipv4-address* **verbose** | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例中PE与CE之间建立的IPv6 BGP对等体的信息	**display bgp vpnv6 vpn-instance** *vpn-instance-name* **peer** [*ipv6-address* **verbose** | **verbose**] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示所有BGP VPNv6路由信息	**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*]
显示指定路由标识符的BGP VPNv6路由信息	**display bgp vpnv6 route-distinguisher** *route-distinguisher* **routing-table** [*network-address prefix-length*] [**|** { **begin** | **exclude** | **include** } *regular-expression*]
显示指定VPN实例的BGP VPNv6路由信息	**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*]

说明

有关查看路由表的命令介绍请参见“三层技术-IP 路由命令参考”中的“IP 路由基础命令”。

2.6 IPv6 MPLS L3VPN典型配置举例

2.6.1 配置IPv6 MPLS L3VPN示例

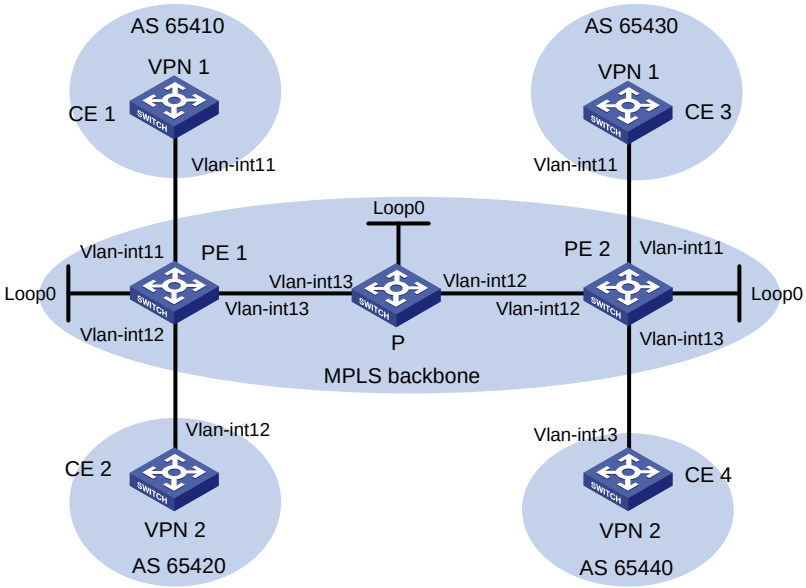
1. 组网需求

- CE 1、CE 3 属于 VPN 1，CE 2、CE 4 属于 VPN 2；
- VPN 1 使用的 VPN Target 属性为 111:1，VPN 2 使用的 VPN Target 属性为 222:2。不同 VPN 用户之间不能互相访问；

- CE 与 PE 之间配置 EBGP 交换 VPN 路由信息；
- PE 与 PE 之间配置 OSPF 实现 PE 内部的互通、配置 MP-IBGP 交换 VPN 路由信息。

2. 组网图

图2-3 配置 IPv6 MPLS L3VPN 组网图



设备	接口	IP地址	设备	接口	IP地址
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			

3. 配置步骤

(1) 在 MPLS 骨干网上配置 IGP 协议，实现骨干网 PE 和 P 的互通

配置 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
```

配置 P。

```
<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
```

配置 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
```

配置完成后，PE 1、P、PE 2 之间应能建立 OSPF 邻居，执行 **display ospf peer** 命令可以看到邻居达到 FULL 状态。执行 **display ip routing-table** 命令可以看到 PE 之间学习到对方的 Loopback 路由。

以 PE 1 为例：

```
[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) 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 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
```

配置 P。

```
[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
```

配置 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
```

上述配置完成后，PE 1、P、PE 2 之间应能建立 LDP 会话，执行 **display mpls ldp session** 命令可以看到显示结果中 Status 字段为 Operational。执行 **display mpls ldp lsp** 命令，可以看到 LDP LSP 的建立情况。

以 PE 1 为例：

```
[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) 在 PE 设备上配置 VPN 实例，将 CE 接入 PE

配置 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::2 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::2 64
[PE1-Vlan-interface12] quit
```

配置 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::2 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::2 64
[PE2-Vlan-interface13] quit
```

按 图 2-3 配置各CE的接口IP地址，配置过程略。

配置完成后，在 PE 设备上执行 **display ip vpn-instance** 命令可以看到 VPN 实例的配置情况。各 PE 能 ping 通自己接入的 CE。

以 PE 1 和 CE 1 为例：

```
[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) 在 PE 与 CE 之间建立 EBGP 对等体，引入 VPN 路由

配置 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
```

说明

另外 3 个 CE 设备（CE 2～CE 4）配置与 CE 1 设备配置类似，配置过程省略。

配置 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
```

说明

PE 2 的配置与 PE 1 类似，配置过程省略。

配置完成后，在 PE 设备上执行 **display bgp vpnv6 vpn-instance peer** 命令，可以看到 PE 与 CE 之间的 BGP 对等体关系已建立，并达到 Established 状态。

以 PE 1 与 CE 1 的对等体关系为例：

```
[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) 在 PE 之间建立 MP-IBGP 对等体

配置 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
```

配置 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
```

配置完成后，在 PE 设备上执行 **display bgp peer** 或 **display bgp vpnv6 all peer** 命令，可以看到 PE 之间的 BGP 对等体关系已建立，并达到 Established 状态。

```
[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) 配置完成后的检验

在 PE 设备上执行 **display ipv6 routing-table vpn-instance** 命令，可以看到去往对端 CE 的路由。

以 PE 1 为例：

```
[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
```

同一 VPN 的 CE 能够相互 Ping 通，不同 VPN 的 CE 不能相互 Ping 通。

例如：CE 1 能够 Ping 通 CE 3（2001:3::1），但不能 Ping 通 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
```

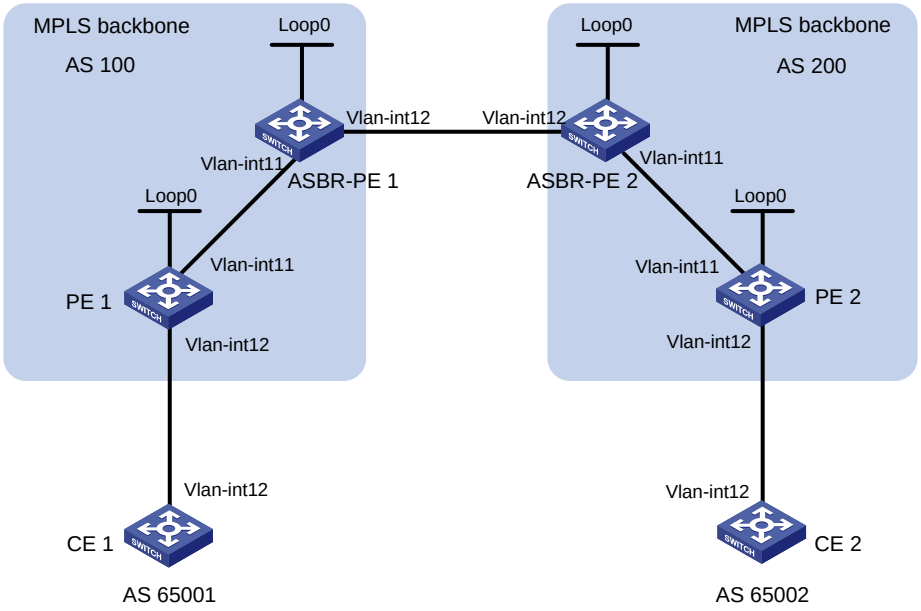
2.6.2 配置IPv6 跨域VPN-OptionA方式示例

1. 组网需求

- CE 1 和 CE 2 属于同一个 VPN。
- CE 1 通过 AS100 的 PE 1 接入，CE 2 通过 AS200 的 PE 2 接入。
- 采用 OptionA 方式实现跨域的 IPv6 MPLS L3VPN，即采用 VRF-to-VRF 方式管理 VPN 路由。
- 同一个 AS 内部的 MPLS 骨干网使用 OSPF 作为 IGP。

2. 组网图

图2-4 配置跨域 VPN-OptionA 方式组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 在 MPLS 骨干网上配置 IGP 协议，实现骨干网内互通

本例中采用 OSPF，具体配置步骤略。

说明

需要将作为 LSR ID 的 LoopBack 接口的 32 位地址通过 OSPF 发布出去。

配置完成后，ASBR-PE 与本 AS 的 PE 之间应能建立 OSPF 邻居，执行 **display ospf peer** 命令可以看到邻居达到 FULL 状态，ASBR-PE 与本 AS 的 PE 之间能学习到对方的 Loopback 地址。ASBR-PE 与本 AS 的 PE 之间能够互相 ping 通。

(2) 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 PE 1 的 MPLS 基本能力，并在与 ASBR-PE 1 相连的接口上使能 LDP。

```
<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
```

配置 ASBR-PE 1 的 MPLS 基本能力，并在与 PE 1 相连的接口上使能 LDP。

```
<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
```

配置 ASBR-PE 2 的 MPLS 基本能力，并在与 PE 2 相连的接口上使能 LDP。

```
<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
```

配置 PE 2 的 MPLS 基本能力，并在与 ASBR-PE 2 相连的接口上使能 LDP。

```
<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
```

上述配置完成后，同一 AS 的 PE 和 ASBR-PE 之间应该建立起 LDP 邻居，在各设备上执行 **display mpls ldp session** 命令可以看到显示结果中 Status 字段为“Operational”。

(3) 在 PE 设备上配置 VPN 实例，将 CE 接入 PE

说明

同一 AS 内的 ASBR-PE 与 PE 的 VPN 实例的 VPN Target 应能匹配，不同 AS 的 PE 的 VPN 实例的 VPN Target 则不需要匹配。

配置 CE 1。

```
<CE1> system-view
[CE1] interface vlan-interface 12
[CE1-Vlan-interface12] ipv6 address 2001:1::1 64
[CE1-Vlan-interface12] quit
```

配置 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
```

配置 CE 2。

```
<CE2> system-view
[CE2] interface vlan-interface 12
[CE2-Vlan-interface12] ipv6 address 2001:2::1 64
[CE2-Vlan-interface12] quit
```

配置 PE 2。

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance] route-distinguisher 200:1
[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
```

配置 ASBR-PE 1：创建 VPN 实例，并将此实例绑定到连接 ASBR-PE 2 的接口（ASBR-PE 1 认为 ASBR-PE 2 是自己的 CE）。

```
[ASBR-PE1] ip vpn-instance vpn1
[ASBR-PE1-vpn-instance-vpn1] route-distinguisher 100:2
[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] ipv6 address 2002:1::1 64
[ASBR-PE1-Vlan-interface12] quit
```

配置 ASBR-PE 2：创建 VPN 实例，并将此实例绑定到连接 ASBR-PE 1 的接口（ASBR-PE 2 认为 ASBR-PE 1 是自己的 CE）。

```
[ASBR-PE2] ip vpn-instance vpn1
[ASBR-PE2-vpn-vpn-vpn1] route-distinguisher 200:2
[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] ipv6 address 2002:1::2 64
```

```
[ASBR-PE2-Vlan-interface12] quit
```

上述配置完成后，在各 PE 设备上执行 **display ip vpn-instance** 命令能正确显示 VPN 实例配置。各 PE 能 ping 通 CE。ASBR-PE 之间也能互相 ping 通。

(4) 在 PE 与 CE 之间建立 EBGP 对等体，引入 VPN 路由

配置 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
```

配置 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
```

配置 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
```

配置 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) PE 与本 AS 的 ASBR-PE 之间建立 IBGP 对等体，ASBR-PE 之间建立 EBGP 对等体

配置 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
[PE1-bgp] quit
```

配置 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
```

配置 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
```

配置 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) 配置完成后的检验

上述配置完成后，CE 之间能学习到对方的接口路由，CE 1 和 CE 2 能够相互 ping 通。

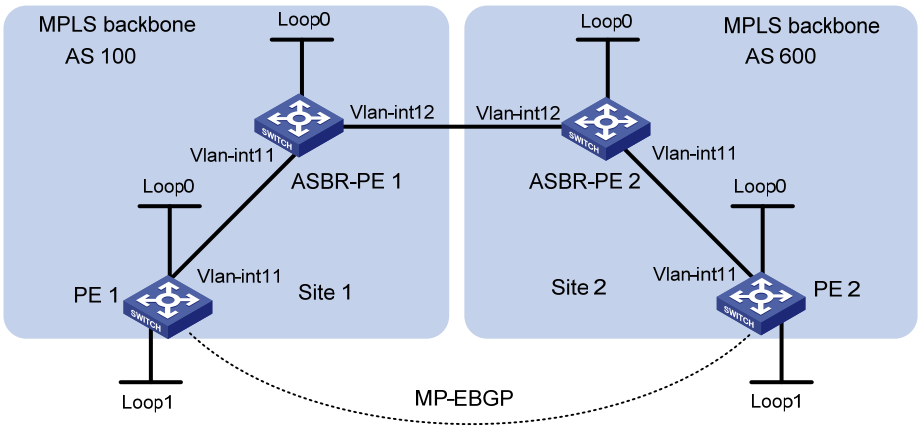
2.6.3 配置IPv6 跨域VPN-OptionC方式示例

1. 组网需求

- Site 1 和 Site 2 属于同一个 VPN，Site 1 通过 AS 100 的 PE 1 接入，Site 2 通过 AS 600 的 PE 2 接入；
- 同一自治系统内的 PE 设备之间运行 IS-IS 作为 IGP；
- PE 1 与 ASBR-PE 1 间通过 MP-IBGP 交换带标签的 IPv4 路由；
- PE 2 与 ASBR-PE 2 间通过 MP-IBGP 交换带标签的 IPv4 路由；
- PE 1 与 PE 2 建立 MP-EBGP 对等体；
- ASBR-PE 1 和 ASBR-PE 2 上分别配置路由策略，对从对方接收的路由压入标签；
- ASBR-PE 1 与 ASBR-PE 2 间通过 MP-EBGP 交换带标签的 IPv4 路由。

2. 组网图

图2-5 IPv6 跨域 VPN-OptionC 方式组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置 PE 1

在 PE 1 上运行 IS-IS。

```
<PE1> system-view
[PE1] isis 1
[PE1-isis-1] network-entity 10.111.111.111.111.00
[PE1-isis-1] quit
```

配置 LSR ID，使能 MPLS 和 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
```

配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```

创建 Loopback0 接口，在接口上运行 IS-IS。

```
[PE1] interface loopback 0
[PE1-LoopBack0] ip address 2.2.2.9 32
```

```
[PE1-LoopBack0] isis enable 1
[PE1-LoopBack0] quit
```
创建 VPN 实例，名称为 vpn1，为其配置 RD 和 VPN Target 属性。
```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 11:11
[PE1-vpn-instance-vpn1] vpn-target 3:3 import-extcommunity
[PE1-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE1-vpn-instance-vpn1] quit
```
创建 Loopback1 接口，并将此接口绑定到 vpn1。
```
[PE1] interface loopback 1
[PE1-LoopBack1] ip binding vpn-instance vpn1
[PE1-LoopBack1] ipv6 address 2001:1::1 128
[PE1-LoopBack1] quit
```
在 PE 1 上运行 BGP。
```
[PE1] bgp 100
```
配置 PE 1 向 IBGP 对等体 3.3.3.9 发布标签路由及从 3.3.3.9 接收标签路由的能力。
```
[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
```
配置 PE 1 到 EBGP 对等体 5.5.5.9 的最大跳数为 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
```
配置对等体 5.5.5.9 作为 VPNv6 对等体。
```
[PE1-bgp] ipv6-family vpnv6
[PE1-bgp-af-vpnv6] peer 5.5.5.9 enable
[PE1-bgp-af-vpnv6] quit
```
将直连路由引入 vpn1 的路由表。
```
[PE1-bgp] ipv6-family vpn-instance vpn1
[PE1-bgp-ipv6-vpn1] import-route direct
[PE1-bgp-ipv6-vpn1] quit
[PE1-bgp] quit
```
(2)　配置 ASBR-PE1

在 ASBR-PE 1 上运行 IS-IS。
```
<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
```
配置 LSR ID，使能 MPLS 和 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
```

配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```

配置接口 Vlan-interface12，并在接口上使能 MPLS。

```
[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
```

创建 Loopback0 接口，在接口上运行 IS-IS。

```
[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
```

创建路由策略。

```
[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
```

在 ASBR-PE 1 上运行 BGP，引入 IS-IS 进程 1 的路由。

```
[ASBR-PE1] bgp 100
[ASBR-PE1-bgp] import-route isis 1
```

对向 IBGP 对等体 2.2.2.9 发布的路由应用已配置的路由策略 policy2。

```
[ASBR-PE1-bgp] peer 2.2.2.9 as-number 100
[ASBR-PE1-bgp] peer 2.2.2.9 route-policy policy2 export
```

向 IBGP 对等体 2.2.2.9 发布标签路由及从 2.2.2.9 接收标签路由的能力。

```
[ASBR-PE1-bgp] peer 2.2.2.9 connect-interface loopback 0
[ASBR-PE1-bgp] peer 2.2.2.9 label-route-capability
```

对向 EBGP 对等体 11.0.0.1 发布的路由应用已配置的路由策略 policy1。

```
[ASBR-PE1-bgp] peer 11.0.0.1 as-number 600
[ASBR-PE1-bgp] peer 11.0.0.1 route-policy policy1 export
```

向 EBGP 对等体 11.0.0.1 发布标签路由及从 11.0.0.1 接收标签路由的能力。

```
[ASBR-PE1-bgp] peer 11.0.0.1 label-route-capability
[ASBR-PE1-bgp] quit
```

(3)　配置 ASBR-PE 2

在 ASBR-PE 2 上运行 IS-IS。

```
<ASBR-PE2> system-view
[ASBR-PE2] isis 1
[ASBR-PE2-isis-1] network-entity 10.333.333.333.333.00
```

```
[ASBR-PE2-isis-1] quit
```
配置 LSR ID，使能 MPLS 和 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
```
配置接口 Vlan-interface11，在接口上运行 IS-IS，并在接口上使能 MPLS 和 LDP。
```
[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
```
创建 Loopback0 接口，在接口上运行 IS-IS。
```
[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
```
配置接口 Vlan-interface12，在接口上使能 MPLS。
```
[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
```
创建路由策略。
```
[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
```
在 ASBR-PE 2 上运行 BGP，引入 IS-IS 进程 1 的路由。
```
[ASBR-PE2] bgp 600
[ASBR-PE2-bgp] import-route isis 1
```
向 IBGP 对等体 5.5.5.9 发布标签路由及从 5.5.5.9 接收标签路由的能力。
```
[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
```
对向 IBGP 对等体 5.5.5.9 发布的路由应用已配置的路由策略 policy2。
```
[ASBR-PE2-bgp] peer 5.5.5.9 route-policy policy2 export
```
对向 EBGP 对等体 11.0.0.2 发布的路由应用已配置的路由策略 policy1。
```
[ASBR-PE2-bgp] peer 11.0.0.2 as-number 100
[ASBR-PE2-bgp] peer 11.0.0.2 route-policy policy1 export
```
向 EBGP 对等体 11.0.0.2 发布标签路由及从 11.0.0.2 接收标签路由的能力。

```
[ASBR-PE2-bgp] peer 11.0.0.2 label-route-capability
[ASBR-PE2-bgp] quit
```

(4) 配置 PE 2

在 PE 2 上运行 IS-IS。

```
<PE2> system-view
[PE2] isis 1
[PE2-isis-1] network-entity 10.444.444.444.444.00
[PE2-isis-1] quit
```

配置 LSR ID，使能 MPLS 和 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
```

配置接口 Vlan-interface11，在接口上运行 IS-IS，并使能 MPLS 和 LDP。

```
[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
```

创建 Loopback0 接口，在接口上运行 IS-IS。

```
[PE2] interface loopback 0
[PE2-LoopBack0] ip address 5.5.5.9 32
[PE2-LoopBack0] isis enable 1
[PE2-LoopBack0] quit
```

创建 VPN 实例，名称为 vpn1，为其配置 RD 和 VPN Target 属性。

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 11:11
[PE2-vpn-instance-vpn1] vpn-target 3:3 import-extcommunity
[PE2-vpn-instance-vpn1] vpn-target 3:3 export-extcommunity
[PE2-vpn-instance-vpn1] quit
```

创建 Loopback1 接口，并将此接口绑定到 vpn1。

```
[PE2] interface loopback 1
[PE2-LoopBack1] ip binding vpn-instance vpn1
[PE2-LoopBack1] ipv6 address 2001:1::2 128
[PE2-LoopBack1] quit
```

在 PE 2 上运行 BGP。

```
[PE2] bgp 600
```

配置 PE 2 向 IBGP 对等体 4.4.4.9 发布标签路由及从 4.4.4.9 接收标签路由的能力。

```
[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
```

配置 PE 2 到 EBGP 对等体 2.2.2.9 的最大跳数为 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
```
配置对等体 2.2.2.9 作为 VPNv6 对等体。
```
[PE2-bgp] ipv6-family vpnv6
[PE2-bgp-af-vpnv6] peer 2.2.2.9 enable
[PE2-bgp-af-vpnv6] quit
```
将直连路由引入 vpn1 的路由表。
```
[PE2-bgp] ipv6-family vpn-instance vpn1
[PE2-bgp-ipv6-vpn1] import-route direct
[PE2-bgp-ipv6-vpn1] quit
[PE2-bgp] quit
```
配置完成后，在 PE2 上对 PE1 进行 Ping 操作，可以 Ping 通：
```
[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
```
在 PE1 上对 PE2 进行 Ping 操作，也可以 Ping 通：
```
[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
```

2.6.4 配置运营商的运营商示例

1. 组网需求

在 图 2-6 中：

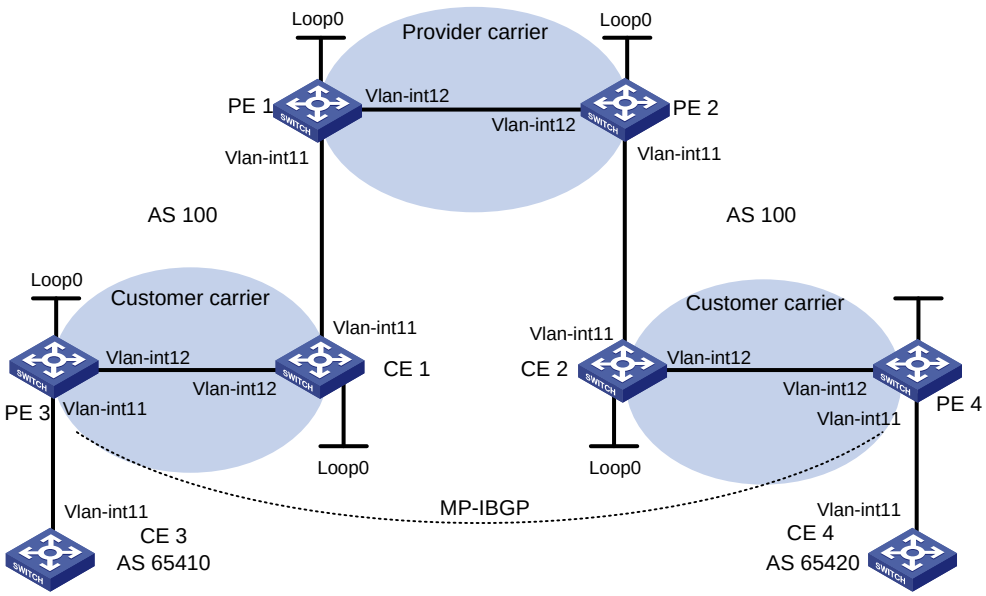
- PE 1 和 PE 2 是一级运营商骨干网的 PE 设备，为二级运营商提供 VPN 服务；
- CE 1 和 CE 2 是同一个二级运营商的设备，作为 CE 接入一级运营商的骨干网；
- PE 3 和 PE 4 是二级运营商的 PE 设备，为二级运营商的客户提供 IPv6 MPLS L3VPN 服务；
- CE 3 和 CE 4 是二级运营商的客户。

配置运营商的运营商关键在于配置两类路由的交换：

- 二级运营商 VPN 内部路由在一级运营商骨干网上的交换：一级运营商将二级运营商作为自己的 CE 接入；
- 二级运营商本身客户的 VPN 路由在二级运营商 PE 设备间的交换：需要在二级运营商 PE 设备（PE 3 和 PE 4）间建立 MP-IBGP 对等体关系。

2. 组网图

图2-6 配置 Carriers' carriers 组网图



设备	接口	IP地址	设备	接口	IP地址
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

3. 配置步骤

(1) 配置一级运营商骨干网的 MPLS L3VPN，使用 IS-IS 作为骨干网的 IGP 协议，PE 1 和 PE 2 之间使能 LDP，并建立 MP-IBGP 对等体关系

配置 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] mpls ldp transport-address interface
[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
```

说明

PE 2 的配置与 PE 1 类似，配置过程省略。

配置完成后，在 PE 1 或 PE 2 上执行 **display mpls ldp session** 命令可以看到 LDP 会话建立成功；执行 **display bgp peer** 命令可以看到 BGP 对等体关系已建立，并达到 Established 状态；执行 **display isis peer** 命令可以看到 IS-IS 邻居关系已建立，状态为 up。

以 PE 1 为例：

```
[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) 配置二级运营商网络：使用 IS-IS 作为 IGP 协议，PE 3 和 CE 1、PE 4 和 CE 2 之间分别使能 LDP

配置 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
```

配置 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
```

配置完成后，PE 3 和 CE 1 之间应能建立 LDP 和 IS-IS 邻居关系。

说明

PE 4 和 CE 2 之间的配置与 PE 3 和 CE 1 之间的配置类似，配置过程省略。

(3) 配置二级运营商 CE 接入到一级运营商的 PE

配置 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
```

配置 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
```

配置完成后，PE 1 和 CE 1 之间应能建立 LDP 和 IS-IS 邻居关系。

说明

PE 2 和 CE 2 之间的配置与 PE 1 和 CE 1 之间的配置类似，配置过程省略。

(4) 配置二级运营商的客户接入 PE

配置 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
```

配置 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
```

说明

PE 4 和 CE 4 之间的配置与 PE 3 和 CE 3 之间的配置类似，配置过程省略。

(5) 在二级运营商的 PE 之间建立 MP-IBGP 对等体关系，交换二级运营商的客户的 VPN 路由

配置 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
```

说明

PE 4 的配置与 PE 3 类似，配置过程省略。

(6) 配置完成后的检验

所有配置完成后，在 PE 1 和 PE 2 上执行 **display ip routing-table** 命令，可以看到 PE 1 和 PE 2 的公网路由表中只有一级运营商网络的路由。以 PE 1 为例：

```
[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
```

在 PE 1 和 PE 2 上执行 **display ip routing-table vpn-instance** 命令，可以看到 VPN 路由表中有二级运营商网络的内部路由。执行 **display ipv6 routing-table vpn-instance** 命令，可以看到没有二级运营商维护的 VPN 路由。以 PE 1 为例：

```
[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
```

在 CE 1 和 CE 2 上执行 **display ip routing-table** 命令，可以看到公网路由表中有二级运营商网络的内部路由。执行 **display ipv6 routing-table vpn-instance** 命令可以看到没有二级运营商维护的 VPN 路由。以 CE 1 为例：

```
[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
```

在 PE 3 和 PE 4 上执行 **display ip routing-table** 命令，可以看到公网路由表中有二级运营商网络的内部路由。以 PE 3 为例：

```
[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 和 PE 4 可以相互 Ping 通：

```
[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 和 CE 4 可以互相 Ping 通。

```
[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
```