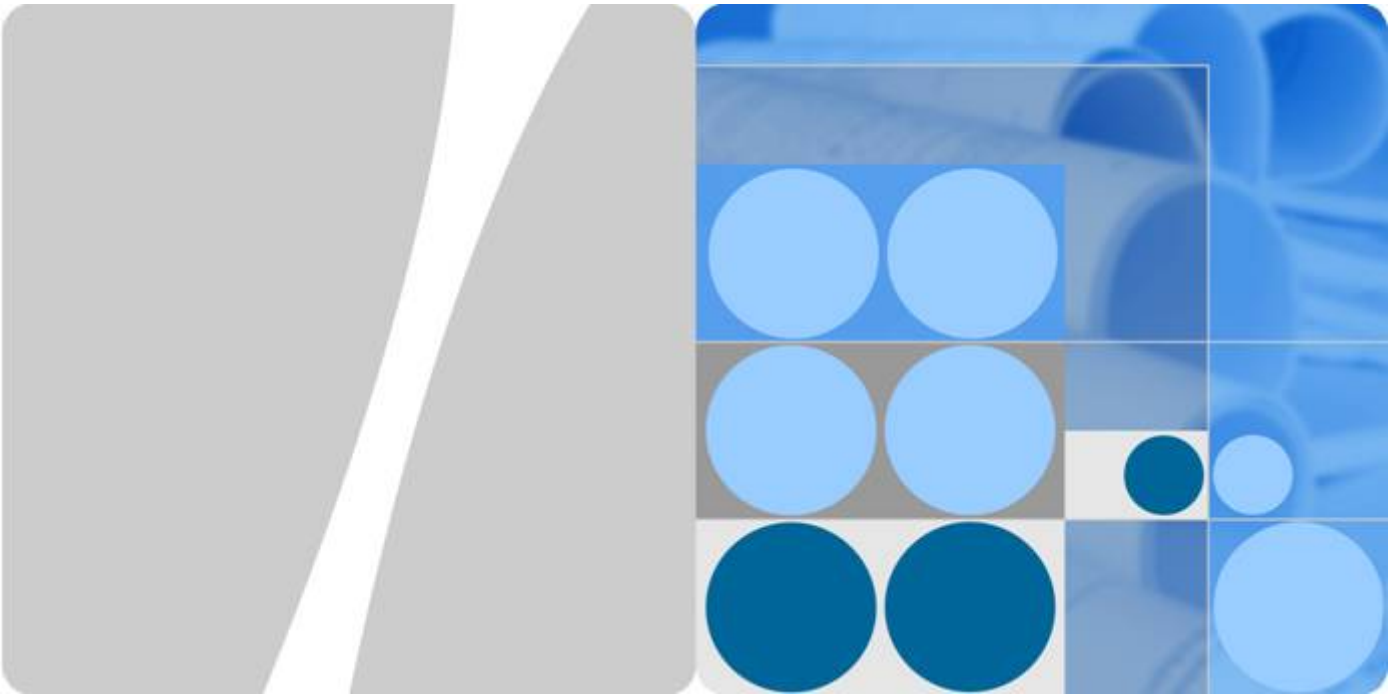


Huawei AR1200 系列企业路由器

V200R002C01

配置指南-VPN

文档版本　01

发布日期　2012-04-20

华为技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址： http://www.huawei.com

客户服务邮箱： support@huawei.com

客户服务电话： 4008302118

前 言

读者对象

本文档介绍了 AR1200 中 VPN 的基本概念、在不同应用场景中的配置过程和配置举例。

本文档提供了 VPN 的配置方法。

本文档主要适用于以下工程师：

- 数据配置工程师
- 调测工程师
- 网络监控工程师
- 系统维护工程师

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
危险	以本标志开始的文本表示有高度潜在危险，如果不能避免，会导致人员死亡或严重伤害。
警告	以本标志开始的文本表示有中度或低度潜在危险，如果不能避免，可能导致人员轻微或中等伤害。
注意	以本标志开始的文本表示有潜在风险，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
窍门	以本标志开始的文本能帮助您解决某个问题或节省您的时间。
说明	以本标志开始的文本是正文的附加信息，是对正文的强调和补充。

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 | ... }	表示从两个或多个选项中选取一个。
[x | y | ...]	表示从两个或多个选项中选取一个或者不选。
{ x | y | ... }*	表示从两个或多个选项中选取多个，最少选取一个，最多选取所有选项。
[x | y | ...]*	表示从两个或多个选项中选取多个或者不选。
&<1-n>	表示符号&前面的参数可以重复 1 ～ n 次。
#	由“#”开始的行表示为注释行。

接口编号约定

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

修订记录

修改记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

文档版本 01 (2012-04-20)

第一次正式发布。

目 录

1 GRE 协议配置

关于本章

GRE 是通用路由封装协议，可以对某些网络层协议的数据报进行封装，使这些被封装的数据报文能够在 IPv4 网络中传输。

1.1 GRE 协议概述
报文在 GRE（Generic Routing Encapsulation）隧道中传输包括封装和解封装两个过程。系统收到需要进行封装和依据路由进行传输的某网络层协议数据时，将首先对其加上 GRE 报文头，使之成为 GRE 报文，再将其封装在另一协议（如 IP）中。

1.2 AR1200 支持的 GRE 协议特性
AR1200 中支持的 GRE 特性包括：扩大了步跳数受限协议的网络的工作范围以及与 IPSec 结合使用，弥补 IPSec 不能保护组播数据的缺陷。

1.3 配置普通 GRE 隧道
在 GRE 的相关配置中，普通 GRE 隧道的配置是常见的一种场景也是后续 GRE 配置的基础。

1.4 配置 CE-PE 间的 GRE 隧道
配置 CE-PE 间的 GRE 隧道，主要实现私网用户边缘设备 CE 使用 GRE 隧道接入公网。

1.5 配置 GRE 支持 Keepalive 特性
在配置隧道策略，选择 GRE 作为 VPN 隧道前，先使能 GRE 隧道的 Keepalive 功能，可防止 VPN 选择对端不可达的 GRE 隧道，避免造成数据丢失。

1.6 维护 GRE
重置隧道接口统计信息、监控 GRE 协议运行状况。

1.7 配置举例
请结合配置流程图了解配置过程。配置示例中包括组网需求、配置注意事项、配置思路等。

1.1 GRE 协议概述

报文在 GRE（Generic Routing Encapsulation）隧道中传输包括封装和解封装两个过程。系统收到需要进行封装和依据路由进行传输的某网络层协议数据时，将首先对其加上 GRE 报文头，使之成为 GRE 报文，再将其封装在另一协议（如 IP）中。

通用路由封装是对某些网络层协议的报文进行封装，使这些被封装的报文能够在另一网络层协议（如 IP）中传输。

GRE 是 VPN 的第三层隧道（Tunnel）协议。Tunnel 是一个虚拟的点对点的连接，可以看成仅支持点对点连接的虚拟接口，这个接口提供了一条通路，使封装的数据报能够在这个通路上传输，并在一个 Tunnel 的两端分别对数据报进行封装及解封装。

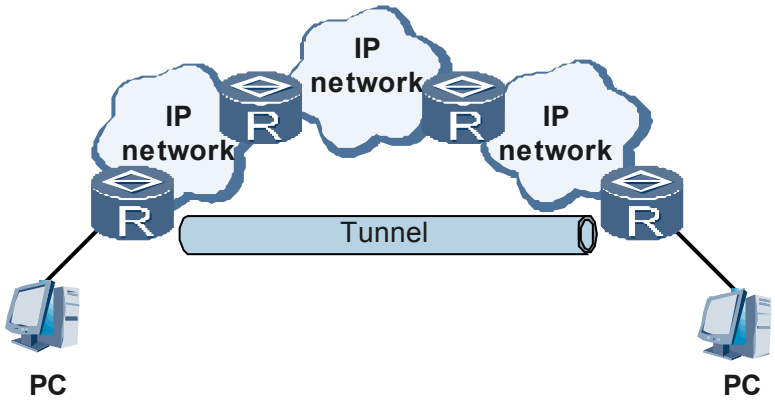
1.2 AR1200 支持的 GRE 协议特性

AR1200 中支持的 GRE 特性包括：扩大了步跳数受限协议的网络的工作范围以及与 IPSec 结合使用，弥补 IPSec 不能保护组播数据的缺陷。

扩大了步跳数受限协议（如 IPX）的网络的工作范围

在图 1-1 中，如果两台终端之间的步跳数超过 15，它们将无法通信。

图 1-1 扩大网络工作范围示意图

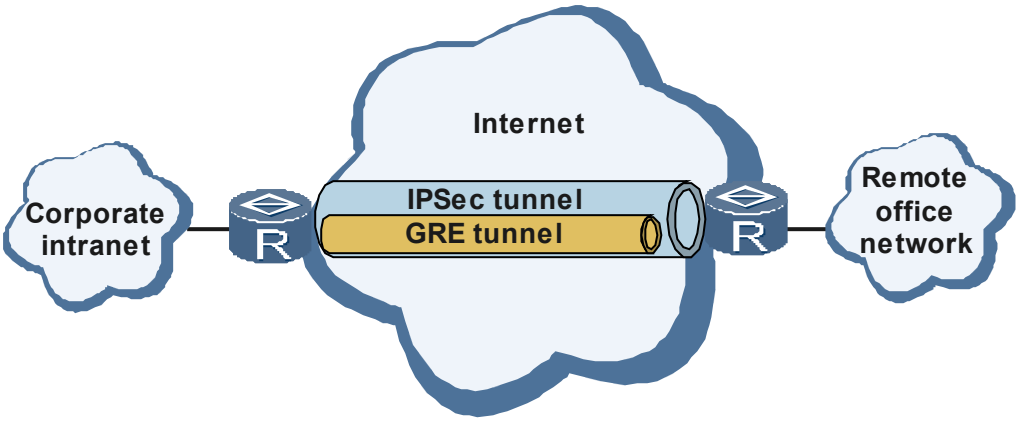


在网络中使用隧道（Tunnel）可以隐藏一部分步跳，从而扩大网络的工作范围。

与 IPSec 结合使用，弥补 IPSec 不能保护组播数据的缺陷

GRE 可以封装组播数据并在 GRE 隧道中传输，而 IPSec 目前只能对单播数据进行加密保护。

图 1-2 GRE-IPSec 隧道应用



如图 1-2 所示，对于组播数据需要在 IPSec 隧道中传输的情况，可以先建立 GRE 隧道，对组播数据进行 GRE 封装，再对封装后的报文进行 IPSec 加密，从而实现组播数据在 IPSec 隧道中的加密传输。

1.3 配置普通 GRE 隧道

在 GRE 的相关配置中，普通 GRE 隧道的配置是常见的一种场景也是后续 GRE 配置的基础。

1.3.1 建立配置任务

在配置普通 GRE 隧道前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

配置 GRE 需要创建 Tunnel 接口，在 Tunnel 接口上进行功能特性的配置。

当删除 Tunnel 接口后，该接口上的所有配置也将被删除。

前置任务

在配置 GRE 之前，需完成以下任务：

- 实现源接口和目的接口之间路由可达。

数据准备

在配置 GRE 之前，需准备以下数据。

序号	数据
1	Tunnel 接口的编号
2	Tunnel 的源地址和目的地址

序号	数据
3	Tunnel 接口的 IP 地址
4	Tunnel 接口的识别关键字

1.3.2 配置 Tunnel 接口

创建 Tunnel 接口后，需要指定封装方式为 GRE、设置 Tunnel 接口的源地址或源接口、设置 Tunnel 接口的目的端地址。此外为使隧道支持动态路由协议，还要配置 Tunnel 接口的 IP 地址。

背景信息

在隧道两端的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，创建 Tunnel 接口，并进入 Tunnel 接口视图。

步骤 3 执行命令 **tunnel-protocol gre**，将 Tunnel 封装为 GRE 隧道。

步骤 4 执行命令 **source** { *source-ip-address* | *interface-type interface-number* }，设置 Tunnel 接口的源地址或源接口。

隧道的源接口不能指定为自身的 Tunnel 接口，但可以指定为其他隧道的 Tunnel 接口。

说明

- GRE 隧道的源地址可以配置为 VRRP 备份组的虚地址。
- bridge-if 接口不可配置为 GRE 隧道的源接口。

步骤 5 执行命令 **destination** *ip-address*，设置 Tunnel 接口的目的端地址。

步骤 6 （可选）执行命令 **mtu** *mtu*，配置接口的 MTU。

如果需要改变 Tunnel 接口最大传输单元，需配置此步骤。执行该步骤后需要先对接口执行 **shutdown** 命令，再执行 **undo shutdown** 命令将接口重启，以保证设置的 MTU 生效。

步骤 7 指定 Tunnel 接口的 IP 地址，选择如下方法之一：

- 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 Tunnel 接口的 IP 地址。
- 执行命令 **ip address unnumbered interface** *interface-type interface-number*，配置 Tunnel 接口借用 IP 地址。

为使隧道支持动态路由协议，还要配置 Tunnel 接口的网络地址。Tunnel 接口的网络地址可以不是公网地址。隧道两端的网络地址应该位于同一网段。

缺省情况下，未设置 Tunnel 接口的网络地址。

----结束

1.3.3 配置 Tunnel 的路由

在源端设备和目的端设备上都必须存在经过 Tunnel 转发的路由，这样，需要进行 GRE 封装的报文才能正确转发。经过 Tunnel 接口的路由可以是静态路由，也可以是动态路由。

背景信息

在隧道两端的路由器上进行如下配置：

说明

在源端设备和目的端设备上都必须存在经过 Tunnel 转发的路由，这样，需要进行 GRE 封装的报文才能正确转发。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 配置经过 Tunnel 接口的路由，选择如下方法之一：

- 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } **tunnel** *interface-number* [**description** *text*]，配置静态路由。

  配置静态路由时，源端设备和目的端设备都需要配置：此路由目的地址不是 Tunnel 的目的地址，也不是对端 Tunnel 接口的地址，而是未进行 GRE 封装的报文的原始目的地址，出接口是本端 Tunnel 接口。

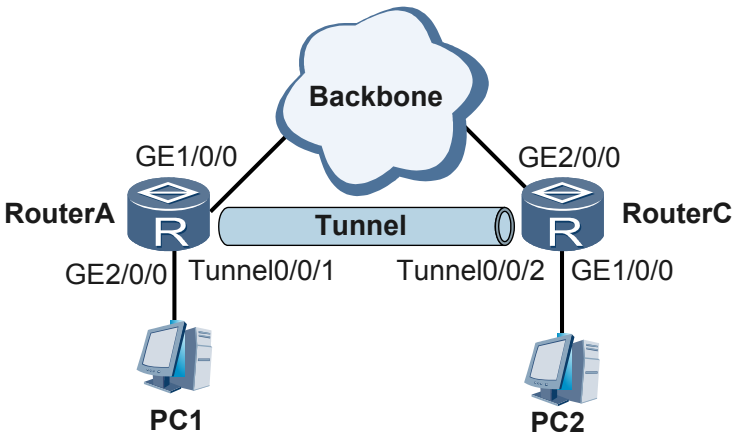
- 配置动态路由。可以使用 IGP 或 BGP，此处不再详述其配置方法。有关动态路由的配置，请参见《*Huawei AR1200 系列 企业路由器 配置指南-IP 路由*》。

  配置动态路由协议时，在 Tunnel 接口和与私网相连的路由器接口上都要使能该动态路由协议。并且，配置去往 Tunnel 目的端实际接口地址的路由时，为保证能够选择正确的路由，应注意 Tunnel 接口不能作为该路由的下一跳。

  例如，在图 1-3 中，对于 RouterA 而言，Tunnel0/0/1 的源端物理接口为 RouterA 的 GE1/0/0，目的端物理接口为 RouterC 的 GE2/0/0。如果使用动态路由协议，则 Tunnel 接口和接入 PC 的 GE 接口上都需要配置动态路由协议，并且，路由表中去往 RouterC 的 GE2/0/0 网段的出接口不能是 Tunnel0/0/1。

  实际配置时，可以采用多进程路由协议或改变 Tunnel 接口度量值，避免 Tunnel 接口被选择为去往 Tunnel 目的端物理地址的路由出接口。

图 1-3 配置 GRE 动态路由协议组网图



----结束

1.3.4 （可选）配置 GRE 的安全选项

为了增强 GRE 隧道的安全性，可以对 GRE 隧道两端进行端到端校验或者设置 GRE 隧道的识别关键字，通过这种安全机制防止错误识别、接收其它地方来的报文。

背景信息

在隧道两端的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，进入 Tunnel 接口视图。

步骤 3 执行命令 **gre checksum**，设置 Tunnel 进行端到端校验。

缺省情况下，禁止 Tunnel 两端进行端到端校验。

步骤 4 执行命令 **gre key** *key-number*，设置 Tunnel 接口的识别关键字。

如果在隧道两端设置 key-number，则必须指定相同的 key-number；或隧道两端都不设置 key-number。

缺省情况下，Tunnel 不使用识别关键字。

说明
步骤 3 和步骤 4 互不影响。

----结束

1.3.5 检查配置结果

普通 GRE 隧道配置成功后，您可以查看到 Tunnel 接口的工作状态和路由信息。

前提条件

已经完成普通 GRE 隧道功能的所有配置。

操作步骤

- 使用 **display interface tunnel** [*interface-number*]命令查看 Tunnel 接口的信息。
- 使用 **display ip routing-table** 命令查看 IPv4 路由表信息。
- 使用 **ping -a** *source-ip-address host* 命令查看隧道两端是否能互通。

----结束

任务示例

配置成功时，在隧道两端执行命令 **display interface tunnel**，可查看隧道接口的状态为 Up。例如：

```
<Huawei> display interface Tunnel 0/0/1
```

```
Tunnel0/0/1 current state : UP
Line protocol current state : UP
Description:HUAWEI, AR Series, Tunnel0/0/1 Interface
Route Port,The Maximum Transmit Unit is 1500
Internet Address is 5.5.5.2/24
Encapsulation is TUNNEL, loopback not set
Tunnel source 150.1.1.1 (Ethernet4/0/0), destination 150.1.1.2
Tunnel protocol/transport GRE/IP, key disabled
keepalive disabled
Checksumming of packets disabled
Current system time: 2008-03-04 19:17:30
    300 seconds input rate 0 bits/sec, 0 packets/sec
    300 seconds output rate 0 bits/sec, 0 packets/sec
    0 seconds input rate 0 bits/sec, 0 packets/sec
    0 seconds output rate 0 bits/sec, 0 packets/sec
    0 packets input,  0 bytes
    0 input error
    0 packets output,  0 bytes
    0 output error
    Input bandwidth utilization  : --
    Output bandwidth utilization : --
```

执行命令 **display ip routing-table**，可发现路由表中有经过该隧道接口转发的路由。例如：

```
<Huawei> display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto  Pre  Cost       Flags NextHop         Interface
       10.1.1.0/24  Direct 0    0              D  10.1.1.2        GigabitEthernet2/0/0
       10.1.1.2/32  Direct 0    0              D  127.0.0.1       InLoopBack0
       10.2.1.0/24  Static 60   0              D  40.1.1.1        Tunnel0/0/2
       20.1.1.1/32  Direct 0    0              D  127.0.0.1       InLoopBack0
       40.1.1.0/24  Direct 0    0              D  40.1.1.1        Tunnel0/0/2
       40.1.1.1/32  Direct 0    0              D  127.0.0.1       InLoopBack0
      127.0.0.0/8   Direct 0    0              D  127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct 0    0              D  127.0.0.1       InLoopBack0
```

执行命令 **ping -a** *source-ip-address host*，可发现从本端隧道接口 ping 目的端隧道接口地址，能 ping 通。

```
<Huawei> ping -a 40.1.1.1 40.1.1.2
  PING 40.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 40.1.1.2: bytes=56 Sequence=1 ttl=255 time=24 ms
    Reply from 40.1.1.2: bytes=56 Sequence=2 ttl=255 time=33 ms
    Reply from 40.1.1.2: bytes=56 Sequence=3 ttl=255 time=48 ms
    Reply from 40.1.1.2: bytes=56 Sequence=4 ttl=255 time=33 ms
    Reply from 40.1.1.2: bytes=56 Sequence=5 ttl=255 time=36 ms
  --- 40.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 24/34/48 ms
```

1.4 配置 CE-PE 间的 GRE 隧道

配置 CE-PE 间的 GRE 隧道，主要实现私网用户边缘设备 CE 使用 GRE 隧道接入公网。

1.4.1 建立配置任务

在配置 CE-PE 间的 GRE 隧道前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

如果 CE 不与 PE 直接连接，但 CE 所接入的用户需要接入到 MPLS VPN 中，这种需求可以通过在 CE-PE 间配置 GRE 隧道实现。要让 CE 所挂的用户接入到 MPLS VPN 中，还需再配置 PE 上的 MPLS VPN，以及 PE 与 CE 间的路由。

通常，在如下 2 种情况下需要创建 CE-PE 之间的 GRE 隧道：

- CE 经过公网与 PE 互连。
- CE 经过二级运营商的 VPN 与 PE 互连。

前置任务

在配置 CE-PE 间的 GRE 隧道之前，需完成以下任务：

- 配置 CE 与 PE 上接口的 IP 地址。
- 配置 CE-PE 间的路由。
- 如果 CE-PE 间的 GRE 隧道穿过另外的 VPN，还需要完成该 VPN 的配置。

数据准备

在配置 CE-PE 间的 GRE 隧道之前，需准备以下数据。

序号	数据
1	CE 上 GRE Tunnel 接口的编号
2	CE 上 GRE Tunnel 的源地址和目的地址
3	PE 上 GRE Tunnel 接口的编号
4	PE 上 GRE Tunnel 接口的源地址和目的地址
5	如果 GRE Tunnel 穿过另外的 VPN，还需知道该 VPN 的名字

1.4.2 在 CE 配置 Tunnel 接口

在 CE 上创建 Tunnel 接口后，需要指定封装方式为 GRE、设置 Tunnel 接口的源地址或源接口、设置 Tunnel 接口的目的端地址。其中，CE 上隧道的源地址与 PE 上隧道的目的地址相同，CE 上隧道的目的地址与 PE 上隧道的源地址相同。

背景信息

在 CE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，创建 Tunnel 接口，并进入 Tunnel 接口视图。

步骤 3 执行命令 **tunnel-protocol gre**，将 Tunnel 封装为 GRE 隧道。

步骤 4 执行命令 **source** { *source-ip-address* | *interface-type interface-number* }，设置 Tunnel 接口的源地址或源接口。

隧道的源接口不能指定为自身的 Tunnel 接口，但可以指定为其他隧道的 Tunnel 接口。

CE 上隧道的源地址与 PE 上隧道的目的地址相同，CE 上隧道的目的地址与 PE 上隧道的源地址相同。

说明

- GRE 隧道的源地址可以配置为 VRRP 备份组的虚地址。
- bridge-if 接口不可配置为 GRE 隧道的源接口。

步骤 5 执行命令 **destination** *ip-address*，设置 Tunnel 接口的目的端地址。

步骤 6 （可选）执行命令 **mtu** *mtu*，配置接口的 MTU。

如果需要改变 Tunnel 接口最大传输单元，需配置此步骤。执行该步骤后需要先对接口执行 **shutdown** 命令，再执行 **undo shutdown** 命令将接口重启，以保证设置的 MTU 生效。

步骤 7 指定隧道接口的 IP 地址，选择如下方法之一：

- 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 Tunnel 接口的 IP 地址。
- 执行命令 **ip address unnumbered interface** *interface-type interface-number*，配置 Tunnel 接口借用 IP 地址。

----结束

1.4.3 在 PE 配置 Tunnel 接口

在 PE 上创建 Tunnel 接口后，需要指定封装方式为 GRE、设置 Tunnel 接口的源地址或源接口、设置 Tunnel 接口的目的端地址。其中，PE 上隧道的源地址与 CE 上隧道的目的地址相同，PE 上隧道的目的地址与 CE 上隧道的源地址相同。

背景信息

在 PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，创建 Tunnel 接口，并进入 Tunnel 接口视图。

步骤 3 执行命令 **tunnel-protocol gre**，将 Tunnel 封装为 GRE 隧道。

步骤 4 执行命令 **source** { *source-ip-address* | *interface-type interface-number* }，设置 Tunnel 接口的源地址或源接口。

隧道的源接口不能指定为自身的 Tunnel 接口，但可以指定为其他隧道的 Tunnel 接口。

PE 上隧道的源地址与 CE 上隧道的目的地址相同，PE 上隧道的目的地址与 CE 上隧道的源地址相同。

说明

- GRE 隧道的源地址可以配置为 VRRP 备份组的虚地址。
- bridge-if 接口不可配置为 GRE 隧道的源接口。

步骤 5 执行命令 **destination** *ip-address*，设置 Tunnel 接口的目的端地址。

步骤 6 （可选）执行命令 **mtu** *mtu*，配置接口的 MTU。

如果需要改变 Tunnel 接口最大传输单元，需配置此步骤。执行该步骤后需要先对接口执行 **shutdown** 命令，再执行 **undo shutdown** 命令将接口重启，以保证设置的 MTU 生效。

步骤 7 指定隧道接口的 IP 地址，选择如下方法之一：

- 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 Tunnel 接口的 IP 地址。
- 执行命令 **ip address unnumbered interface** *interface-type interface-number*，配置 Tunnel 接口借用 IP 地址。

----结束

1.4.4 在 PE 上将 GRE Tunnel 与 CE 所在的 VPN 进行绑定

在 PE 上配置 VPN 实例与连接 CE 的 Tunnel 接口进行关联。关联后，Tunnel 接口成为私网接口。从该接口进入的报文使用 VPN 实例中的转发信息进行转发。

背景信息

在 PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，创建 Tunnel 接口，并进入 Tunnel 接口视图。

步骤 3 执行命令 **ip binding vpn-instance** *vpn-instance-name*，将 Tunnel 与 VPN 实例进行绑定。

说明

执行 **ip binding vpn-instance** 命令将删除 Tunnel 接口上已经配置的 IP 地址、路由协议等三层特性，如果需要应重新配置。

Tunnel 接口不能与未使能任何地址族的 VPN 实例绑定。

当去使能 VPN 实例的某地址族时，也将删除与 VPN 实例绑定 Tunnel 接口下的 IP 地址、路由协议等三层特性。当去使能 VPN 实例下所有地址族时，去使能 Tunnel 接口与该 VPN 实例的绑定。

步骤 4 设置 Tunnel 接口的 IP 地址，选择如下方法之一，选择如下方法之一：

- 执行命令 **ip address** *ip-address* { *mask* | *mask-length* } [**sub**]，配置 Tunnel 接口的 IP 地址。
- 执行命令 **ip address unnumbered interface** *interface-type interface-number*，配置 Tunnel 接口借用 IP 地址。

----结束

1.4.5 检查配置结果

CE-PE 间的 GRE 隧道配置成功后，您可以查看到指定 VPN 的路由信息。

前提条件

已经完成 CE-PE 间的 GRE 隧道功能的所有配置。

操作步骤

- 使用 **display interface tunnel** [*interface-number*]命令查看 Tunnel 接口的工作状态。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 PE 上查看 VPN 路由表。
- 使用 **display ip routing-table** 命令在 CE 上查看路由表。
- 使用 **ping -a** *source-ip-address host* 命令查看隧道两端是否能互通。

----结束

任务示例

配置成功时，在隧道两端执行命令 **display interface tunnel**，可查看隧道接口的状态为 Up。以 PE 端的显示为例：

```
<Huawei> display interface Tunnel 0/0/1
Tunnel0/0/1 current state : UP
Line protocol current state : UP
Last line protocol up time : 2008-03-03 10:51:44
Description:HUAWEI, AR Series, Tunnel0/0/1 Interface
Route Port,The Maximum Transmit Unit is 1500
Internet Address is 5.5.5.2/24
Encapsulation is TUNNEL, loopback not set
Tunnel source 150.1.1.1 (Ethernet4/0/0), destination 150.1.1.2
Tunnel protocol/transport GRE/IP, key disabled
keepalive disabled
Checksumming of packets disabled
Current system time: 2008-03-04 19:17:30
    300 seconds input rate 0 bits/sec, 0 packets/sec
    300 seconds output rate 0 bits/sec, 0 packets/sec
    0 seconds input rate 0 bits/sec, 0 packets/sec
    0 seconds output rate 0 bits/sec, 0 packets/sec
    0 packets input,  0 bytes
    0 input error
    0 packets output,  0 bytes
    0 output error
    Input bandwidth utilization  : --
    Output bandwidth utilization : --
```

1.5 配置 GRE 支持 Keepalive 特性

在配置隧道策略，选择 GRE 作为 VPN 隧道前，先使能 GRE 隧道的 Keepalive 功能，可防止 VPN 选择对端不可达的 GRE 隧道，避免造成数据丢失。

1.5.1 建立配置任务

在配置 GRE 支持 Keepalive 特性前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

图 1-4 支持 Keepalive 的 GRE 隧道组网图

在源端配置支持 Keepalive 特性的 GRE，使源端能够检测隧道状态，避免因对端不可达而造成的数据丢失。

前置任务

在配置 Keepalive 之前，需完成以下任务：

- 配置接口的链路层属性。
- 配置接口的 IP 地址。
- 建立 GRE 隧道，且隧道状态为 Up。

数据准备

在配置 Keepalive 之前，需准备以下数据。

序号	数据
1	发送 Keepalive 报文的周期
2	不可达计数器参数

1.5.2 使能 Keepalive 功能

GRE 隧道的 Keepalive 功能是单向的。要使两端都具备 Keepalive 功能，需在两端都使能 GRE 隧道的 Keepalive 功能。

背景信息

在需要使能 Keepalive 功能的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，进入 Tunnel 接口视图。

步骤 3 执行命令 **tunnel-protocol gre**，将 Tunnel 封装为 GRE 隧道。

步骤 4 执行命令 **keepalive** [**period** *period* [**retry-times** *retry-times*]]，使能 GRE 的 Keepalive 功能。

GRE 隧道的 Keepalive 功能是单向的。要使两端都具备 Keepalive 功能，需在两端都使能 GRE 隧道的 Keepalive 功能。对端是否支持 Keepalive 功能不影响本端的 Keepalive 功能。但建议在隧道两端都使能 Keepalive 功能。

窍门

在配置隧道策略，选择 GRE 作为 VPN 隧道前，先使能 GRE 隧道的 Keepalive 功能，可防止 VPN 选择对端不可达的 GRE 隧道，避免造成数据丢失。因为：

在没有使能 Keepalive 功能的情况下，即使对端不可达，本端 Tunnel 接口状态也可能为 Up。

使能本端的 Keepalive 后，如果对端不可达，本端的 Tunnel 接口状态会被置为 Down。因此，如果对端不可达，本端 VPN 就不会选择该 GRE 隧道，避免造成数据丢失。

----结束

1.5.3 检查配置结果

GRE 支持 Keepalive 配置成功后，您可以查看到 GRE tunnel 接口收发 Keepalive 报文和 Keepalive 响应报文的情况。

前提条件

已经完成 GRE 支持 Keepalive 功能的所有配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，进入 Tunnel 接口视图。

步骤 3 执行 **display keepalive packets count** 命令查看 GRE tunnel 接口收发 Keepalive 报文和 Keepalive 响应报文的情况。

----结束

任务示例

在使能了 GRE 的 Keepalive 功能的 Tunnel 接口上执行命令 **display keepalive packets count**，可以看到该 GRE 隧道接口发送给对端的 keepalive 报文的数量和 keepalive 响应报文的数量，以及从对端接收的 keepalive 报文的数量和 keepalive 响应报文的数量。如果本端 GRE Tunnel 接口的 Keepalive 功能配置成功，则本端发送给对端的 keepalive 报文的数量和本端收到的 keepalive 响应报文的数量不为 0。

```
[Huawei] interface tunnel 0/0/1
[Huawei-Tunnel0/0/1] tunnel-protocol gre
[Huawei-Tunnel0/0/1] keepalive
[Huawei-Tunnel0/0/1] display keepalive packets count
Send 34 keepalive packets to peers, Receive 34 keepalive response packets from peers
Receive 0 keepalive packets from peers, Send 0 keepalive response packets to peers.
```

1.6 维护 GRE

重置隧道接口统计信息、监控 GRE 协议运行状况。

1.6.1 重置隧道接口统计信息

当需要重置隧道接口的统计信息时，可以执行 reset 命令来将 GRE 隧道接口发送给对端的 Keepalive 报文数量和 Keepalive 响应报文数量，以及从对端接收的 Keepalive 报文数量和 Keepalive 响应报文数量的统计信息清零。

操作步骤

- 请在系统视图下执行 **reset counters interface tunnel** [*interface-number*]命令重置 Tunnel 接口的统计信息。
- 重置 Tunnel 接口的 Keepalive 报文相关统计信息。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface tunnel** *interface-number*，进入 Tunnel 接口视图。
  3. 使用 **reset keepalive packets count** 命令重置 Tunnel 接口的 Keepalive 报文相关统计信息。

     说明

     命令 **reset keepalive packets count** 只能在 Tunnel 接口视图下执行，且该接口的隧道协议必须配置为 GRE。

----结束

1.6.2 监控 GRE 协议运行状况

在日常维护工作中，执行与 GRE 相关的 display 命令，了解 GRE 协议的运行情况。

背景信息

在日常维护工作中，可以在任意视图下选择执行以下命令，了解 GRE 协议的运行情况。

操作步骤

- 使用 **display interface tunnel** [*interface-number*]命令查看 Tunnel 接口的信息。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 PE 上查看 VPN 路由表。
- 使用 **display ip routing-table** 命令在 CE 上查看路由表。
- 使用 **ping** [**-a** *source-ip-address* | **-vpn-instance** *vpn-instance-name*] * *host* 命令查看隧道两端是否能互通。

----结束

1.6.3 调试 GRE 配置

在出现 GRE 运行故障时，执行 debugging 命令进行调试，查看调试信息，定位故障并分析故障的原因。

背景信息

注意

打开调试开关将影响系统的性能。调试完毕后，应及时执行 **undo debugging all** 命令关闭调试开关。

在出现 GRE 运行故障时，请在用户视图下执行下面的 **debugging** 命令进行调试，查看调试信息，定位故障并分析故障的原因。

操作步骤

- 在用户视图下执行 **debugging tunnel keepalive** 命令打开 GRE Tunnel 的 Keepalive 功能调试开关。

----结束

1.7 配置举例

请结合配置流程图了解配置过程。配置示例中包括组网需求、配置注意事项、配置思路等。

1.7.1 配置 GRE 使用静态路由示例

以典型组网为背景，介绍如何配置 GRE 使用静态路由。使得用户端之间的流量通过 GRE 隧道传输，设备到与其相连的客户端之间需要配置静态路由。

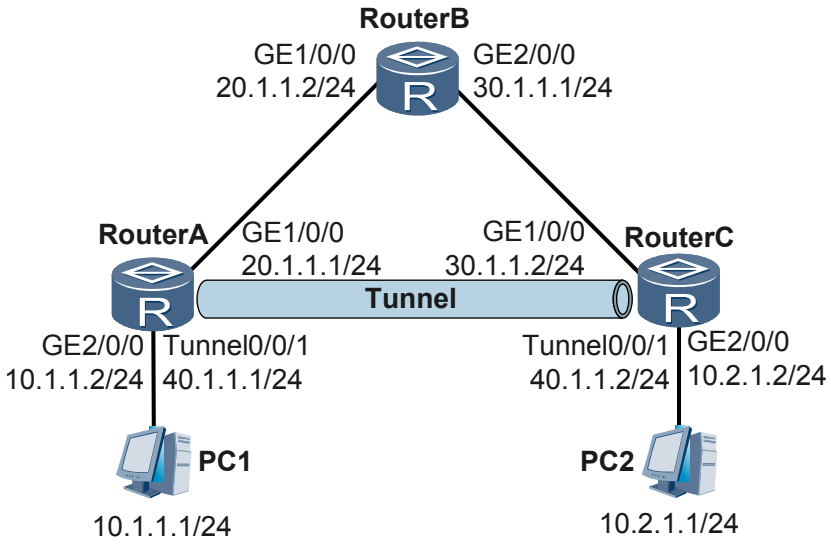
组网需求

如图 1-5，RouterA、RouterB、RouterC 属于 VPN 骨干网，它们之间运行 OSPF。

RouterA 和 RouterC 之间使用三层隧道协议 GRE，实现 PC1 和 PC2 互联。

PC1 和 PC2 上分别指定 RouterA、RouterC 为自己的缺省网关。

图 1-5 配置 GRE 使用静态路由组网图



配置思路

配置 GRE 使用静态路由的思路如下：

1. 路由器运行动态路由协议实现互通。
2. 在 RouterA 和 RouterC 上创建 Tunnel 接口，指定 Tunnel 的源地址和目的地址。注意 Tunnel 的源地址是发出报文的实际接口 IP 地址，目的地址是接收报文的实际接口 IP 地址。
3. 为使隧道支持动态路由协议，需要配置 Tunnel 接口的网络地址。
4. 为使 PC1 和 PC2 之间的流量通过 GRE 隧道传输，RouterA 和 RouterC 上需配置到各自相连的 PC 的静态路由，出接口为本端的 Tunnel 接口。

数据准备

为完成此配置例，需准备如下的数据：

- 运行 OSPF 所需数据
- GRE 隧道两端的源地址、目的地址、Tunnel 接口 IP 地址

操作步骤

步骤 1 配置各接口 IP 地址

按照图 1-5 配置各接口的 IP 地址，具体配置过程略。

步骤 2 配置 VPN 骨干网的 IGP

配置 RouterA。

```
[RouterA] ospf 1
[RouterA-ospf-1] area 0
[RouterA-ospf-1-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[RouterA-ospf-1-area-0.0.0.0] quit
```

```
[RouterA-ospf-1] quit
```

配置 RouterB。

```
[RouterB] ospf 1
[RouterB-ospf-1] area 0
[RouterB-ospf-1-area-0.0.0.0] network 20.1.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] network 30.1.1.0 0.0.0.255
[RouterB-ospf-1-area-0.0.0.0] quit
[RouterB-ospf-1] quit
```

配置 RouterC。

```
[RouterC] ospf 1
[RouterC-ospf-1] area 0
[RouterC-ospf-1-area-0.0.0.0] network 30.1.1.0 0.0.0.255
[RouterC-ospf-1-area-0.0.0.0] quit
[RouterC-ospf-1] quit
```

配置完成后，在 RouterA 和 RouterC 上执行 **display ip routing-table** 命令，可以看到它们能够学到去往对端接口网段地址的 OSPF 路由。

以 RouterA 的显示为例。

```
[RouterA] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        10.1.1.0/24  Direct  0    0           D   10.1.1.2        GigabitEthernet2/0/0
        10.1.1.2/32  Direct  0    0           D   127.0.0.1       InLoopBack0
        20.1.1.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
        30.1.1.0/24  OSPF    10   2           D   20.1.1.2        GigabitEthernet1/0/0
       127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
       127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

步骤 3 配置 Tunnel 接口

配置 RouterA。

```
[RouterA] interface tunnel 0/0/1
[RouterA-Tunnel0/0/1] tunnel-protocol gre
[RouterA-Tunnel0/0/1] ip address 40.1.1.1 255.255.255.0
[RouterA-Tunnel0/0/1] source 20.1.1.1
[RouterA-Tunnel0/0/1] destination 30.1.1.2
[RouterA-Tunnel0/0/1] quit
```

配置 RouterC。

```
[RouterC] interface tunnel 0/0/1
[RouterC-Tunnel0/0/1] tunnel-protocol gre
[RouterC-Tunnel0/0/1] ip address 40.1.1.2 255.255.255.0
[RouterC-Tunnel0/0/1] source 30.1.1.2
[RouterC-Tunnel0/0/1] destination 20.1.1.1
[RouterC-Tunnel0/0/1] quit
```

配置完成后，Tunnel 接口状态变为 Up，Tunnel 接口之间可以 Ping 通。

以 RouterA 的显示为例：

```
[RouterA] ping -a 40.1.1.1 40.1.1.2
  PING 40.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 40.1.1.2: bytes=56 Sequence=1 ttl=255 time=24 ms
    Reply from 40.1.1.2: bytes=56 Sequence=2 ttl=255 time=33 ms
    Reply from 40.1.1.2: bytes=56 Sequence=3 ttl=255 time=48 ms
    Reply from 40.1.1.2: bytes=56 Sequence=4 ttl=255 time=33 ms
    Reply from 40.1.1.2: bytes=56 Sequence=5 ttl=255 time=36 ms
  --- 40.1.1.2 ping statistics ---
```

```
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 24/34/48 ms
```

步骤 4 配置静态路由

配置 RouterA。

```
[RouterA] ip route-static 10.2.1.0 255.255.255.0 tunnel 0/0/1
```

配置 RouterC。

```
[RouterC] ip route-static 10.1.1.0 255.255.255.0 tunnel 0/0/1
```

配置完成后，在 RouterA 和 RouterC 上执行 **display ip routing-table** 命令，可以看到使用 Tunnel 接口去往对端用户侧网段的静态路由。

以 RouterA 的显示为例。

```
[RouterA] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 11        Routes : 11

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        10.1.1.0/24  Direct 0    0           D   10.1.1.2        GigabitEthernet2/0/0
        10.1.1.2/32  Direct 0    0           D   127.0.0.1       InLoopBack0
        10.2.1.0/24  Static 60   0           D   40.1.1.1        Tunnel0/0/1
        20.1.1.0/24  Direct 0    0           D   20.1.1.1        GigabitEthernet1/0/0
        20.1.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
        30.1.1.0/24  OSPF   10   2           D   20.1.1.2        GigabitEthernet1/0/0
        40.1.1.0/24  Direct 0    0           D   40.1.1.1        Tunnel0/0/1
        40.1.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
       127.0.0.0/8   Direct 0    0           D   127.0.0.1       InLoopBack0
       127.0.0.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
```

PC1 和 PC2 可以相互 Ping 通。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
interface GigabitEthernet1/0/0
 ip address 20.1.1.1 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 10.1.1.2 255.255.255.0
#
interface Tunnel0/0/1
 ip address 40.1.1.1 255.255.255.0
 tunnel-protocol gre
 source 20.1.1.1
 destination 30.1.1.2
#
ospf 1
 area 0.0.0.0
  network 20.1.1.0 0.0.0.255
#
ip route-static 10.2.1.0 255.255.255.0 Tunnel0/0/1
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface GigabitEthernet1/0/0
ip address 20.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
ip address 30.1.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 20.1.1.0 0.0.0.255
  network 30.1.1.0 0.0.0.255
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
interface GigabitEthernet1/0/0
ip address 30.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 10.2.1.2 255.255.255.0
#
interface Tunnel0/0/1
 ip address 40.1.1.2 255.255.255.0
 tunnel-protocol gre
 source 30.1.1.2
 destination 20.1.1.1
#
ospf 1
 area 0.0.0.0
  network 30.1.1.0 0.0.0.255
#
ip route-static 10.1.1.0 255.255.255.0 Tunnel0/0/1
#
return
```

1.7.2 配置 GRE 使用动态路由协议示例

以典型组网为背景，介绍如何配置 GRE 使用动态路由。使得用户端之间的流量通过 GRE 隧道传输，设备到与其相连的客户端之间需要配置动态路由协议。

组网需求

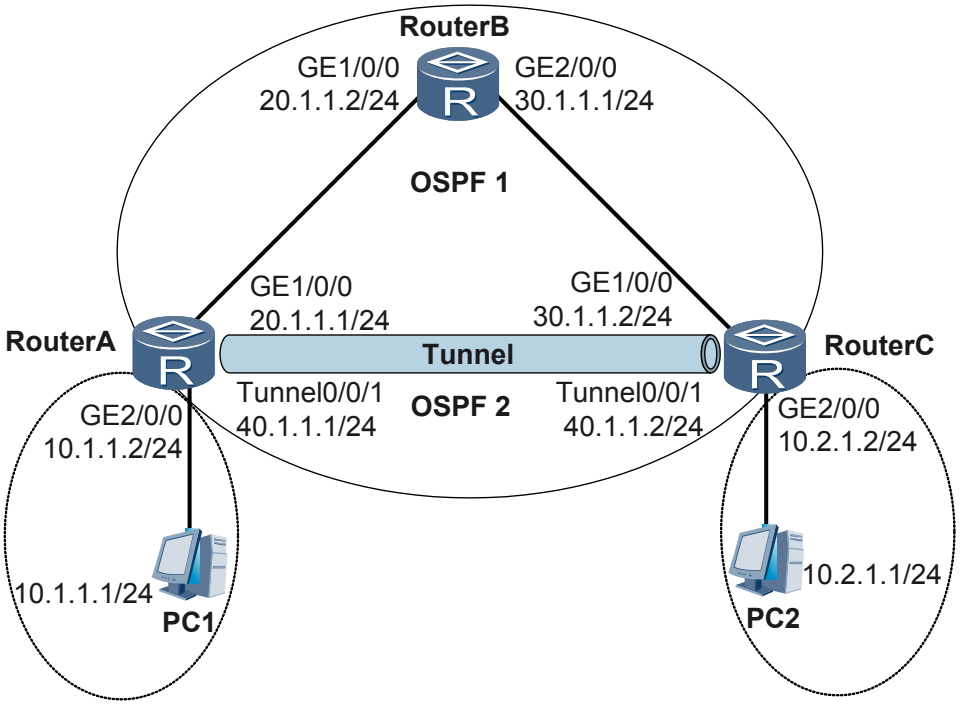
如图 1-6，RouterA、RouterB、RouterC 属于 VPN 骨干网，它们之间运行 OSPF。

RouterA 和 RouterC 之间使用三层隧道协议 GRE，实现 PC1 和 PC2 互联。

PC1 和 PC2 上分别指定 RouterA、RouterC 为自己的缺省网关。

Tunnel 接口启用动态路由协议 OSPF。VPN 骨干网上使用 OSPF 进程 1，用户接入部分使用 OSPF 进程 2。

图 1-6 配置 GRE 使用动态路由协议组网图



配置思路

配置 GRE 使用动态路由协议的思路如下：

1. 在骨干网上各路由器运行 IGP 协议实现互通，这里用 OSPF 进程 1。
2. 与 PC 相连的路由器之间建立 GRE 隧道，使其彼此之间传输都通过 GRE 隧道。
3. PC 接入骨干网的那部分网段运行动态路由协议，这里用的是 OSPF 进程 2。

数据准备

为完成此配置例，需准备如下的数据：

- GRE 隧道两端的源地址、目的地址
- 两端 Tunnel 接口 IP 地址

操作步骤

步骤 1 配置各接口 IP 地址

按照图 1-6 配置各接口的 IP 地址，具体配置过程略。

步骤 2 配置 VPN 骨干网的 IGP

与**配置 GRE 使用静态路由示例**相同，具体配置过程略。

步骤 3 配置 Tunnel 接口

与**配置 GRE 使用静态路由示例**相同，具体配置过程略。

步骤 4 配置 Tunnel 接口的 OSPF 协议

配置 RouterA。

```
[RouterA] ospf 2
[RouterA-ospf-2] area 0
[RouterA-ospf-2-area-0.0.0.0] network 40.1.1.0 0.0.0.255
[RouterA-ospf-2-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[RouterA-ospf-2-area-0.0.0.0] quit
[RouterA-ospf-2] quit
```

配置 RouterC。

```
[RouterC] ospf 2
[RouterC-ospf-2] area 0
[RouterC-ospf-2-area-0.0.0.0] network 40.1.1.0 0.0.0.255
[RouterC-ospf-2-area-0.0.0.0] network 10.2.1.0 0.0.0.255
[RouterC-ospf-2-area-0.0.0.0] quit
[RouterC-ospf-2] quit
```

步骤 5 检查配置结果

配置完成后，在 RouterA 和 RouterC 上执行 **display ip routing-table** 命令，可以看到经过 Tunnel 接口去往对端用户侧网段的 OSPF 路由，并且，去往 Tunnel 目的端物理地址（30.1.1.0/24）的路由下一跳不是 Tunnel 接口。

以 RouterA 的显示为例。

```
[RouterA] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 11        Routes : 11

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        10.1.1.0/24  Direct 0    0           D   10.1.1.2   GigabitEthernet2/0/0
        10.1.1.2/32  Direct 0    0           D   127.0.0.1       InLoopBack0
        10.2.1.0/24  OSPF   10   2           D   40.1.1.2        Tunnel0/0/1
        20.1.1.0/24  Direct 0    0           D   20.1.1.1        GigabitEthernet1/0/0
        20.1.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
        30.1.1.0/24  OSPF   10   2           D   20.1.1.2        GigabitEthernet1/0/0
        40.1.1.0/24  Direct 0    0           D   40.1.1.1        Tunnel0/0/1
        40.1.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
       127.0.0.0/8   Direct 0    0           D   127.0.0.1       InLoopBack0
       127.0.0.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
```

PC1 和 PC2 可以相互 Ping 通。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
interface GigabitEthernet1/0/0
ip address 20.1.1.1 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 10.1.1.2 255.255.255.0
#
interface Tunnel0/0/1
 ip address 40.1.1.1 255.255.255.0
 tunnel-protocol gre
 source 20.1.1.1
 destination 30.1.1.2
#
```

```
ospf 1
 area 0.0.0.0
  network 20.1.1.0 0.0.0.255
#
ospf 2
 area 0.0.0.0
  network 40.1.1.0 0.0.0.255
  network 10.1.1.0 0.0.0.255
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface GigabitEthernet1/0/0
ip address 20.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
ip address 30.1.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 20.1.1.0 0.0.0.255
  network 30.1.1.0 0.0.0.255
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
interface GigabitEthernet1/0/0
ip address 30.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 10.2.1.2 255.255.255.0
#
interface Tunnel0/0/1
 ip address 40.1.1.2 255.255.255.0
 tunnel-protocol gre
 source 30.1.1.2
 destination 20.1.1.1
#
ospf 1
 area 0.0.0.0
  network 30.1.1.0 0.0.0.255
#
ospf 2
 area 0.0.0.0
  network 40.1.1.0 0.0.0.255
  network 10.2.1.0 0.0.0.255
#
return
```

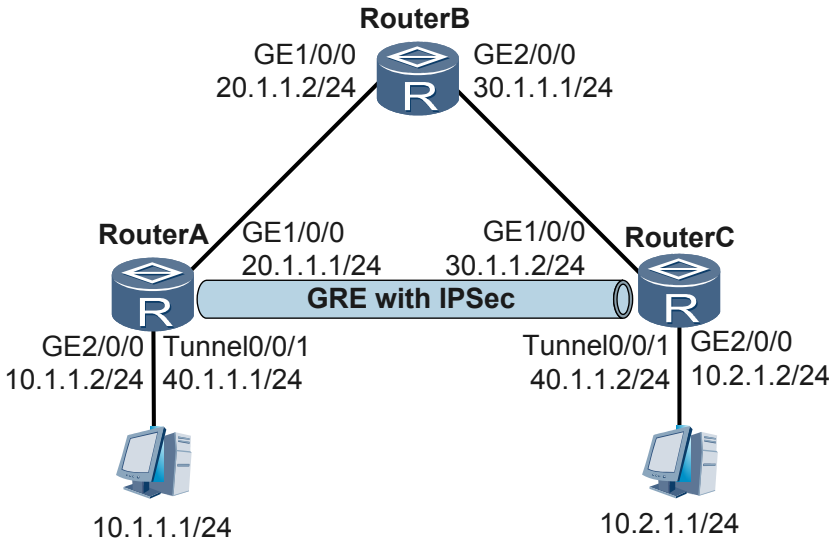
1.7.3 配置 GRE 封装 IPSec 传输组播数据示例

以典型组网为背景，介绍如何配置 GRE 封装 IPSec 传输组播数据。设备之间建立 GRE 隧道，先对组播报文进行 GRE 封装，然后建立 IPSec 隧道加密 GRE 封装后的报文。

组网需求

如图 1-7 所示，RouterA 和 RouterC 之间传输组播数据，并要对数据进行 IPSec 加密。由于组播数据无法直接应用 IPSec，因此先对组播数据进行 GRE 封装，再进行 IPSec 加密。

图 1-7 配置 GRE 封装 IPSec 传输组播数据组网图



配置思路

配置 GRE 封装 IPSec 的思路如下：

1. 在骨干网设备 RouterA、RouterB 和 RouterC 上运行 OSPF 协议实现互通。
2. RouterA 与 RouterC 之间建立 GRE 隧道，对组播报文进行封装。
3. 在 RouterA 和 RouterC 上建立 IPSec 隧道加密 GRE 封装后的报文。

数据准备

为完成此配置例，需准备如下的数据：

- 配置骨干网路由协议所需数据
- GRE 两端的源地址、目的地址和 Tunnel 接口 IP 地址
- 配置 IKE 所需数据，如 pre-shared-key、remote-name
- 配置 IPSec 时所需数据，如安全提议名称、ACL

配置过程

1. 配置路由协议

   在 RouterA、RouterB、RouterC 上配置路由协议，实现互通。本例采用 OSPF 协议，具体配置过程略。

   完成此步配置后，RouterA 与 RouterC 之间有可达的路由。RouterA 可以 ping 通 RouterC 的接口 GE1/0/0；RouterC 可以 ping 通 RouterA 的接口 GE1/0/0。

2. 配置 GRE 隧道接口

   # 在 RouterA 上配置。

```
[RouterA] interface tunnel0/0/1
[RouterA-Tunnel0/0/1] ip address 40.1.1.1 255.255.255.0
[RouterA-Tunnel0/0/1] tunnel-protocol gre
[RouterA-Tunnel0/0/1] source 20.1.1.1
```

```
[RouterA-Tunnel0/0/1] destination 30.1.1.2
[RouterA-Tunnel0/0/1] quit
```

在 RouterC 上配置。

```
[RouterC] interface tunnel0/0/1
[RouterC-Tunnel0/0/1] ip address 40.1.1.2 255.255.255.0
[RouterA-Tunnel0/0/1] tunnel-protocol gre
[RouterC-Tunnel0/0/1] source 30.1.1.2
[RouterC-Tunnel0/0/1] destination 20.1.1.1
[RouterC-Tunnel0/0/1] quit
```

上述配置完成后，RouterA 与 RouterC 之间的 GRE 隧道已建立，Tunnel 接口状态为 Up。

3. 使能组播

全局使能组播路由协议，在 Tunnel 接口下使能 PIM DM，并在与 PC 相连的接口使能 PIM DM 和 IGMP。

配置 RouterA。

```
[RouterA] multicast routing-enable
[RouterA] interface gigabitethernet 2/0/0
[RouterA-GigabitEthernet2/0/0] pim dm
[RouterA-GigabitEthernet2/0/0] igmp enable
[RouterA-GigabitEthernet2/0/0] quit
[RouterA] interface tunnel0/0/1
[RouterA-Tunnel0/0/1] pim dm
[RouterA-Tunnel0/0/1] quit
```

配置 RouterC。

```
[RouterC] multicast routing-enable
[RouterC] interface gigabitethernet 2/0/0
[RouterC-GigabitEthernet2/0/0] pim dm
[RouterC-GigabitEthernet2/0/0] igmp enable
[RouterC-GigabitEthernet2/0/0] quit
[RouterC] interface tunnel0/0/1
[RouterC-Tunnel0/0/1] pim dm
[RouterC-Tunnel0/0/1] quit
```

使能组播后，RouterA 和 RouterC 之间的组播数据将通过 GRE 隧道传输。

4. 配置 RouterA 和 RouterC 之间采用野蛮模式进行 IKE 协商

说明

先进行 GRE 封装，在进行 IPSec 加密，要求 ike peer 模式下的 remote-address 为本端 Tunnel 的 destination 的地址。

配置 RouterA。

```
[RouterA] ike local-name rta
[RouterA] ike peer RouterC v1
[RouterA-ike-peer-routerc] exchange-mode aggressive
[RouterA-ike-peer-routerc] local-id-type name
[RouterA-ike-peer-routerc] pre-shared-key 12345
[RouterA-ike-peer-routerc] remote-name rtc
[RouterA-ike-peer-routerc] remote-address 30.1.1.2
[RouterA-ike-peer-routerc] quit
```

配置 RouterC。

```
[RouterC] ike local-name rtc
[RouterC] ike peer RouterA v1
[RouterC-ike-peer-routera] exchange-mode aggressive
[RouterC-ike-peer-routera] local-id-type name
[RouterC-ike-peer-routera] pre-shared-key 12345
[RouterC-ike-peer-routera] remote-name rta
[RouterC-ike-peer-routera] remote-address 20.1.1.1
[RouterC-ike-peer-routera] quit
```

5. 配置 IPSec

说明

先进行 GRE 封装，再进行 IPSec 加密，要求 ipsec policy 的 ACL 匹配本端 Tunnel 的 source、destination 地址，并将该 policy 应用在实际传送数据的物理接口。

在 RouterA 和 RouterC 上进行 IPSec 配置，本例使用缺省的安全提议参数。

配置 RouterA。

```
[RouterA] acl number 3000
[RouterA-acl-adv-3000] rule permit gre source 20.1.1.1 0 destination 30.1.1.2 0
[RouterA-acl-adv-3000] quit
[RouterA] ipsec proposal p1
[RouterA-ipsec-proposal-p1] quit
[RouterA] ipsec policy policy1 1 isakmp
[RouterA-ipsec-policy-isakmp-policy1-1] security acl 3000
[RouterA-ipsec-policy-isakmp-policy1-1] ike-peer RouterC
[RouterA-ipsec-policy-isakmp-policy1-1] proposal p1
[RouterA-ipsec-policy-isakmp-policy1-1] quit
[RouterA] interface gigabitethernet 1/0/0
[RouterA-GigabitEthernet1/0/0] ipsec policy policy1
[RouterA-GigabitEthernet1/0/0] quit
```

配置 RouterC。

```
[RouterC] acl number 3000
[RouterC-acl-adv-3000] rule permit gre source 30.1.1.2 0 destination 20.1.1.1 0
[RouterC-acl-adv-3000] quit
[RouterC] ipsec proposal p1
[RouterC-ipsec-proposal-p1] quit
[RouterC] ipsec policy policy1 1 isakmp
[RouterC-ipsec-policy-isakmp-policy1-1] security acl 3000
[RouterC-ipsec-policy-isakmp-policy1-1] ike-peer RouterA
[RouterC-ipsec-policy-isakmp-policy1-1] proposal p1
[RouterC-ipsec-policy-isakmp-policy1-1] quit
[RouterC] interface gigabitethernet 1/0/0
[RouterC-GigabitEthernet1/0/0] ipsec policy policy1
[RouterC-GigabitEthernet1/0/0] quit
```

完成此步骤后，可以利用通过 IPSec 加密的 GRE 隧道在 RouterA 和 RouterC 之间传输组播数据。

6. 在隧道的源端设备和目的端设备上配置 Tunnel 转发路由

配置 RouterA。

```
[RouterA] ip route-static 10.2.1.0 255.255.255.0 tunnel 0/0/1
```

配置 RouterC。

```
[RouterC] ip route-static 10.1.1.0 255.255.255.0 tunnel 0/0/1
```

7. 检查配置结果

PC1 和 PC2 相互 Ping 通后，在路由器上看到 IKE 协商已建立，IPSec 加密已生效。

```
[RouterA] display ike sa
   Conn-ID  Peer            VPN   Flag(s)                 Phase
  -----------------------------------------------------------------
       16    30.1.1.2        0     RD                      1
       17    30.1.1.2        0     RD                      2

  Flag Description:
  RD--READY   ST--STAYALIVE   RL--REPLACED   FD--FADING   TO--TIMEOUT
  HRT--HEARTBEAT   LKG--LAST KNOWN GOOD SEQ NO.   BCK--BACKED UP
[RouterA] display ips sa
==============================
Interface: GigabitEthernet1/0/0
    path MTU: 1500
==============================
  ------------------------------
  IPsec policy name: "policy1"
  sequence number: 1
```

```
  mode: isakmp
  -----------------------------
    connection id: 17
    encapsulation mode: tunnel
    tunnel local : 20.1.1.1    tunnel remote: 30.1.1.2
    [inbound ESP SAs]
      spi: 2970386335 (0xb10c7f9f)
      proposal: ESP-ENCRYPT-DES ESP-AUTH-MD5
      sa remaining key duration (bytes/sec): 1887434624/3081
      max received sequence-number: 32
      udp encapsulation used for nat traversal: N
    [outbound ESP SAs]
      spi: 1720763150 (0x6690c30e)
      proposal: ESP-ENCRYPT-DES ESP-AUTH-MD5
      sa remaining key duration (bytes/sec): 1887434112/3081
      max sent sequence-number: 33
      udp encapsulation used for nat traversal: N
[RouterC] display ike sa[RouterA] display ike sa
    Conn-ID  Peer            VPN   Flag(s)                 Phase
  -------------------------------------------------------------  ----
       20    20.1.1.2        0     RD|ST                     1
       21    20.1.1.2        0     RD|ST                     2

  Flag Description:
  RD--READY   ST--STAYALIVE   RL--REPLACED   FD--FADING   TO--TIMEOUT
  HRT--HEARTBEAT   LKG--LAST KNOWN GOOD SEQ NO.   BCK--BACKED UP
[RouterC] display ips sa
==============================
Interface: GigabitEthernet1/0/0
    path MTU: 1500
==============================
  -----------------------------
  IPsec policy name: "policy1"
  sequence number: 1
  mode: isakmp
  -----------------------------
    connection id: 21
    encapsulation mode: tunnel
    tunnel local : 30.1.1.2    tunnel remote: 20.1.1.1
    [inbound ESP SAs]
      spi: 1720763150 (0x6690c30e)
      proposal: ESP-ENCRYPT-DES ESP-AUTH-MD5
      sa remaining key duration (bytes/sec): 1887434624/3041
      max received sequence-number: 32
      udp encapsulation used for nat traversal: N
    [outbound ESP SAs]
      spi: 2970386335 (0xb10c7f9f)
      proposal: ESP-ENCRYPT-DES ESP-AUTH-MD5
      sa remaining key duration (bytes/sec): 1887434112/3041
      max sent sequence-number: 33
      udp encapsulation used for nat traversal: N
```

配置文件

- RouterA 的配置

```
#
 sysname RouterA
#
 ike local-name rta
#
 multicast routing-enable
#
acl number 3000
 rule 5 permit gre source 20.1.1.1 0.0.0.0 destination 30.1.1.2 0.0.0.0
#
ike peer routerc v1
 exchange-mode aggressive
 pre-shared-key 12345
 local-id-type name
```

```
 remote-name rtc
 remote-address 30.1.1.2
#
ipsec proposal p1
#
ipsec policy policy1 1 isakmp
 security acl 3000
 ike-peer RouterC
 proposal p1
#
interface GigabitEthernet1/0/0
ip address 20.1.1.1 255.255.255.0
ipsec policy policy1
#
interface GigabitEthernet2/0/0
 ip address 10.1.1.2 255.255.255.0
 pim dm
 igmp enable
#
interface Tunnel0/0/1
 ip address 40.1.1.1 255.255.255.0
 tunnel-protocol gre
 source 20.1.1.1
 destination 30.1.1.2
 pim dm
#
ospf 1
 area 0.0.0.0
  network 20.1.1.1 0.0.0.0
#
ip route-static 10.2.1.0 255.255.255.0 Tunnel0/0/1
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface GigabitEthernet1/0/0
ip address 20.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
ip address 30.1.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 20.1.1.0 0.0.0.255
  network 30.1.1.0 0.0.0.255
#
return
```

- RouterC 的配置文件

```
#
 sysname RouterC
#
 ike local-name rtc
#
 multicast routing-enable
#
acl number 3000
 rule 5 permit gre source 30.1.1.2 0.0.0.0 destination 20.1.1.1 0.0.0.0
#
ike peer routera v1
 exchange-mode aggressive
 pre-shared-key 12345
 local-id-type name
 remote-name rta
 remote-address 20.1.1.1
#
ipsec proposal p1
```

```
#
ipsec policy policy1 1 isakmp
 security acl 3000
 ike-peer RouterA
 proposal p1
#
interface GigabitEthernet1/0/0
ip address 30.1.1.2 255.255.255.0
ipsec policy policy1
#
interface GigabitEthernet2/0/0
 ip address 10.2.1.2 255.255.255.0
 pim dm
 igmp enable
#
interface Tunnel0/0/1
 ip address 40.1.1.2 255.255.255.0
 tunnel-protocol gre
 source 30.1.1.2
 destination 20.1.1.1
 pim dm
#
ospf 1
 area 0.0.0.0
  network 30.1.1.2 0.0.0.0
#
ip route-static 10.1.1.0 255.255.255.0 Tunnel0/0/1
#
return
```

1.7.4 配置 CE 使用公网 GRE 隧道接入 VPN 示例

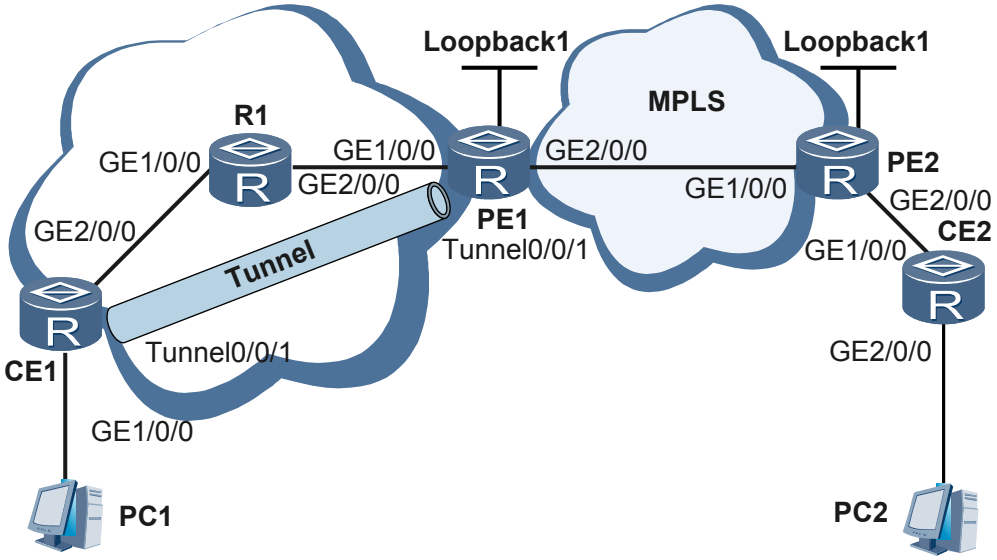
以典型组网为背景，介绍如何配置 CE 使用公网 GRE 隧道接入 VPN。PE 上没有与 CE 直连的接口，无法将 VPN 实例与物理接口进行绑定。通过在 CE 与 PE 之间建立一条 GRE 隧道穿越公网，在 PE 上将 VPN 与 GRE 隧道绑定，实现 CE 通过 GRE 隧道接入到 VPN 中。

组网需求

如图 1-8 所示：

- 路由器 PE1 和 PE2 位于 MPLS 骨干网。
- CE1 和 PE1 之间为公共网络，其间通过设备 R1 互连。
- CE2 与 PE2 直连。
- CE1 与 CE2 属于同一个 VPN，要求它们之间能互通。

图 1-8 配置 CE 使用公网 GRE 隧道接入 VPN 组网图



路由器	接口	IP 地址
CE1	GE1/0/0	21.1.1.2/24
CE1	GE2/0/0	30.1.1.1/24
CE1	Tunnel0/0/1	2.2.2.1/24
R1	GE1/0/0	30.1.1.2/24
R1	GE2/0/0	50.1.1.1/24
PE1	Loopback1	1.1.1.9/32
PE1	GE1/0/0	50.1.1.2/24
PE1	GE2/0/0	110.1.1.1/24
PE1	Tunnel0/0/1	2.2.2.2/24
PE2	Loopback1	3.3.3.9/32
PE2	GE1/0/0	110.1.1.2/24
PE2	GE2/0/0	11.1.1.2/24
CE2	GE1/0/0	11.1.1.1/24
CE2	GE2/0/0	41.1.1.2/24

配置思路

PE1 上没有与 CE1 直连的接口，无法将 VPN 实例与物理接口进行绑定。因此，在 CE1 与 PE1 之间建立一条 GRE 隧道，在 PE1 上将 VPN1 与 GRE 隧道绑定，实现 CE1 通过 GRE 隧道接入到 VPN 中。

配置 CE 使用公网 GRE 隧道接入 VPN 的思路如下：

1. 在骨干网设备 PE1 和 PE2 上运行 OSPF10 路由协议实现互通，并且使能 MPLS。
2. 在公网设备 CE1、R1 和 PE1 上运行 OSPF20 路由协议实现互通。
3. 在 CE1 和 PE1 之间建立 GRE 隧道。
4. 在 PE1 和 PE2 上建立 VPN 实例，并在 PE1 上将 VPN 实例与 GRE 隧道接口进行绑定，在 PE2 上将 VPN 实例与连接 CE2 的物理接口绑定。
5. 在 CE1 和 CE2 上配置到达各自连接 PE 的路由，这里使用 IS-IS。

6. 在 PE 之间配置 BGP，完成 CE1 和 CE2 之间的互通。

数据准备

为完成此配置例，需准备如下的数据：

- 接口 IP 地址、路由协议进程号及 AS 号
- GRE 隧道的源地址及目的地址
- 在 PE 创建的 VPN 实例名称、RD 和 VPN-target

操作步骤

步骤 1 配置各接口 IP 地址及 MPLS 骨干网的路由协议

在 PE1 和 PE2 之间运行 OSPF10，配置 MPLS、LDP，具体配置过程略。

步骤 2 配置公网 CE1、R1 和 PE1 之间的路由协议

在 CE1、R1 和 PE1 之间运行 OSPF20，具体配置过程略。

步骤 3 配置 CE1 和 PE1 之间的 GRE 隧道

在 CE1 上配置。

```
[CE1] interface tunnel0/0/1
[CE1-Tunnel0/0/1] ip address 2.2.2.1 255.255.255.0
[CE1-Tunnel0/0/1] tunnel-protocol gre
[CE1-Tunnel0/0/1] source 30.1.1.1
[CE1-Tunnel0/0/1] destination 50.1.1.2
[CE1-Tunnel0/0/1] quit
```

在 PE1 上配置。

```
[PE1] interface tunnel0/0/1
[PE1-Tunnel0/0/1] ip address 2.2.2.2 255.255.255.0
[PE1-Tunnel0/0/1] tunnel-protocol gre
[PE1-Tunnel0/0/1] source 50.1.1.2
[PE1-Tunnel0/0/1] destination 30.1.1.1
[PE1-Tunnel0/0/1] quit
```

上述配置完成后，CE1 与 PE1 之间的 GRE 隧道已建立。

步骤 4 在 PE1 创建 VPN 实例 vpn1 并与 GRE 隧道绑定

```
[PE1]ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] route-distinguisher 100:1
[PE1-vpn-instance-vpn1-af-ipv4] vpn-target 111:1 export-extcommunity
[PE1-vpn-instance-vpn1-af-ipv4] vpn-target 111:1 import-extcommunity
[PE1-vpn-instance-vpn1-af-ipv4] quit
[PE1-vpn-instance-vpn1] quit
[PE1] interface tunnel0/0/1
[PE1-Tunnel0/0/1] ip binding vpn-instance vpn1
[PE1-Tunnel0/0/1] ip address 2.2.2.2 255.255.255.0
[PE1-Tunnel0/0/1] quit
```

步骤 5 在 PE2 创建 VPN 实例 vpn1 并与 GE 接口绑定

```
[PE2]ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] route-distinguisher 200:1
[PE2-vpn-instance-vpn1-af-ipv4] vpn-target 111:1 export-extcommunity
[PE2-vpn-instance-vpn1-af-ipv4] vpn-target 111:1 import-extcommunity
[PE2-vpn-instance-vpn1-af-ipv4] quit
[PE2-vpn-instance-vpn1] quit
[PE2] interface gigabitethernet2/0/0
[PE2- GigabitEthernet2/0/0] ip binding vpn-instance vpn1
```

```
[PE2- GigabitEthernet2/0/0] ip address 11.1.1.2 255.255.255.0
[PE2- GigabitEthernet2/0/0] quit
```

步骤 6 配置 CE1 与 PE1 之间的 IS-IS 路由

在 CE1 上配置。

```
[CE1] isis 50
[CE1-isis-50] network-entity 50.0000.0000.0001.00
[CE1-isis-50] quit
[CE1] interface gigabitethernet1/0/0
[CE1-GigabitEthernet1/0/0] isis enable 50
[CE1-GigabitEthernet1/0/0] quit
[CE1] interface tunnel0/0/1
[CE1-Tunnel0/0/1] isis enable 50
[CE1-Tunnel0/0/1] quit
```

在 PE1 上配置。

```
[PE1] isis 50 vpn-instance vpn1
[PE1-isis-50] network-entity 50.0000.0000.0002.00
[PE1-isis-50] quit
[PE1] interface tunnel0/0/1
[PE1-Tunnel0/0/1] isis enable 50
[PE1-Tunnel0/0/1] quit
```

步骤 7 配置 CE2 与 PE2 之间的 IS-IS 路由

在 CE2 上配置。

```
[CE2] isis 50
[CE2-isis-50] network-entity 50.0000.0000.0004.00
[CE2-isis-50] quit
[CE2] interface gigabitethernet1/0/0
[CE2-GigabitEthernet1/0/0] isis enable 50
[CE2-GigabitEthernet1/0/0] quit
[CE2] interface gigabitethernet2/0/0
[CE2-GigabitEthernet2/0/0] isis enable 50
[CE2-GigabitEthernet2/0/0] quit
```

在 PE2 上配置。

```
[PE2] isis 50 vpn-instance vpn1
[PE2-isis-50] network-entity 50.0000.0000.0003.00
[PE2-isis-50] quit
[PE2] interface gigabitethernet2/0/0
[PE2-GigabitEthernet2/0/0] isis enable 50
[PE2-GigabitEthernet2/0/0] quit
```

步骤 8 在 PE 之间建立 MP-IBGP 对等体

配置 PE1，指定 PE2 为 IBGP 对等体，使用 loopback 接口建立 IBGP 连接，启动对等体交换 VPN-IPv4 路由信息。

```
[PE1] bgp 100
[PE1-bgp] peer 3.3.3.9 as-number 100
[PE1-bgp] peer 3.3.3.9 connect-interface loopback 1
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 3.3.3.9 enable
[PE1-bgp-af-vpnv4] quit
```

进入 BGP 的 vpn1 实例，引入直连路由和 IS-IS 路由。

```
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import-route direct
[PE1-bgp-vpn1] import-route isis 50
```

配置 PE2，指定 PE1 为 IBGP 对等体，使用 loopback 接口建立 IBGP 连接，启动对等体交换 VPN-IPv4 路由信息。

```
[PE2] bgp 100
[PE2-bgp] peer 1.1.1.9 as-number 100
[PE2-bgp] peer 1.1.1.9 connect-interface loopback 1
[PE2-bgp] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 1.1.1.9 enable
[PE2-bgp-af-vpnv4] quit
```

进入 BGP 的 vpn1 实例，引入直连路由和 IS-IS 路由。

```
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] import-route direct
[PE2-bgp-vpn1] import-route isis 50
```

步骤 9 在 IS-IS 中引入 BGP 路由

配置 PE1。

```
[PE1] isis 50
[PE1-isis-50] import-route bgp
```

配置 PE2。

```
[PE2] isis 50
[PE2-isis-50] import-route bgp
```

步骤 10 检查配置结果

以上配置完成后，CE1 和 CE2 之间可以相互 ping 通。

```
<CE1> ping 41.1.1.2
  PING 41.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 41.1.1.2: bytes=56 Sequence=1 ttl=253 time=190 ms
    Reply from 41.1.1.2: bytes=56 Sequence=2 ttl=253 time=110 ms
    Reply from 41.1.1.2: bytes=56 Sequence=3 ttl=253 time=110 ms
    Reply from 41.1.1.2: bytes=56 Sequence=4 ttl=253 time=110 ms
    Reply from 41.1.1.2: bytes=56 Sequence=5 ttl=253 time=100 ms
  --- 41.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 100/124/190 ms
<CE2> ping 21.1.1.2
  PING 21.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 21.1.1.2: bytes=56 Sequence=1 ttl=253 time=120 ms
    Reply from 21.1.1.2: bytes=56 Sequence=2 ttl=253 time=110 ms
    Reply from 21.1.1.2: bytes=56 Sequence=3 ttl=253 time=120 ms
    Reply from 21.1.1.2: bytes=56 Sequence=4 ttl=253 time=90 ms
    Reply from 21.1.1.2: bytes=56 Sequence=5 ttl=253 time=60 ms
  --- 21.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 60/100/120 ms
```

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
isis 50
 network-entity 50.0000.0000.0001.00
#
interface GigabitEthernet1/0/0
 ip address 21.1.1.2 255.255.255.0
 isis enable 50
```

```
#
interface GigabitEthernet2/0/0
ip address 30.1.1.1 255.255.255.0
#
interface Tunnel0/0/1
 ip address 2.2.2.1 255.255.255.0
 tunnel-protocol gre
 source 30.1.1.1
 destination 50.1.1.2
 isis enable 50
#
ospf 20
 area 0.0.0.0
  network 30.1.1.0 0.0.0.255
#
return
```

- R1 的配置文件

```
#
 sysname R1
#
interface GigabitEthernet1/0/0
ip address 30.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
ip address 50.1.1.1 255.255.255.0
#
ospf 20
 area 0.0.0.0
  network 30.1.1.0 0.0.0.255
  network 50.1.1.0 0.0.0.255
#
return
```

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpn1
 route-distinguisher 100:1
 vpn-target 111:1 export-extcommunity
 vpn-target 111:1 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
  lsp-trigger all
#
mpls ldp
#
isis 50 vpn-instance vpn1
 network-entity 50.0000.0000.0002.00
 import-route bgp
#
interface GigabitEthernet1/0/0
ip address 50.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
ip address 110.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
interface Tunnel0/0/1
 ip binding vpn-instance vpn1
 ip address 2.2.2.2 255.255.255.0
 tunnel-protocol gre
 source 50.1.1.2
 destination 30.1.1.1
```

```
 isis enable 50
#
bgp 100
 peer 3.3.3.9 as-number 100
 peer 3.3.3.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 3.3.3.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 3.3.3.9 enable
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  import-route isis 50
#
ospf 10
 area 0.0.0.0
  network 1.1.1.9 0.0.0.0
  network 110.1.1.0 0.0.0.255
#
ospf 20
 area 0.0.0.0
  network 50.1.1.0 0.0.0.255
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpn1
 route-distinguisher 200:1
 vpn-target 111:1 export-extcommunity
 vpn-target 111:1 import-extcommunity
#
 mpls lsr-id 3.3.3.9
 mpls
  lsp-trigger all
#
mpls ldp
#
isis 50 vpn-instance vpn1
 network-entity 50.0000.0000.0003.00
 import-route bgp
#
interface GigabitEthernet1/0/0
ip address 110.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
ip binding vpn-instance vpn1
 ip address 11.1.1.2 255.255.255.0
 isis enable 50
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
#
bgp 100
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
 #
 ipv4-family vpnv4
```

```
  policy vpn-target
  peer 1.1.1.9 enable
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  import-route isis 50
#
ospf 10
 area 0.0.0.0
  network 3.3.3.9 0.0.0.0
  network 110.1.1.0 0.0.0.255
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
isis 50
 network-entity 50.0000.0000.0004.00
#
interface GigabitEthernet1/0/0
ip address 11.1.1.1 255.255.255.0
 isis enable 50
#
interface GigabitEthernet2/0/0
 ip address 41.1.1.2 255.255.255.0
 isis enable 50
#
return
```

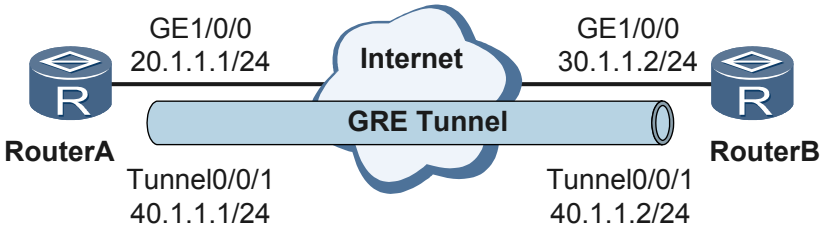
1.7.5 配置 GRE 支持 Keepalive 特性示例

以典型组网为背景，介绍如何配置 GRE 支持 Keepalive 特性，防止 VPN 选择对端不可达的 GRE 隧道，避免造成数据丢失。

组网需求

如图 1-9 所示，RouterA 和 RouterB 端配置了 GRE 隧道协议，要求 GRE 隧道的两端具备 Keepalive 功能。

图 1-9 配置 GRE 支持 Keepalive 特性组网图



配置思路

使本端的 GRE 具有 Keepalive 特性，只需在本端路由器的隧道接口视图下键入 **keepalive** 命令。

窍门

源端实现 Keepalive 功能并不要求对端也具备 Keepalive 功能，对端只需实现转发功能。

数据准备

为完成此配置例，需准备如下的数据：

- 配置骨干网路由协议所需数据
- GRE 两端的源地址、目的地址和 Tunnel 接口 IP 地址
- 发送 keepalive 报文的周期
- 不可达计数器参数

操作步骤

步骤 1 实现 RouterA 与 RouterB 的互通

具体配过程略。

步骤 2 在 Router A 上配置隧道，使能 Keepalive

```
<RouterA> system-view
[RouterA] interface tunnel 0/0/1
[RouterA-Tunnel0/0/1] ip address 40.1.1.1 255.255.255.0
[RouterA-Tunnel0/0/1] tunnel-protocol gre
[RouterA-Tunnel0/0/1] source 20.1.1.1
[RouterA-Tunnel0/0/1] destination 30.1.1.2
[RouterA-Tunnel0/0/1] keepalive period 20 retry-times 3
[RouterA-Tunnel0/0/1] quit
```

步骤 3 在 Router B 上配置隧道，使能 Keepalive

```
<RouterB> system-view
[RouterB] interface tunnel 0/0/1
[RouterB-Tunnel0/0/1] ip address 40.1.1.2 255.255.255.0
[RouterB-Tunnel0/0/1] tunnel-protocol gre
[RouterB-Tunnel0/0/1] source 30.1.1.2
[RouterB-Tunnel0/0/1] destination 20.1.1.1
[RouterB-Tunnel0/0/1] keepalive period 20 retry-times 3
[RouterB-Tunnel0/0/1] quit
```

步骤 4 检查配置结果

从 Router A 上的 Tunnel 接口应该能 ping 通 Router B 的 Tunnel 接口。

```
<RouterA> ping -a 40.1.1.1 40.1.1.2
  PING 40.1.1.2: 56  data bytes, press CTRL_C to break
    Reply from 40.1.1.2: bytes=56 Sequence=1 ttl=255 time=9 ms
    Reply from 40.1.1.2: bytes=56 Sequence=2 ttl=255 time=7 ms
    Reply from 40.1.1.2: bytes=56 Sequence=3 ttl=255 time=7 ms
    Reply from 40.1.1.2: bytes=56 Sequence=4 ttl=255 time=7 ms
    Reply from 40.1.1.2: bytes=56 Sequence=5 ttl=255 time=7 ms
  --- 40.1.1.2 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 7/7/9 ms
```

打开 Router A 的调试开关查看 Keepalive 报文的消息。

```
<RouterA> terminal monitor
<RouterA> terminal debugging
<RouterA> debugging tunnel keepalive
May 18 2011 11:36:11.590.1+00:00 AR1220 TUNNEL/7/debug:GRE_KEEP:Judge keepalive
finished. Received keepalive detecting packet from peer router.
<RouterA>
May 18 2011 11:36:11.590.2+00:00 AR1220 TUNNEL/7/debug:GRE_KEEP_NSR: Mainboard u
lKeepaliveReceiveOpposite++ then send mbuf to slave when RECEIVE keepalive packe
t.
<RouterA>
```

```
May 18 2011 11:36:11.590.3+00:00 AR1220 TUNNEL/7/debug:GRE_FWD: Receive peer kee
palive on mainboard successfully. Put into decapsulation.
<RouterA>
May 18 2011 11:36:15.120.1+00:00 AR1220 TUNNEL/7/debug:GRE_KEEP:Judge keepalive
finished. Received keepalive response packet from peer router.
<RouterA>
May 18 2011 11:36:15.120.2+00:00 AR1220 TUNNEL/7/debug:GRE_FWD: Receive the resp
onse keepalive packet on mainboard successfully, keepalive finished.
<RouterA>
May 18 2011 11:36:15.120.3+00:00 AR1220 TUNNEL/7/debug:GRE_KEEP_NSR: Mainboard s
end mbuf to slaveboard when RECEIVE response packet.
```

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
interface GigabitEthernet1/0/0
 ip address 20.1.1.1 255.255.255.0
#
interface Tunnel0/0/1
 ip address 40.1.1.1 255.255.255.0
 tunnel-protocol gre
 source 20.1.1.1
 destination 30.1.1.2
 keepalive period 20
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
interface GigabitEthernet1/0/0
ip address 30.1.1.2 255.255.255.0
#
interface Tunnel0/0/1
 ip address 40.1.1.2 255.255.255.0
 tunnel-protocol gre
 source 30.1.1.2
 destination 20.1.1.1
 keepalive period 20
#
return
```

2 BGP/MPLS IP VPN 配置

关于本章

BGP/MPLS IP VPN 配置包括 VPN 概述、VPN 常见组网的配置，VPN 可靠性配置。

2.1 BGP/MPLS IP VPN 概述
介绍 PE、P 和 CE 设备的概念和作用。

2.2 AR1200 支持的 BGP/MPLS IP VPN
AR1200 支持的 BGP/MPLS IP VPN 主要包括 BGP/MPLS IP VPN 的基本组网、典型组网、可靠性和 QoS 等特性。

2.3 配置使能 IPv4 地址族的 VPN 实例
VPN 实例的作用是将 VPN 私网路由与公网路由隔离。配置使能 IPv4 地址族的 VPN 实例，使 PE 设备可以通告 IPv4 路由和转发数据。

2.4 配置基本 BGP/MPLS IP VPN
基本 BGP/MPLS IP VPN 是指：只包括一个运营商、MPLS 骨干网不跨域，PE、P、CE 设备不兼任其它功能（没有一台设备既是 PE，又是 CE）。

2.5 配置 Hub and Spoke
Hub and Spoke 组网是通过在 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。

2.6 配置跨域 VPN-OptionA
跨域 VPN-OptionA 中，ASBR 把对端 ASBR 看作自己的 CE 设备，使用 EBGP 方式向对端发布 VPNv4 路由。

2.7 配置跨域 VPN-OptionB
跨域 VPN-OptionB 中，ASBR 之间通过 MP-EBGP 交换它们从各自 AS 的 PE 设备接收的 VPNv4 路由。

2.8 配置跨域 VPN-OptionC（方案一）
不同 AS 的 PE 之间建立 Multihop 方式的 EBGP 连接，交换 VPNv4 路由。

2.9 配置跨域 VPN-OptionC（方案二）
通过为带标签的公网 BGP 路由建立 LDP LSP，在不同 AS 的 PE 之间建立 Multihop 方式的 EBGP 连接，交换 VPNv4 路由。

2.10 配置 HoVPN

HoVPN 是具有层次化的 VPN 网络，由多个 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，以降低对 PE 设备的性能要求。

2.11 配置 Multi-VPN-Instance CE
通过在 CE 上配置 OSPF 多实例实现局域网不同业务的隔离。

2.12 配置 VPN 与 Internet 互联
一般 VPN 内的用户只能相互通信，不能与 Internet 用户通信，也不能接入 Internet。如果 VPN 的各个 site 需要访问 internet，需要配置 VPN 与 Internet 互联。

2.13 配置私网 IP FRR
VPN site 中的多个 CE 接入到同一台 PE 上时，配置 IP FRR 特性，当 PE 与 CE 之间转发不通时，可以快速将流量切换到另一条 PE 与 CE 相连的链路上。

2.14 配置 VPN FRR
在 CE 多归属组网中，配置 VPN FRR 可以保证 PE 设备发生故障时实现 VPN 业务端到端的快速切换。

2.15 配置路由反射器优化 VPN 骨干层
使用路由反射器，可以减少 PE 之间的 MP-IBGP 连接的数量，既减轻了 PE 的负担，也给维护和管理带来方便。

2.16 配置路由反射器优化 VPN 接入层
当 PE 及其接入的多个 CE 位于同一个 AS 时，部署 BGP 路由反射器，可以减少 CE 之间的 IBGP 连接的数量，给维护和管理带来方便。

2.17 维护 BGP/MPLS IP VPN
维护 BGP/MPLS IP VPN 包括查看 L3VPN 流量、监测网络连通性、复位 BGP 连接。

2.18 配置举例
介绍 VPN 各种组网的配置举例。配置示例中包括组网需求、配置注意事项、配置思路、配置过程和配置文件。

2.1 BGP/MPLS IP VPN 概述

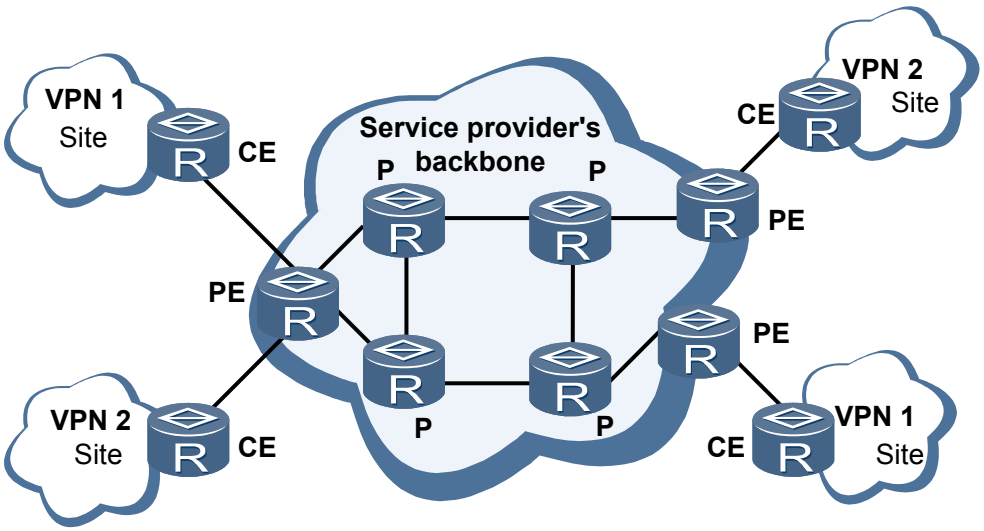
介绍 PE、P 和 CE 设备的概念和作用。

BGP/MPLS IP VPN 是提供商 VPN 解决方案 PPVPN（Provider Provisioned VPN）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。

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

图 2-1 是 BGP/MPLS IP VPN 模型的示意图：

图 2-1 BGP/MPLS IP VPN 模型



BGP/MPLS IP VPN 模型由三部分组成：CE、PE 和 P。

- CE（Customer Edge）：用户网络边缘设备，有接口直接与服务提供商 SP（Service Provider）网络相连。CE 可以是路由器，或交换机，也可以是一台主机。通常情况下，CE“感知”不到 VPN 的存在，也不需要支持 MPLS。
- PE（Provider Edge）：服务提供商边缘设备，是服务提供商网络的边缘设备，与 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）：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，不维护 VPN 信息。

2.2 AR1200 支持的 BGP/MPLS IP VPN

AR1200 支持的 BGP/MPLS IP VPN 主要包括 BGP/MPLS IP VPN 的基本组网、典型组网、可靠性和 QoS 等特性。

基本组网

AR1200 使用 MP-BGP 在 PE 之间的 VPN 路由交换，PE 和 CE 间的路由交换可以采用静态路由、RIP 多实例、OSPF 多实例、IS-IS 多实例，也可以采用 EBGP；AR1200 使用

VPN-Target 属性控制 VPN 路由的收发，从而实现多种 VPN 组网拓扑，如 Intranet、Extranet 和Hub&Spoke。

一般使用 LSP 隧道作为 VPN 骨干网隧道；如果 PE 具备 MPLS 功能，但 P 设备只提供纯 IP 功能不具备 MPLS 功能，则可使用 GRE。

典型组网

AR1200 实现以下典型 VPN 组网：

- 跨域 VPN

  如果 VPN 骨干网跨越多个 AS，则需要部署跨域 VPN。有三种模式实现跨域 VPN，分别为 OptionA 方式、OptionB 方式和 OptionC 方式。

- HoVPN（Hierarchy of VPN）

  为了减轻 PE 设备的负担，可部署 HoVPN，选择汇聚层或接入层的设备作为 UPE，与骨干层的 PE（作为 SPE）共同完成一个 PE 的功能。

- Multi-VPN-Instance CE

  目前，局域网不同业务的隔离一般是通过交换机的 VLAN 功能实现的，但交换机的路由功能相对较弱。为了保证局域网的安全隔离并提高局域网的路由能力，可部署 Multi-VPN-Instance CE 特性，以较低的成本解决局域网的安全问题。

- VPN 与 Internet 互联

  AR1200 支持 VPN 与 Internet 互联的功能。本章介绍在 PE 侧通过配置静态路由和策略路由实现 VPN 与 Internet 互联。

可靠性

为了提高 VPN 网络的可靠性，通常采用如下组网模型：

- 骨干层采用全连接，多级备份的 MPLS 网络。各设备一般使用高速接口互连。当 PE 设备较多时，采用 BGP 路由反射器组网，反射 VPNv4 路由，以减少 MP-IBGP 连接数目。
- 汇聚层根据需要组成网状或环状网。

2.3 配置使能 IPv4 地址族的 VPN 实例

VPN 实例的作用是将 VPN 私网路由与公网路由隔离。配置使能 IPv4 地址族的 VPN 实例，使 PE 设备可以通告 IPv4 路由和转发数据。

2.3.1 建立配置任务

在配置 VPN 实例前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

BGP/MPLS IP VPN 为每个 VPN 进行实例化，构建本 VPN 私有转发信息实例，即 VPN 实例（VPN-instance）。VPN 实例也称为 VPN 路由转发表 VRF（VPN Routing and Forwarding table）。在 RFC4364（BGP/MPLS IP VPNs）中，VPN 实例被称为 per-site forwarding table。

VPN 实例用于将 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。在所有 BGP/MPLS IP VPN 组网方案中，都需要配置 VPN 实例。

VPN 实例 IPv4 地址族通过 RD 实现地址空间独立，通过 VPN Target 属性实现对直连 site 的 VPN 成员关系和路由规则控制。

在通过 VPN Target 属性控制 VPN 路由收发的同时，如果需要更精确地控制 VPN 路由，还可以使用入方向或出方向路由策略。入方向路由策略可以过滤引入到 VPN 实例 IPv4 地址族的路由信息，而出方向路由策略则可以进一步过滤可发布给其他 PE 的路由。

前置任务

配置使能 IPv4 地址族的 VPN 实例之前，需完成以下任务：

- 如果对 VPN 实例 IPv4 地址族应用出或入方向路由策略，需配置路由策略。
- 如果 VPN 实例 IPv4 地址族进行隧道负载分担，或者要改变 LSPGRE 隧道的默认选择顺序时，需配置隧道策略。

数据准备

在配置 VPN 实例之前，需准备以下数据。

序号	数据
1	VPN 实例的名称
2	（可选）VPN 实例的描述信息
3	VPN 实例 IPv4 地址族的 RD、VPN Target 属性
4	（可选）VPN 实例 IPv4 地址族中允许的最大路由数
5	（可选）控制 VPN 路由信息收发的路由策略
6	（可选）隧道策略

2.3.2 创建 VPN 实例

所有涉及到 VPN 的配置都需要配置 VPN 实例，其作用是创建 VPN 私网路由表和转发表。

背景信息

在接入 CE 的 PE 设备上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip vpn-instance** *vpn-instance-name*，创建 VPN 实例，并进入 VPN 实例视图。

说明

VPN 实例的名字区分大小写。例如，“vpn1”和“VPN1”将被认为是不同的 VPN 实例。

PE 上没有缺省的 VPN 实例，一个 PE 上可以创建多个 VPN 实例。

步骤 3 （可选）执行命令 **description** *description-information*，配置 VPN 实例的描述信息。

描述信息的作用类似于主机名和接口描述信息，建议用户选择合适的描述信息进行配置。

步骤 4 （可选）执行命令 **service-id** *service-id*，配置 VPN 实例的业务标识值。

业务标识值用来区别网络上不同的 VPN 服务，在同一台设备上具有唯一性。

----结束

2.3.3 配置 VPN 实例 IPv4 地址族的相关属性

为方便地管理 VPN 实例 IPv4 地址族的路由，还需要配置相关属性，比如：VPN-Target、路由限制、路由策略等。

背景信息

在配置了 VPN 实例的 PE 设备上进行如下配置。

说明

建议第 6 步和第 7 步只选其中之一配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip vpn-instance** *vpn-instance-name*，进入 VPN 实例视图。

步骤 3 执行命令 **ipv4-family**，使能 VPN 实例 IPv4 地址族，并进入 VPN 实例 IPv4 地址族视图。

步骤 4 执行命令 **route-distinguisher** *route-distinguisher*，配置 VPN 实例 IPv4 地址族的 RD。

VPN 实例 IPv4 地址族只有配置了 RD 后才生效。同一 PE 上的不同 VPN 实例 IPv4 地址族下的 RD 不能相同。

说明

RD 配置后不能被修改或删除。如果要修改 RD 或删除 RD，需要先删除对应的 VPN 实例或者去使能 VPN 实例 IPv4 地址族。

步骤 5 执行命令 **vpn-target** *vpn-target* &<1-8> [**both** | **export-extcommunity** | **import-extcommunity**]，为 VPN 实例 IPv4 地址族配置 VPN-target 扩展团体属性。

VPN Target 是 BGP 的扩展团体属性，用来控制 VPN 路由信息的接收和发布。一条 **vpn-target** 命令最多可以配置 8 个 VPN Target。

步骤 6 （可选）执行命令 **routing-table limit** *number* { *alert-percent* | **simply-alert** }，配置 VPN 实例 IPv4 地址族的最大路由数。

为防止 PE 设备从 CE 引入的路由数量过多，可以配置一个 VPN 实例能够支持的最大路由数。

说明

配置了 **routing-table limit** 命令，当注入到 VPN 实例 IPv4 地址族路由表的路由超限时，系统会给出提示信息。执行 **routing-table limit** 命令增大 VPN 实例 IPv4 地址族下支持的最大路由数或者执行 **undo routing-table limit** 命令取消路由表限制后，对于这些超限的路由，还要进行如下处理：

- 对于超限的静态路由，需要手动重新配置。
- 通过 IGP 多实例路由协议从 CE 学到的路由，需要在 PE 上重启路由协议的多实例进程。

通过 MP-IBGP 学到的远端交叉路由和从 CE 上学来的 BGP 路由，系统可以自动刷新。

步骤 7 （可选）执行命令 **prefix limit** *number* { *alert-percent* [**route-unchanged**] | **simply-alert** }，配置 VPN 实例 IPv4 地址族的最大路由前缀数。

为防止 PE 设备从 CE 引入的路由前缀数量过多，可以配置一个 VPN 实例 IPv4 地址族能够支持的最大路由前缀数。

步骤 8 （可选）执行命令 **limit-log-interval** *interval*，配置 VPN 实例 IPv4 地址族的路由超出限制后输出日志的频率。

步骤 9 （可选）执行命令 **import route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族入方向路由策略。

步骤 10 （可选）执行命令 **export route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族出方向路由策略。

----结束

2.3.4 （可选）配置基于 VPN 实例 IPv4 地址族分配 MPLS 标签

配置基于 VPN 实例 IPv4 地址族分配 MPLS 标签，本端 PE 设备将为 VPN 实例 IPv4 地址族中的所有路由分配一个标签。当 VPN 路由数量比较多时，可以降低 PE 设备对 MPLS 标签的维护量。

背景信息

在配置了 VPN 实例 IPv4 地址族的 PE 设备上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip vpn-instance** *vpn-instance-name*，进入 VPN 实例视图。

步骤 3 执行命令 **ipv4-family**，使能 VPN 实例 IPv4 地址族，进入 VPN 实例 IPv4 地址族视图。

步骤 4 执行命令 **apply-label per-instance**，配置基于 VPN 实例 IPv4 地址族分配 MPLS 标签，使 VPN 实例中的所有路由都使用同一个标签。

通常情况下，标签的分配方式是每条路由一个标签（one label per route）。当路由数量比较多时，占用的标签资源会增多。

AR1200 实现基于 VPN 实例的 MPLS 标签分配，即，VPN 实例 IPv4 地址族中的所有路由都使用同一个标签。

----结束

2.3.5 检查配置结果

配置了 VPN 实例后，可以看到本设备上创建的 VPN 实例 IPv4 地址族的信息，包括 RD 值及其相关属性。

前提条件

已经完成 VPN 实例功能的所有配置。

操作步骤

- 使用 **display ip vpn-instance verbose** *vpn-instance-name* 命令查看指定 VPN 实例的详细信息，包括 VPN 实例 IPv4 地址族下的信息。
- 使用 **display ip vpn-instance** *vpn-instance-name* 命令查看指定 VPN 实例的简要信息。
- 使用 **display ip vpn-instance import-vt** *ivt-value* 命令查看所有具备指定入口 vpn-target 属性的 VPN 实例信息。

----结束

任务示例

在配置成功时，执行命令 **display ip vpn-instance**，可以看到本设备上创建的 VPN 实例的简要信息，例如：

```
<Huawei> display ip vpn-instance
Total VPN-Instances configured : 4
 VPN-Instance Name               Address-family
 vrf1                            ipv4
 vrf2
 vrf3                            ipv4
 vrf4                            ipv4
```

配置成功时，执行命令 **display ip vpn-instance verbose**，可以看到本设备上创建的 VPN 实例的详细信息，例如：

```
<Huawei> display ip vpn-instance verbose
Total VPN-Instances configured : 1

VPN-Instance Name and ID : vpn1, 1
  Description : vrf1
  Service ID : 123
Address family ipv4
  Create date : 2010/03/05 16:26:27
  Up time : 0 days, 00 hours, 09 minutes and 12 seconds
  Route Distinguisher : 100:1
  Export VPN Targets :  1:1
  Import VPN Targets :  1:1
  Label Policy : label per instance
  Per-Instance Label : 1029
  Import Route Policy : rp1
  Export Route Policy : rp2
  Tunnel Policy : tp1
  Maximum Routes Limit : 200
  Threshold Routes Limit : 10%
  Prefix Routes Limit : 200
  Threshold Prefixes Limit : 20%
  Install Mode : route-unchanged
  Log Interval : 30
```

配置成功时，执行命令 **display ip vpn-instance import-vt** *ivt-value*，查看本设备上所有具备指定入口 vpn-target 属性的 VPN 实例信息。

```
<Huawei> display ip vpn-instance import-vt 1:1
```

```
The number of ipv4-family matched the import-vt : 3
 VPN-Instance Name and ID : vrf1, 1
 VPN-Instance Name and ID : vrf4, 5
 VPN-Instance Name and ID : vrf5, 4
```

2.4 配置基本 BGP/MPLS IP VPN

基本 BGP/MPLS IP VPN 是指：只包括一个运营商、MPLS 骨干网不跨域，PE、P、CE 设备不兼任其它功能（没有一台设备既是 PE，又是 CE）。

2.4.1 建立配置任务

在配置基本 BGP/MPLS IP VPN 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

本节介绍基本的 BGP/MPLS IP VPN 组网配置：只包括一个运营商、MPLS 骨干网不跨域，PE、P、CE 设备不兼任其它功能（没有一台设备既是 PE，又是 CE）。

如果需要部署一些特殊的 BGP/MPLS IP VPN 组网方案，例如 HoVPN、跨域 VPN 等，则还需要另外进行相关的配置，请参见本章中相应小节的介绍。

配置 BGP/MPLS IP VPN 的关键在于管理 VPN 路由在 MPLS 骨干网上的发布，包括 PE-PE 间的路由发布管理以及 PE-CE 间的路由发布管理。

PE-PE 间的路由交换采用 MP-IBGP。PE-CE 间的路由交换可以采用静态路由、RIP 多实例、OSPF 多实例、IS-IS 多实例，也可以采用 BGP。根据实际组网情况选择一种进行配置即可。

说明

如果一个 VPN 接收本 VPN 以外的、非 PE 发布的路由，并将这些路由发布给 PE，这类 VPN 称为过渡 VPN（transit VPN）。

只接收本 VPN 路由以及 PE 发布的路由的 VPN 称为 stub VPN。通常情况下，静态路由只用于 stub VPN 的 CE 与 PE 间交换路由。

前置任务

在配置基本 BGP/MPLS IP VPN 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 基本能力和 MPLS LDP
- 在 PE 之间根据隧道策略建立所需隧道
- 在 CE 上配置接入 PE 的接口的 IP 地址

数据准备

在配置基本 BGP/MPLS IP VPN 之前，需准备以下数据。

序号	数据
1	配置 VPN 实例所需的数据，包括： ● VPN 实例名称 ●（可选）VPN 实例的描述信息 ● VPN 实例 IPv4 地址族的 RD、VPN Target 属性 ●（可选）控制 VPN 路由信息收发的路由策略 ●（可选）隧道策略 ●（可选）VPN 实例 IPv4 地址族中允许的最大路由数
2	PE 上接入 CE 的接口的 IP 地址
3	PE-CE 间采用的路由交换方式：静态路由、RIP、OSPF、IS-IS 或 BGP
4	PE 的 AS 号
5	PE 用来建立 BGP 对等体的 IP 地址和接口

2.4.2 配置 VPN 实例

配置 VPN 实例，用以管理 VPN 路由。

操作步骤

步骤 1 VPN 实例的配置方法请参见**配置 VPN 实例**。

----结束

2.4.3 配置接口与 VPN 实例绑定

通过配置接口与 VPN 实例绑定，该接口成为私网接口，从该接口进入的报文使用 VPN 实例中的转发信息进行转发。同时将删除该接口上已经配置的 IP 地址、路由协议等三层特性，如果需要应重新配置。

背景信息

在每个接入 CE 的 PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入要绑定 VPN 实例的接口视图。

步骤 3 执行命令 **ip binding vpn-instance** *vpn-instance-name*，将当前接口与 VPN 实例绑定。

说明

执行 **ip binding vpn-instance** 命令将删除接口上已经配置的 IP 地址、路由协议等三层特性，如果需要应重新配置。

接口不能与未使能任何地址族的 VPN 实例绑定。

当去使能 VPN 实例的某地址族时，也将删除与 VPN 实例绑定接口下的 IP 地址、路由协议等三层特性。当去使能 VPN 实例下所有地址族时，去使能接口与该 VPN 实例的绑定。

步骤 4 执行命令 **ip address** *ip-address* { *mask* | *mask-length* }，配置接口的 IP 地址。

----结束

2.4.4 （可选）配置 BGP VPN 实例 IPv4 地址族 Router ID

通过为 BGP VPN 实例 IPv4 地址族配置 Router ID，使得同一台设备上不同 BGP VPN 实例 IPv4 地址族的 Router ID 不同。

背景信息

缺省情况下，BGP VPN 实例 IPv4 地址族下没有配置 Router ID，直接使用的是 BGP Router ID，这样同一台设备的不同 VPN 实例的 IPv4 地址族 Router ID 是相同的。而有些情形下，需要为不同的 VPN 实例 IPv4 地址族配置不同的 Router ID，例如：同一 PE 上不同 VPN 实例 IPv4 地址族需要建立 BGP 对等体。

为 BGP VPN 实例 IPv4 地址族配置 Router ID 有两种方式：

操作步骤

- 为所有的 BGP VPN 实例 IPv4 地址族配置 Router ID
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **router-id vpn-instance auto-select**，为所有的 BGP VPN 实例 IPv4 地址族配置自动选择 Router ID。

     说明

     为 BGP VPN 实例 IPv4 地址族自动选择 Router ID 的规则如下：

     - 如果使能 IPv4 地址族的 VPN 实例关联了配置 IP 地址的 Looback 接口，则选择 Loopback 接口地址中最大的 IP 地址作为 Router ID。
     - 如果使能 IPv4 地址族的 VPN 实例没有关联配置了 IP 地址的 Loopback 接口，则从 VPN 实例绑定的其他接口中选择最大的 IP 地址作为 Router ID（不考虑接口的 UP/DOWN 状态）。
- 为指定的 BGP VPN 实例 IPv4 地址族配置 Router ID
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP VPN 实例 IPv4 地址族视图。
  4. 执行命令 **router-id** { *ipv4-address* | **auto-select** }，为当前 BGP VPN 实例 IPv4 地址族配置 Router ID 或者自动选择 Router ID。

----结束

2.4.5 配置 PE 与 PE 间使用 MP-IBGP

MP-IBGP 通过在 BGP 中引入扩展团体属性，使其能够在 PE 设备之间传播 VPNv4 路由。

背景信息

在每个接入 CE 的 PE 上进行如下配置。

操作步骤

步骤 1 执行 **system-view**，进入系统视图。

步骤 2 执行 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行 **peer** *ipv4-address* **as-number** *as-number*，将对端 PE 配置为对等体。

步骤 4 执行 **peer** *ipv4-address* **connect-interface loopback** *interface-number*，指定建立 TCP 连接的接口。

说明

PE 之间必须使用 32 位掩码的 Loopback 接口地址来建立 MP-IBGP 对等体关系，以便能够迭代到隧道。到 Loopback 接口的路由通过 MPLS 骨干网上的 IGP 发布给对端 PE。

步骤 5 执行 **ipv4-family vpnv4**，进入 BGP-VPNv4 子地址族视图。

步骤 6 执行 **peer** *ipv4-address* **enable**，使能对等体交换 VPNv4 路由信息的能力。

----结束

2.4.6 配置 PE 和 CE 间路由交互

PE 与 CE 之间的路由协议可以是：EBGP、IBGP、静态路由、RIP、OSPF、IS-IS。配置时根据实际情况选择其一即可。

背景信息

根据实际情况，选择如下配置之一：

- **配置 PE 和 CE 间使用 EBGP**
- **配置 PE 和 CE 间使用 IBGP**
- **配置 PE 和 CE 间使用静态路由**
- **配置 PE 和 CE 间使用 RIP**
- **配置 PE 和 CE 间使用 OSPF**
- **配置 PE 和 CE 间使用 IS-IS**

操作步骤

- 配置 PE 和 CE 间使用 EBGP

  在 PE 上执行如下配置：

  1. 在 PE 上，执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。

3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
4. （可选）执行命令 **as-number** *as-number*，为 VPN 实例 IPv4 地址族配置单独的 AS 号。

   当进行网络迁移或业务标识时，如果需要将一台物理设备在逻辑上模拟为多台 BGP 设备，可通过 **as-number** 命令为每个 VPN 实例 IPv4 地址族配置不同的 AS 号。

   说明

   在 BGP-VPN 实例 IPv4 地址族中配置单独的 AS 号时，不可以与 BGP 视图下配置的 AS 号相同。
5. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，将 CE 配置为 VPN 私网对等体。
6. （可选）执行命令 **peer** { *ipv4-address* | *group-name* } **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。

   通常情况下，EBGP 对等体之间必须具有直连的物理链路，如果不满足这一要求，则必须使用 **peer ebgp-max-hop** 命令允许它们之间经过多跳建立 TCP 连接。
7. （可选）当需要将到本地 CE 的直连路由引入 VPN 路由表中，以发布给对端 PE 时，可选择如下配置之一：
   - 执行命令 **import-route direct** [**med** *med* | **route-policy** *route-policy-name*]*，引入到本地 CE 的直连路由。
   - 执行命令 **network** *ipv4-address* [*mask* | *mask-length*] [**route-policy** *route-policy-name*]，发布到本地 CE 的直连路由。

     说明

     PE 会自动学习到本地 CE 直连路由，该路由优于本地 CE 通过 EBGP 发布过来的直连路由，因此如果不配置此步骤，PE 不会将该直连路由通过 MP-BGP 发布给对端 PE。
8. （可选）执行命令 **peer** { *group-name* | *ipv4-address* } **soo** *site-of-origin*，对于指定的 CE 对等体，配置 Site-of-Origin（SoO）属性。

   VPN 某站点有多个 CE 采用 BGP 协议接入不同的 PE 时，从 CE 发往 PE 的 VPN 路由可能经过骨干网又回到了该站点，这样很可能会引起 VPN 站点内路由环路。

   应用 SoO 特性后，当 PE 收到 CE 发来的路由后，会为该路由添加 SoO 属性并发布给其他的 PE 对等体。其他 PE 对等体向接入的 CE 发布路由时会检查 VPN 路由携带的 SoO 属性，如果与本地配置的 SoO 属性相同，PE 则不会向 CE 发布该路由。
9. （可选）执行命令 **peer** *ip-address* **allow-as-loop** [*number*]，允许路由环路。

   此步骤用于 Hub and Spoke 组网方案。

   通常情况下，BGP 通过 AS 号检测路由环路。但在 Hub and Spoke 组网方式下，如果在 Hub 节点的 PE 和 CE 之间运行 EBGP，当 Hub-PE 将路由信息通告给 Hub-CE 时带上本自治系统的 AS 号。再从 Hub-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，Hub-PE 就不能接收这条路由更新信息。为保证 Hub and Spoke 组网方式中路由能够正确传递，从 Hub-CE

发布私网路由到 Spoke-CE 途中经过的相关 BGP 对等体需要配置允许 AS 重复 1 次的路由通过。

10. （可选）执行命令 **peer** *ip-address* **substitute-as**，使能 BGP 的 AS 号替换功能。

    此步骤用于物理分散的 CE 复用相同 AS 号的组网方案，在 PE 上配置。

> **注意**
> 在 CE 多归属的情况下，使能 BGP AS 号替换功能可能引起路由环路。

在 CE 上执行如下配置：

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，将 PE 配置为对等体。
4. （可选）执行命令 **peer** { *ipv4-address* | *group-name* } **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。

   通常情况下，EBGP 对等体之间必须具有直连的物理链路，如果不满足这一要求，则必须使用 **peer ebgp-max-hop** 命令允许 EBGP 对等体之间经过多跳建立 TCP 连接。
5. 执行命令 **import-route** { **direct** | **static** | **rip** *process-id* | **ospf** *process-id* | **isis** *process-id* } [**med** *med* | **route-policy** *route-policy-name*]*，引入本站点的路由。

   CE 将自己的 VPN 网段地址发布给接入的 PE，通过 PE 发布给对端 CE。根据实际组网情况，该步骤中需要引入的路由类型有所不同。

- 配置 PE 和 CE 间使用 IBGP

  在 PE 上执行如下配置：

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
  4. （可选）执行命令 **as-number** *as-number*，为 VPN 实例 IPv4 地址族配置单独的 AS 号。

     当进行网络迁移或业务部署时，如果需要将一台物理设备在逻辑上模拟为多台 BGP 设备，可通过 **as-number** 命令为每个 VPN 实例 IPv4 地址族配置不同的 AS 号。

     说明

     在 BGP-VPN 实例 IPv4 地址族中配置单独的 AS 号时，不可以与 BGP 视图下配置的 AS 号相同。
  5. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，将 CE 配置为 VPN 私网对等体。
  6. （可选）当需要将到本端 CE 的直连路由引入 VPN 路由表中，以发布给对端 PE 时，可选择如下配置之一：
     - 执行命令 **import-route direct** [**med** *med* | **route-policy** *route-policy-name*]*，引入到本地 CE 的直连路由。

- 执行命令 **network** *ipv4-address* [*mask* | *mask-length*] [**route-policy** *route-policy-name*]，发布到本地 CE 的直连路由。

  说明

  PE 会自动学习到本地 CE 直连路由，该路由优于本地 CE 通过 IBGP 发布过来的直连路由，因此如果不配置此步骤，PE 不会将该直连路由通过 MP-BGP 发布给对端 PE。

在 CE 上进行如下配置：

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
3. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，将 PE 配置为 IBGP 对等体。
4. 执行命令 **import-route** { **direct** | **static** | **rip** *process-id* | **ospf** *process-id* | **isis** *process-id* } [**med** *med* | **route-policy** *route-policy-name*]*，引入本站点的路由。

   CE 将自己的 VPN 网段地址发布给接入的 PE，通过 PE 发布给对端 CE。根据实际组网情况，该步骤中需要引入的路由类型有所不同。

- 配置 PE 和 CE 间使用静态路由

  在 PE 上进行如下配置。CE 上的配置方法与普通静态路由相同，此处不再详述。

  说明

  有关静态路由的详细配置，请参见《Huawei AR1200 系列企业路由器 配置指南 IP 路由》。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ip route-static vpn-instance** *vpn-source-name destination-address* { *mask* | *mask-length* } *interface-type interface-number* [*nexthop-address*] [**preference** *preference* | **tag** *tag*] *，为指定 VPN 实例 IPv4 地址族配置静态路由。
  3. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  4. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
  5. 执行命令 **import-route static** [**med** *med* | **route-policy** *route-policy-name*]*，将配置的静态路由引入 BGP-VPN 实例 IPv4 地址族路由表。

- 配置 PE 和 CE 间使用 RIP

  在 PE 上进行如下配置。CE 上配置普通 RIPv1 或 RIPv2 协议，此处不再详述。

  说明

  有关 RIP 的详细配置，请参见《Huawei AR1200 系列企业路由器 配置指南 IP 路由》。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **rip** *process-id* **vpn-instance** *vpn-instance-name*，创建 PE 和 CE 间的 RIP 实例，并进入 RIP 视图。

     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。属于公网的 RIP 进程不能再绑定到 VPN 实例。
  3. 执行命令 **network** *network-address*，在 VPN 实例绑定的接口所在网段运行 RIP。
  4. 执行命令 **import-route bgp** [**cost** { *cost* | **transparent** } | **route-policy** *route-policy-name*]*，引入 BGP 路由。

     在 RIP 视图下执行 **import-route bgp** 命令后，PE 把从对端 PE 学到的 VPNv4 路由引入到 RIP 中，进而发布给自己的 CE。

5. 执行命令 **quit**，退回系统视图。
6. 执行命令 **bgp** *as-number*，进入 BGP 视图。
7. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
8. 执行命令 **import-route rip** *process-id* [**med** *med* | **route-policy** *route-policy-name*]*，将 RIP 路由引入 BGP-VPN 实例 IPv4 地址族路由表。

   在 BGP-VPN 实例 IPv4 地址族视图下执行 **import-route rip** 命令后，PE 把从自己的 CE 学到的 VPN 路由引入 BGP 中，形成 VPN-IPv4 路由发布给对端 PE。

说明

删除 VPN 实例或者去使能 VPN 实例 IPv4 地址族后，所有相关的 RIP 进程也全部被删除。

- 配置 PE 和 CE 间使用 OSPF

  在 PE 上进行如下配置，CE 上配置普通 OSPF 即可，此处不再详述。

说明

有关 OSPF 的详细配置，请参见《Huawei AR1200 系列企业路由器 配置指南 IP 路由》。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **ospf** *process-id* [**router-id** *router-id*] **vpn-instance** *vpn-instance-name*，创建 PE-CE 间的 OSPF 实例，并进入 OSPF 视图。

   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。属于公网的 OSPF 进程不能再绑定到 VPN 实例。

   绑定到 VPN 实例的 OSPF 进程不使用系统视图下配置的公网 Router ID，用户需要在启动进程时指定 Router ID。如果不指定 Router ID，则 OSPF 会根据 Router ID 选取规则在所有绑定了该 VPN 实例的接口 IP 地址中选取一个作为 Router ID。
3. （可选）执行命令 **domain-id** *domain-id* [**secondary**]，配置域 ID。

   域 ID 可以用整数表示，也可以用点分十进制表示。

   每个 OSPF 进程可以配置两个域 ID，不同进程的域 ID 相互没有影响。PE 上不同 VPN 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。

   缺省情况下，域 ID 为 0。
4. （可选）执行命令 **route-tag** *tag*，配置 VPN route tag。

   缺省情况下，OSPF 根据算法自动分配一个 Tag。

   - 如果本地设备没有启用 BGP 进程，则缺省情况下，Tag 值为 0。
   - 如果本地设备启用了 BGP 进程，则缺省情况下，Tag 值的前面两个字节为固定的 0xD000，后面的两个字节为本端 BGP 的 AS 号，即 Tag 值 = 3489660928 + BGP 的 AS 号。
5. 执行命令 **import-route bgp** [**cost** *cost* | **route-policy** *route-policy-name* | **tag** *tag* | **type** *type*] *，引入 BGP 路由。
6. 执行命令 **area** *area-id*，进入 OSPF 区域视图。

7. 执行命令 **network** *ip-address wildcard-mask*，在 VPN 实例绑定的接口所在网段运行 OSPF。

   一个网段只能属于一个区域，或者说每个运行 OSPF 协议的接口必须指明属于某一个特定的区域。

   满足下面两个条件，接口上才能正常运行 OSPF 协议：

   - 接口的 IP 地址掩码长度≥**network** 命令中的掩码长度。
   - 接口的主 IP 地址必须在 **network** 命令指定的网段范围内。

   对于 Loopback 接口，缺省情况下 OSPF 以 32 位主机路由的方式对外发布其 IP 地址，与接口上配置的掩码长度无关。
8. 执行命令 **quit**，退回 OSPF 视图。
9. 执行命令 **quit**，退回系统视图。
10. 执行命令 **bgp** *as-number*，进入 BGP 视图。
11. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
12. 执行命令 **import-route ospf** *process-id* [**med** *med* | **route-policy** *route-policy-name*]*，将 OSPF 路由引入 BGP-VPN 实例 IPv4 地址族路由表。

说明

删除 VPN 实例或者去使能 VPN 实例 IPv4 地址族后，相关的所有 OSPF 进程也将全部被删除。

- 配置 PE 和 CE 间使用 IS-IS

  在 PE 上进行如下配置，CE 上配置普通 IS-IS 即可，此处不再详述。

  说明

  有关 IS-IS 的详细配置，请参见《Huawei AR1200 系列企业路由器 配置指南 IP 路由》。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **isis** *process-id* **vpn-instance** *vpn-instance-name*，创建 PE-CE 间的 IS-IS 实例，并进入 IS-IS 视图。

     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。属于公网的 IS-IS 进程不能再绑定到 VPN 实例。
  3. 执行命令 **network-entity** *net*，设置网络实体名称。

     网络实体名称 NET（Network Entity Title）同时定义了当前 IS-IS 的区域地址和路由器的系统 ID。在一台路由器的一个进程中最多可以配置 3 个 NET。
  4. （可选）执行命令 **is-level** { **level-1** | **level-1-2** | **level-2** }，设置路由器的 Level 级别。

     缺省情况下，路由器的 Level 级别为 **level-1-2**。
  5. 执行命令 **import-route bgp** [**cost-type** { **external** | **internal** } | **cost** *cost* | **tag** *tag* | **route-policy** *route-policy-name* | [**level-1** | **level-2** | **level-1-2**]] *，引入 BGP 路由。

     配置该命令时，如果没有指定 IS-IS 的 Level，BGP 路由被引入到 Level-2 的路由表中。
  6. 执行命令 **quit**，退回系统视图。

7. 执行命令 **interface** *interface-type interface-number*，进入绑定 VPN 实例的接口视图。
8. 执行命令 **isis enable** [*process-id*]，在该接口上运行 IS-IS。
9. 执行命令 **quit**，退回系统视图。
10. 执行命令 **bgp** *as-number*，进入 BGP 视图。
11. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
12. 执行命令 **import-route isis** *process-id* [**med** *med* | **route-policy** *route-policy-name*]*，将 IS-IS 路由引入 BGP-VPN 实例 IPv4 地址族路由表。

说明

删除 VPN 实例或者去使能 VPN 实例 IPv4 地址族后，相关的所有 IS-IS 进程也将全部被删除。

----结束

2.4.7 检查配置结果

配置了基本 BGP/MPLS IP VPN 后，可以在 PE 设备和 CE 设备上查看到本端站点和远端站点的 IPv4 VPN 路由信息。

前提条件

已经完成基本 BGP/MPLS IP VPN 功能的所有配置。

操作步骤

- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 PE 上查看指定 VPN 实例 IPv4 地址族的路由信息。
- 使用 **display ip routing-table** 命令在 CE 上查看路由信息。
- 使用 **display ip vpn-instance** [*vpn-instance-name*] **interface** 命令查看指定 VPN 实例所绑定的接口信息。

----结束

任务示例

在配置成功时，在 PE 上执行命令 **display ip routing-table vpn-instance** *vpn-instance-name* 查看指定 VPN 的路由信息，可以看到 PE 上存在相关 CE 的 VPN 路由。

在 CE 上执行命令 **display ip routing-table**，可以看到 CE 上有到所有对端 CE 的路由。

配置成功时，PE 上执行命令 **display ip vpn-instance** [*vpn-instance-name*] **interface**，可以查看到有接口绑定到了 VPN 实例。

2.5 配置 Hub and Spoke

Hub and Spoke 组网是通过在 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。

2.5.1 建立配置任务

在配置 Hub and Spoke 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

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 and Spoke 组网方案，使得 Spoke 站点之间的通信流经 Hub 站点。

前置任务

在配置 Hub and Spoke 之前，需完成如下任务：

- 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
- 在 CE 上配置接入 PE 的接口的 IP 地址

数据准备

在配置 Hub and Spoke 之前，需准备以下数据。

序号	数据
1	配置 VPN 实例所需的数据，包括： ● VPN 实例名称 ●（可选）VPN 实例的描述信息 ● VPN 实例 IPv4 地址族的 RD、VPN Target 属性 ●（可选）控制 VPN 路由信息收发的路由策略 ●（可选）VPN 实例 IPv4 地址族中允许的最大路由数 ●（可选）VPN 实例 IPv4 地址族中允许的最大路由前缀数 ●（可选）VPN 实例 IPv4 地址族的路由超出限制后输出日志的频率
2	PE 上接入 CE 的接口的 IP 地址
3	Hub-PE 与 Hub-CE 间，及 Spoke-PE 与 Spoke-CE 间配置路由交换（静态路由、RIP、OSPF、IS-IS 或 EBGP）所需数据

2.5.2 配置 VPN 实例

配置 VPN 实例，用以管理 VPN 路由。

背景信息

在每个 Spoke-PE 及 Hub-PE 上进行如下配置。

每个 Spoke-PE 上配置一个 VPN 实例；Hub-PE 需配置两个 VPN 实例（VPN-in 和 VPN-out）：

- VPN-in 用于接收并维护所有 Spoke-PE 发布的 VPNv4 路由。
- VPN-out 用于维护 Hub 站点及所有 Spoke 站点的路由，并发布给所有 Spoke-PE。

说明

- 每个 VPN 实例的配置都相似，只是同一台设备上不同 VPN 实例，其名称、RD 及描述信息不同。
- 建议第 6 步和第 7 步只选其中之一配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip vpn-instance** *vpn-instance-name*，创建 VPN 实例，并进入 VPN 实例视图。

VPN 实例的名字区分大小写。例如，“vpn1”和“VPN1”将被认为是不同的 VPN 实例。

步骤 3 （可选）执行命令 **description** *description-information*，配置 VPN 实例的描述信息。

描述信息的作用类似于主机名和接口描述信息，建议用户配置合适的描述信息。

步骤 4 执行命令 **ipv4-family**，使能 VPN 实例 IPv4 地址族，并进入 VPN 实例 IPv4 地址族视图。

步骤 5 执行命令 **route-distinguisher** *route-distinguisher*，配置 VPN 实例 IPv4 地址族的 RD。

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，不能配置其他任何参数。

步骤 6 （可选）执行命令 **apply-label per-instance**，基于 VPN 实例 IPv4 地址族分配 MPLS 标签，使 VPN 实例 IPv4 地址族中的所有路由都使用同一个标签。

通常情况下，标签的分配方式是每条路由一个标签（one label per route）。AR1200 实现基于 VPN 的 MPLS 标签分配，即，VPN 实例 IPv4 地址族中的所有路由都使用同一个标签。

步骤 7 （可选）执行命令 **routing-table limit** *number* { *alert-percent* | **simply-alert** }，配置 VPN 实例 IPv4 地址族的最大路由数。

为防止 PE 设备引入的路由数量过多，可以配置一个 VPN 实例 IPv4 地址族能够支持的最大路由数。

说明

配置了 **routing-table limit** 命令，当注入到 VPN 实例 IPv4 地址族路由表的路由超限时，系统会给出提示信息。执行 **routing-table limit** 命令增大 VPN 实例 IPv4 地址族下支持的最大路由数或者执行 **undo routing-table limit** 命令取消路由表限制后，对于这些超限的路由，还要进行如下处理：

- 对于超限的静态路由，需要手动重新配置。
- 通过 IGP 多实例路由协议从 CE 学到的路由，需要在 PE 上重启路由协议的多实例进程。

通过 MP-IBGP 学到的远端交叉路由和从 CE 上学来的 BGP 路由，系统可以自动刷新。

步骤 8 （可选）执行命令 **prefix limit** *number* { *alert-percent* [**route-unchanged**] | **simply-alert** }，配置 VPN 实例 IPv4 地址族的最大路由前缀数。

为防止 PE 设备引入的路由前缀数量过多，可以配置一个 VPN 实例 IPv4 地址族能够支持的最大路由前缀数。

步骤 9 （可选）执行命令 **limit-log-interval** *interval*，配置 VPN 实例 IPv4 地址族的路由超出限制后输出日志的频率。

----结束

2.5.3 配置 VPN 实例的路由相关属性

通过配置 VPN-Target 来控制路由的发布和接受。

操作步骤

- 配置 Hub-PE
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ip vpn-instance** *vpn-instance-name1*，进入 VPN-in 的 VPN 实例视图。
  3. 执行命令 **ipv4-family**，进入 VPN 实例 IPv4 地址族视图。
  4. 执行命令 **vpn-target** *vpn-target1* &<1-8> **import-extcommunity**，为 VPN 实例 IPv4 地址族配置 VPN-target 扩展团体，使得该实例 IPv4 地址族可以接收所有 Spoke-PE 发布的 VPNv4 路由。

     这里的 *vpn-target1* 列表包含所有 Spoke-PE 的 *vpn-target* 出方向团体属性值。
  5. （可选）执行命令 **import route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族入方向路由策略。
  6. （可选）执行命令 **export route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族出方向路由策略。
  7. 执行命令 **quit**，退回系统视图。
  8. 执行命令 **ip vpn-instance** *vpn-instance-name2*，进入 VPN-out 的 VPN 实例视图。
  9. 执行命令 **ipv4-family**，进入 VPN 实例 IPv4 地址族视图。
  10. 执行命令 **vpn-target** *vpn-target2* &<1-8> **export-extcommunity**，配置 VPN-target 扩展团体，发布 Hub 站点及所有 Spoke 站点的路由。

      这里的 *vpn-target2* 列表包括所有 Spoke-PE 的 *vpn-target* 入方向团体属性值。
  11. （可选）执行命令 **import route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族入方向路由策略。
  12. （可选）执行命令 **export route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族出方向路由策略。
- 配置 Spoke-PE
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ip vpn-instance** *vpn-instance-name*，进入 VPN-in 的 VPN 实例视图。
  3. 执行命令 **ipv4-family**，进入 VPN 实例 IPv4 地址族视图。
  4. 执行命令 **vpn-target** *vpn-target2* &<1-8> **import-extcommunity**，配置 VPN-target 扩展团体，使得该实例 IPv4 地址族可以接收 Hub-PE 发布的 VPNv4 路由。

     这里的 *vpn-target2* 需要包含于 Hub-PE 的 *vpn-target* 出方向团体属性值列表中。
  5. 执行命令 **vpn-target** *vpn-target1* &<1-8> **export-extcommunity**，配置 VPN-target 扩展团体，发布本 Spoke-PE 所接入的站点的路由。

     这里的 *vpn-target1* 需要包含于 Hub-PE 的 *vpn-target* 入方向团体属性值列表中。

6. （可选）执行命令 **import route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族入方向路由策略。
7. （可选）执行命令 **export route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族出方向路由策略。

----结束

2.5.4 配置接口与 VPN 实例绑定

通过配置接口与 VPN 实例绑定，该接口成为私网接口，从该接口进入的报文使用 VPN 实例中的转发信息进行转发。同时将删除该接口上已经配置的 IP 地址、路由协议等三层特性，如果需要应重新配置。

背景信息

Hub-PE 上需要使用两个接口或子接口：一个绑定 VPN-in，用于接收 Spoke-PE 发来的路由；另一个绑定 VPN-out，用于发布 Hub 站点及所有 Spoke 站点的路由。

在 Hub-PE 及所有 Spoke-PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入要绑定接口的接口视图。

步骤 3 执行命令 **ip binding vpn-instance** *vpn-instance-name*，将当前接口与 VPN 实例绑定。

说明

执行 **ip binding vpn-instance** 命令将删除接口上已经配置的 IP 地址、路由协议等三层特性，如果需要应重新配置。

接口不能与未使能任何地址族的 VPN 实例绑定。

当去使能 VPN 实例的某地址族时，也将删除与 VPN 实例绑定接口下的 IP 地址、路由协议等三层特性。当去使能 VPN 实例下所有地址族时，去使能接口与该 VPN 实例的绑定。

步骤 4 执行命令 **ip address** *ip-address* { *mask* | *mask-length* }，配置接口的 IP 地址。

----结束

2.5.5 配置 Hub-PE 与 Spoke-PE 间使用 MP-IBGP

MP-IBGP 通过在 BGP 中引入扩展团体属性，使其能够在 PE 设备之间传播 VPNv4 路由。

背景信息

Hub-PE 与所有的 Spoke-PE 都需要建立 MP-IBGP 对等体，但 Spoke-PE 间不要建立 MP-IBGP 对等体。

在 Hub-PE 与 Spoke-PE 间建立 MP-IBGP 对等体，就是分别在 Hub-PE 和 Spoke-PE 上进行如下配置。

操作步骤

步骤 1 执行 **system-view**，进入系统视图。

步骤 2 执行 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行 **peer** *ipv4-address* **as-number** *as-number*，将对端 PE 配置为对等体。

步骤 4 执行 **peer** *ipv4-address* **connect-interface loopback** *interface-number*，指定建立 TCP 连接的接口。

说明

PE 之间必须使用 32 位掩码的 Loopback 接口地址来建立 MP-IBGP 对等体关系，以便能够迭代到隧道。到 Loopback 接口的路由通过 MPLS 骨干网上的 IGP 发布给对端 PE。

步骤 5 执行 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 子地址族视图。

步骤 6 执行 **peer** *ipv4-address* **enable**，使能对等体交换 VPNv4 路由信息。

----结束

2.5.6 配置 PE 与 CE 间路由交换

PE 与 CE 之间的路由协议可以是：EBGP、静态路由和 IGP。配置时根据实际情况选择其一即可。

背景信息

Hub-PE 与 Hub-CE 间路由交换有以下形式。

操作步骤

- Hub-PE 与 Hub-CE 间使用 EBGP

  这种方式中，Spoke-PE 与 Spoke-CE 间可使用 EBGP，IGP 或静态路由。

  当 Spoke-PE 与 Spoke-CE 间及 Hub-PE 与 Hub-CE 使用 EBGP 时，需要在 Hub-PE 上执行如下命令。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
  4. 执行命令 **peer** *ip-address* **allow-as-loop** [*number*]，允许路由环路。这里的 *number* 取 1，允许 AS 重复 1 次的路由通过。

- Hub-PE 与 Hub-CE 间使用 IGP

  这种方式中，Spoke-PE 与 Spoke-CE 间只能使用 IGP 或静态路由，不能使用 BGP。详细介绍请参见《Huawei AR1200 系列企业路由器 特性描述 VPN》的“BGP/MPLS IP VPN”一章。

- Hub-PE 与 Hub-CE 间使用静态路由

  这种方式中，Spoke-PE 与 Spoke-CE 间可使用 EBGP，IGP 或静态路由。

  如果 Hub-CE 使用默认路由接入 Hub-PE，为了将此默认路由发布给所有 Spoke-PE 需要在 Hub-PE 上进行以下配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **ip route-static vpn-instance** *vpn-source-name* **0.0.0.0 0.0.0.0** *nexthop-address* [**preference** *preference* | **tag** *tag*]* [**description** *text*]

   这里的 *vpn-source-name* 是 VPN-out，*nexthop-address* 是绑定 VPN-out 的接口所在链路的 Hub-CE 侧接口 IP 地址。
3. 执行命令 **bgp** *as-number*，进入 BGP 视图。
4. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。

   这里的 *vpn-instance-name* 也是 VPN-out。
5. 在 BGP-VPN 实例 IPv4 地址族视图下配置 **network 0.0.0.0 0**，通过 MP-BGP 发布缺省路由给所有 Spoke-PE

----结束

后续处理

根据实际情况，选择其中的一种方式即可。具体配置过程请参见配置 PE 和 CE 间路由交换。

2.5.7 检查配置结果

配置了 Hub and Spoke 后，可以在 PE 设备和 CE 设备上查看到 VPN 的路由信息。

前提条件

已经完成 Hub and Spoke 功能的所有配置。

操作步骤

- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 Hub-PE 上查看 VPN-in 和 VPN-out 的路由信息。
- 使用 **display ip routing-table** 命令在 Hub-CE 和所有 Spoke-CE 上查看路由信息。

----结束

任务示例

在配置成功时，查看 VPN-in 和 VPN-out 的路由信息，可以看到 VPN-in 的路由表中有到所有 Spoke 站点的路由；VPN-out 的路由有到 Hub 站点及到所有 Spoke 站点的路由。

Hub-CE 和所有 Spoke-CE 上有到 Hub 站点和到所有 Spoke 站点的路由。

```
<Huawei> display ip routing-table
 Total Number of Routes: 6
 BGP Local router ID is 100.1.1.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
     Network         NextHop      MED    LocPrf    PrefVal Path/Ogn
 *>  100.1.1.0/24    0.0.0.0       0                0       ?
 *                   100.1.1.2     0                0       100?
 *>  100.1.1.1/32    0.0.0.0       0                0       ?
 *>  110.1.1.0/24    100.1.1.2                      0       100 65430?
 *>  110.2.1.0/24    100.1.1.2                      0       100?
 *>  120.1.1.0/24    100.1.1.2                      0       100 65430 100?
```

```
<Huawei> display ip routing-table vpn-instance
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpn1
         Destinations : 3        Routes : 3

Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface

        1.1.1.0/24  Direct 0    0           D  1.1.1.1         Ethernet2/0/0
        1.1.1.1/32  Direct 0    0           D  127.0.0.1       Ethernet2/0/0
        5.5.5.0/24  Static 60   0          RD  1.1.1.2         Ethernet2/0/0
```

2.6 配置跨域 VPN-OptionA

跨域 VPN-OptionA 中，ASBR 把对端 ASBR 看作自己的 CE 设备，使用 EBGP 方式向对端发布 VPNv4 路由。

2.6.1 建立配置任务

在配置跨域 VPN-OptionA 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

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。

当 PE 上接入的 VPN 数量及 VPN 路由数量都比较少时可以采用跨域 VPN-OptionA 方案。采用跨域 VPN-OptionA 方案，要求 AS 的边界设备 ASBR 支持 VPN 实例，能够管理 VPN 路由。并且，ASBR 上必须要为每个跨域的 VPN 准备专用的接口（可以是子接口、物理接口、捆绑的逻辑接口）。因此，此方案对 ASBR 的性能要求较高，但 ASBR 上不必为跨域做任何特殊的配置。

前置任务

在配置 OptionA 方式跨域 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
- 在 PE 和 ASBR 上配置 MPLS 基本能力和 MPLS LDP
- 为同一 AS 的 PE 与 ASBR 之间建立隧道（LSP 或 GRE）
- 在 CE 上配置接入 PE 的接口的 IP 地址

数据准备

在配置 OptionA 方式跨域 VPN 之前，需准备以下数据。

序号	数据
1	PE 和 ASBR 上配置 VPN 实例所需的数据，包括： ● VPN 实例名称 ●（可选）VPN 实例的描述信息 ● VPN 实例 IPv4 地址族的 RD、VPN Target 属性 ●（可选）控制 VPN 路由信息收发的路由策略 ●（可选）隧道策略 ●（可选）VPN 实例 IPv4 地址族中允许的最大路由数
2	PE 上接入 CE 的接口的 IP 地址
3	PE 的 AS 号
4	ASBR 之间接口的 IP 地址
5	PE 和 CE 间采用的路由交换方式：静态路由、RIP、OSPF、IS-IS 或 BGP
6	PE 与 ASBR 间用来建立 IBGP 对等体的 IP 地址和接口

2.6.2 配置 OptionA 方式跨域 VPN

在 PE 和 ASBR 上分别配置 VPN 实例，前者用于接入 CE，后者用于接入对端 ASBR。

背景信息

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。

操作步骤

步骤 1 对各 AS 分别配置基本 BGP/MPLS IP VPN。

步骤 2 对于 ASBR，将对端 ASBR 看作自己的 CE 配置即可。

步骤 3 在 PE 和 ASBR 上分别配置 VPN 实例，前者用于接入 CE，后者用于接入对端 ASBR。

说明

在跨域 VPN-OptionA 方式中，对于同一个 VPN，同一 AS 内的 ASBR 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。

----结束

2.6.3 检查配置结果

配置了跨域 VPN-OptionA 后，可以查看到所有 BGP 对等体关系的建立情况、PE 或 ASBR 上的 IPv4 VPN 路由信息。

前提条件

已经完成跨域 VPN-OptionA 功能的所有配置。

操作步骤

- 使用 **display bgp vpnv4 all peer** 命令在 PE 或 ASBR 上检查所有 BGP 对等体关系的建立情况。
- 使用 **display bgp vpnv4 all routing-table** 命令查看 PE 或 ASBR 上的 VPNv4 路由。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 PE 或 ASBR 上检查 VPN 路由表。

----结束

任务示例

配置成功时，在 PE 或 ASBR 上执行 **display bgp vpnv4 all peer** 命令，可看到同一 AS 的 PE 和 ASBR 之间 BGP VPNv4 对等体关系的状态为“Established”。

```
<Huawei> display bgp vpnv4 all peer

 BGP local router ID : 10.1.1.1
 Local AS number : 100
 Total number of peers : 1                 Peers in established state : 1

  Peer            V          AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv

  2.2.2.2         4         100        2        2     0 00:00:00 Established      0
```

在 PE 或 ASBR 上执行 **display bgp vpnv4 all routing-table** 命令，可以看到 ASBR 上的 VPNv4 路由。

```
<Huawei> display bgp vpnv4 all all routing-table
 Local AS number : 100
BGP Local router ID is 2.2.2.9
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
 Total number of routes from all PE: 5
 Route Distinguisher: 100:1


      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  10.1.1.0/24        1.1.1.9         0          100        0      ?

 Route Distinguisher: 100:2


      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>   10.2.1.0/24        192.1.1.2                             0      200?
 *>   192.1.1.0          0.0.0.0         0                     0      ?
 *                       192.1.1.2       0                     0      200?
 *>   192.1.1.1/32       0.0.0.0         0                     0      ?

 VPN-Instance vpn1, router ID 2.2.2.9:

 Total Number of Routes: 5
      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  10.1.1.0/24        1.1.1.9         0          100        0      ?
 *>   10.2.1.0/24        192.1.1.2                             0      200?
 *>   192.1.1.0          0.0.0.0         0                     0      ?
                         192.1.1.2       0                     0      200?
 *>   192.1.1.1/32       0.0.0.0         0                     0      ?
```

在 PE 或 ASBR 上执行 **display ip routing-table vpn-instance** 命令，可以看到 PE 和 ASBR 上的 VPN 路由表中有所有相关 VPN 的路由。

```
<Huawei> display ip routing-table vpn-instance
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpn1
         Destinations : 3        Routes : 3

Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface

        1.1.1.0/24  Direct 0    0           D  1.1.1.1         Ethernet2/0/0
        1.1.1.1/32  Direct 0    0           D  127.0.0.1       Ethernet2/0/0
        5.5.5.0/24  Static 60   0          RD  1.1.1.2         Ethernet2/0/0
```

2.7 配置跨域 VPN-OptionB

跨域 VPN-OptionB 中，ASBR 之间通过 MP-EBGP 交换它们从各自 AS 的 PE 设备接收的 VPNv4 路由。

2.7.1 建立配置任务

在配置跨域 VPN-OptionB 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

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。

如果 ASBR 能够管理 VPN 路由，但是没有足够的接口为每个跨域 VPN 专用，可以选择跨域 VPN-OptionB。此方案要求 ASBR 设备参与 VPN-IPv4 路由的维护和发布。

前置任务

在配置 OptionB 方式跨域 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 基本能力和 MPLS LDP
- 在与 CE 相连的 PE 上配置 VPN 实例，并配置接口与 VPN 实例关联
- 在 CE 上配置接入 PE 的接口的 IP 地址

数据准备

在配置 OptionB 方式跨域 VPN 之前，需准备以下数据。

序号	数据
1	PE 上配置 VPN 实例所需的数据，包括： ● VPN 实例名称 ●（可选）VPN 实例的描述信息 ● VPN 实例 IPv4 地址族的 RD、VPN Target 属性 ●（可选）控制 VPN 路由信息收发的路由策略 ●（可选）VPN 实例 IPv4 地址族中允许的最大路由数
2	PE 上接入 CE 的接口的 IP 地址

序号	数据
3	PE 的 AS 号
4	ASBR 之间接口的 IP 地址
5	PE 和 CE 间采用的路由交换方式：静态路由、RIP、OSPF、IS-IS 或 BGP
6	PE 与 ASBR 间用来建立 IBGP 对等体的 IP 地址和接口

2.7.2 配置 PE 和域内 ASBR 间使用 MP-IBGP

MP-IBGP 通过在 BGP 中引入扩展团体属性，使其能够在 PE 和 ASBR 之间传播 VPNv4 路由。

背景信息

在同一 AS 的 PE 与 ASBR 上分别进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** *ipv4-address* **as-number** *as-number*，在同一 AS 内的 PE 和 ASBR 之间建立 IBGP 对等体。

步骤 4 执行命令 **peer** *ipv4-address* **connect-interface loopback** *interface-number*，指定 Loopback 接口为 BGP 会话的出接口。

说明

PE 之间必须使用 32 位掩码的 Loopback 接口地址来建立 MP-IBGP 对等体关系，以便能够迭代到隧道。到 Loopback 接口的路由通过 MPLS 骨干网上的 IGP 发布给对端 PE。

步骤 5 执行命令 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 地址族视图。

步骤 6 执行命令 **peer** *ipv4-address* **enable**，使能同一 AS 内的 PE 和 ASBR 之间交换 VPNv4 路由的能力。

说明

当 ASBR 在向 PE 发送 VPNv4 路由时，AR1200 仅支持 ASBR 自动将路由的下一跳改变为自身的 IP 地址。

----结束

2.7.3 配置不同 AS 的 ASBR 间使用 MP-EBGP

ASBR 之间建立 MP-EBGP 对等体关系，用来传播从本 AS 来的 VPNv4 路由到对端 ASBR。

背景信息

在 ASBR 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，在 ASBR 上进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入连接对端 ASBR 的接口视图。

步骤 3 执行命令 **ip address** *ip-address* { *mask* | *mask-length* }，配置接口 IP 地址。

步骤 4 执行命令 **mpls**，使能 MPLS 能力。

步骤 5 执行命令 **quit**，退回系统视图。

步骤 6 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 7 执行命令 **peer** *ipv4-address* **as-number** *as-number*，指定对端 ASBR 为自己的 EBGP 对等体。

步骤 8 （可选）执行命令 **peer** { *ipv4-address* | *group-name* } **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。

通常情况下，EBGP 对等体之间必须具有直连的物理链路，如果不满足这一要求，则必须使用 **peer ebgp-max-hop** 命令允许它们之间经过多跳建立 TCP 连接。

步骤 9 执行命令 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 子地址族视图。

步骤 10 执行命令 **peer** *ipv4-address* **enable**，使能与对端 ASBR 交换 VPNv4 路由的能力。

----结束

2.7.4 使用策略控制 VPN 路由收发

ASBR 上可以保存所有的 VPNv4 路由，也可以通过路由策略对 VPN-Target 进行过滤，只保存部分 VPNv4 路由。

背景信息

在 ASBR 上控制 VPN 路由收发有多种方法，这里介绍以下两种：

- 不进行 VPN-Target 过滤，即 ASBR 上保存所有的 VPN-IPv4 路由。
- 进行 VPN-Target 过滤，即 ASBR 上只保存部分 VPNv4 路由，通过路由策略实现。

实际应用中，只选择其中一种。

操作步骤

- 不进行 VPN-Target 过滤

  在 ASBR 上进行如下配置。

  1. 执行命令 **system-view**，在 ASBR 上进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 子地址族视图。
  4. 执行命令 **undo policy vpn-target**，不对 VPNv4 路由进行 VPN-Target 过滤。

     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。因此，如果 PE 没有配置 VPN 实例，或者 VPN 实例没有配置 VPN-Target，则 PE 丢弃所有收到的 VPNv4 路由。

在跨域 VPN-OptionB 方式中，ASBR 可以不保存 VPN 实例信息，但是 ASBR 需要保存所有 VPNv4 路由信息，以通告给对端 ASBR。这种情况下，ASBR 应接收所有的 VPNv4 路由信息，不对它们进行 VPN-Target 过滤。

- 进行 VPN-Target 过滤

  在 ASBR 上进行如下配置。

  1. 执行命令 **system-view**，在 ASBR 上进入系统视图。
  2. 执行命令 **ip extcommunity-filter** { *basic-extcomm-filter-num* | **basic** *basic-extcomm-filter-name* | *advanced-extcomm-filter-num* | **advanced** *advanced-extcomm-filter-name* } { **permit** | **deny** } { **rt** { *as-number*:*nn* | *4as-number*:*nn* | *ipv4-address*:*nn* } } &<1-16>，配置扩展团体属性过滤器。
  3. 执行命令 **route-policy** *route-policy-name* **permit node** *node*，配置路由策略。
  4. 执行命令 **if-match extcommunity-filter** { { *basic-extcomm-filter-num* | *advanced-extcomm-filter-num* } &<1-16> | *advanced-extcomm-filter-name* | *basic-extcomm-filter-name* }，为路由策略设置一个基于扩展团体属性过滤器的匹配规则。
  5. 执行命令 **quit**，退回系统视图。
  6. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  7. 执行命令 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 子地址族视图。
  8. 执行命令 **peer** *ipv4-address* **route-policy** *route-policy-name* { **export** | **import** }，应用路由策略控制 VPNv4 路由信息的收发。

----结束

2.7.5 （可选）ASBR 保存 VPN 实例信息

对需要经过 ASBR 收发 VPNv4 路由信息的 VPN，在 ASBR 上配置相应的 VPN 实例。

背景信息

对需要经过 ASBR 收发 VPN-IPv4 路由信息的 VPN，在 ASBR 上配置相应的实例。不需要经过该 ASBR 收发路由信息的 VPN，不必配置对应的实例。

在 ASBR 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，在 ASBR 上进入系统视图。

步骤 2 执行命令 **ip vpn-instance** *vpn-instance-name*，创建 VPN 实例，并进入 VPN 实例视图。

步骤 3 执行命令 **ipv4-family**，使能 VPN 实例 IPv4 地址族，并进入 VPN 实例 IPv4 地址族视图。

步骤 4 执行命令 **route-distinguisher** *route-distinguisher*，配置 VPN 实例 IPv4 地址族的 RD。

步骤 5 执行命令 **vpn-target** *vpn-target* &<1-8> **import-extcommunity**，为 VPN 实例 IPv4 地址族配置 VPN-target 扩展团体属性。

在跨域 VPN-OptionB 方式中，对于同一个 VPN，同一 AS 内的 ASBR 与 PE 的 VPN 实例 IPv4 地址族的 VPN-Target 应能匹配，不同 AS 的 PE 上 VPN 实例 IPv4 地址族的 VPN-Target 也需要匹配。

步骤 6 （可选）执行命令 **apply-label per-instance**，配置基于 VPN 实例 IPv4 地址族分配 MPLS 标签，使 VPN 实例中的所有路由都使用同一个标签。

步骤 7 （可选）执行命令 **routing-table limit** *number* { *alert-percent* | **simply-alert** }，配置 VPN 实例 IPv4 地址族的最大路由数。

步骤 8 （可选）执行命令 **prefix limit** *number* { *alert-percent* [**route-unchanged**] | **simply-alert** }，配置 VPN 实例 IPv4 地址族的最大路由前缀数。

步骤 9 （可选）执行命令 **limit-log-interval** *interval*，配置 VPN 实例 IPv4 地址族的路由超出限制后输出日志的频率。

步骤 10 （可选）执行命令 **import route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族入方向路由策略。

步骤 11 （可选）执行命令 **export route-policy** *policy-name*，配置 VPN 实例 IPv4 地址族出方向路由策略。

----结束

2.7.6 （可选）ASBR 按下一跳分标签

为了节省 ASBR 上的标签资源，可以在 ASBR 上使能按下一跳分标签。需要注意，ASBR 上按下一跳分标签和 PE 上的每实例每标签需要配合使用。

背景信息

跨域 VPN Option B 场景中，ASBR 使能按下一跳分标签后，对于下一跳和出标签相同的 VPNv4 路由，ASBR 只分配一个标签。相比为每一条 VPN 路由分标签，按下一跳分标签可以大大节省 ASBR 上的标签资源。

请在 ASBR 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family vpnv4**，进入 BGP-VPVv4 视图。

步骤 4 执行命令 **apply-label per-nexthop**，使能 ASBR 按下一跳为 VPNv4 路由分标签。

注意

使能或者去使能 ASBR 按下一跳分标签时，ASBR 申请的标签会发生变化，导致流量丢失。

----结束

2.7.7 配置 CE 和 PE 间路由交换

CE 与 PE 之间的路由协议可以是：BGP、静态路由、IGP。配置时，根据实际情况选择其一即可。

操作步骤

步骤 1 根据实际情况，选择其中一种，具体请参见配置 PE 和 CE 间路由交换。

----结束

2.7.8 检查配置结果

配置了跨域 VPN-OptionB 后，可以查看到所有 BGP 对等体关系的建立情况、PE 或 ASBR 上的 VPNv4 路由信息。

前提条件

已经完成跨域 VPN-OptionB 功能的所有配置。

操作步骤

- 使用 **display bgp vpnv4 all peer** 命令在 PE 或 ASBR 上检查所有 BGP 对等体关系的建立情况。
- 使用 **display bgp vpnv4 all routing-table** 命令查看 PE 或 ASBR 上的 VPNv4 路由。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 PE 上检查 VPN 路由表。
- 使用 **display mpls lsp** 命令查看 ASBR 上的 LSP 和标签信息。

----结束

任务示例

在 ASBR 执行命令 **display bgp vpnv4 all routing-table**，可以查看到 ASBR 上有为 VPN 维护的 IPv4 路由。

```
<Huawei> display bgp vpnv4 all all routing-table
 Local AS number : 100
BGP Local router ID is 2.2.2.9
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
 Total number of routes from all PE: 5
 Route Distinguisher: 100:1


      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  10.1.1.0/24        1.1.1.9         0          100        0      ?

 Route Distinguisher: 100:2


      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>   10.2.1.0/24        192.1.1.2                             0      200?
 *>   192.1.1.0          0.0.0.0         0                     0      ?
 *                       192.1.1.2       0                     0      200?
 *>   192.1.1.1/32       0.0.0.0         0                     0      ?

 VPN-Instance vpn1, router ID 2.2.2.9:

 Total Number of Routes: 5
      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  10.1.1.0/24        1.1.1.9         0          100        0      ?
```

```
 *>   10.2.1.0/24       192.1.1.2                        0      200?
 *>   192.1.1.0         0.0.0.0         0                0      ?
                        192.1.1.2       0                0      200?
 *>   192.1.1.1/32      0.0.0.0         0                0      ?
```

在 PE 或 ASBR 上执行命令 **display bgp vpnv4 all peer**，可以看到所有同一 AS 的 PE 与 ASBR 之间的 IBGP 对等体关系状态为“Established”；所在不同域的两个直连 ASBR 之间的 EBGP 对等体关系状态也为“Established”。

```
<Huawei> display bgp vpnv4 all peer

 BGP local router ID : 10.1.1.1
 Local AS number : 100
 Total number of peers : 1                 Peers in established state : 1

  Peer            V          AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv

  2.2.2.2         4         100        2        2     0 00:00:00 Established      0
```

在 PE 上执行命令 **display ip routing-table vpn-instance**，可以看到 PE 的 VPN 路由表中有所有相关 VPN 的路由。

```
<Huawei> display ip routing-table vpn-instance
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpn1
         Destinations : 3        Routes : 3

Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface

        1.1.1.0/24  Direct 0    0           D  1.1.1.1         Ethernet2/0/0
        1.1.1.1/32  Direct 0    0           D  127.0.0.1       Ethernet2/0/0
        5.5.5.0/24  Static 60   0          RD  1.1.1.2         Ethernet2/0/0
```

在 ASBR 上执行命令 **display mpls lsp**，可以看到 ASBR 上的 LSP 和标签信息。如果 ASBR 上使能了按下一跳分标签，可以看到对于下一跳和出标签相同的 VPN 路由，只分配一个标签。

```
<Huawei> display mpls lsp
------------------------------------------------------------------------------
                LSP Information: LDP LSP
------------------------------------------------------------------------------
  FEC                In/Out Label  In/Out IF                      Vrf Name
  2.2.2.9/32         NULL/3        -/Pos1/0/0
  2.2.2.9/32         1024/3        -/Pos1/0/0
  3.3.3.9/32         NULL/3        -/Pos1/0/1
  3.3.3.9/32         1025/3        -/Pos1/0/1
```

2.8 配置跨域 VPN-OptionC（方案一）

不同 AS 的 PE 之间建立 Multihop 方式的 EBGP 连接，交换 VPNv4 路由。

2.8.1 建立配置任务

在配置跨域 VPN-OptionC 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

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。

当每个 AS 都有大量的 VPN 路由需要交换时，可选择跨域 VPN-OptionC 方式，防止 ASBR 成为阻碍网络进一步扩展的瓶颈。实现跨域 VPN-OptionC 方式可以采用以下两种方式：

- 方案一：当本端 ASBR 从对端的 ASBR 学到对端 AS 域内的带标签 BGP 公网路由后，通过策略为该路由分配标签，发布给支持标签能力的 IBGP 邻居 PE，从而建立一条完整的公网 LSP。
- 方案二：在 PE 和 ASBR 之间不用配置 IBGP 邻居。当 ASBR 从对端的 ASBR 学到对端 AS 域的带标签 BGP 公网路由后，通过将本端 ASBR 上的 BGP 路由引入 IGP 协议中，并触发为带标签的公网 BGP 路由建立 LDP LSP，从而建立一条完整的公网 LSP。

这里对方案一进行说明，方案二在 **2.9 配置跨域 VPN-OptionC（方案二）** 中介绍。

前置任务

在配置 OptionC 方式跨域 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 基本能力和 MPLS LDP
- 为同一 AS 的 PE 与 ASBR 之间建立 IBGP 对等体关系
- 在与 CE 相连的 PE 上配置 VPN 实例，并配置接口与 VPN 实例关联
- 在 CE 上配置接入 PE 的接口的 IP 地址

数据准备

在配置 OptionC 方式跨域 VPN 之前，需准备以下数据。

序号	数据
1	PE 上配置 VPN 实例所需的数据，包括： ● VPN 实例名称 ●（可选）VPN 实例的描述信息 ● VPN 实例 IPv4 地址族的 RD、VPN Target 属性 ● 控制 VPN 路由信息收发的路由策略 ●（可选）VPN 实例 IPv4 地址族中允许的最大路由数
2	PE 上接入 CE 的接口的 IP 地址
3	PE 的 AS 号
4	ASBR 之间接口的 IP 地址
5	ASBR 上使用的路由策略
6	PE-CE 间采用的路由交换方式：静态路由、RIP、OSPF、IS-IS 或 BGP
7	PE 与 ASBR 间用来建立 IBGP 对等体的 IP 地址和接口

说明

跨域 OptionC 方式中，在 ASBR 之间不要使能 LDP。

因为，如果在 ASBR 之间的接口上使能 LDP，则会在 ASBR 之间建立 LDP Session。这样，ASBR 建立 Egress LSP 并向上游的 ASBR 发送 Mapping 消息，上游收到该 Mapping 消息后建立 Transit LSP。在 BGP 路由很大的情况下，在 ASBR 之间的接口上使能 LDP，会大量占用 LDP 标签。

2.8.2 使能标签 IPv4 路由交换

跨域 VPN-OptionC 场景下，需要建立一条跨域的 BGP LSP，要求 BGP 对等体能够交换标签 IPv4 路由。

操作步骤

- 配置 PE 侧
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** *ipv4-address* **label-route-capability**，配置与本 AS 的 ASBR 之间能够交换带标签的 IPv4 路由。
- 配置 ASBR 侧
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入连接对端 ASBR 的接口视图。
  3. 执行命令 **ip address** *ip-address* { *mask* | *mask-length* }，配置接口 IP 地址。
  4. 执行命令 **mpls**，使能 MPLS 能力。
  5. 执行命令 **quit**，退回系统视图。
  6. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  7. 执行命令 **peer** *ipv4-address* **label-route-capability**，配置与本 AS 的 PE 之间能够交换带标签的 IPv4 路由。

     在 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 与对端 ASBR 建立 EBGP 对等体关系，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 RFC3107（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。

     缺省情况下，BGP 对等体不能处理标签 IPv4 路由。
  8. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，将对端 ASBR 配置为 EBGP 对等体。
  9. （可选）执行命令 **peer** { *ipv4-address* | *group-name* } **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。

     通常情况下，EBGP 对等体之间必须具有直连的物理链路，如果不满足这一要求，则必须使用 **peer ebgp-max-hop** 命令允许它们之间经过多跳建立 TCP 连接。
  10. 执行命令 **peer** *ipv4-address* **label-route-capability** [**check-tunnel-reachable**]，配置与对端 ASBR 之间能够交换带标签的 IPv4 路由。

- 如果使能 check-tunnel-reachable 功能，则当路由隧道不可达时向邻居发布 IPv4 单播路由，当隧道可达时发布标签路由。在 VPN 场景下，这样可以防止出现 PE 间建立 MP-EBGP 对等体成功而其中一段 LSP 建立失败，造成数据转发失败的情况。
- 如果不使能 check-tunnel-reachable 功能，则不论引入路由隧道是否可达均发布标签路由。

----结束

2.8.3 配置路由策略控制标签分配

跨域 BGP LSP 需要配置路由策略来控制标签的分配，对于向本 AS 的 PE 发布的路由，如果是带标签的 IPv4 路由，为其重新分配 MPLS 标签；对于从本 AS 的 PE 接收的路由，在向对端 ASBR 发布时，分配 MPLS 标签。

操作步骤

- 创建路由策略

  在 ASBR 上进行如下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **route-policy** *policy-name1* **permit node** *node*，创建用于本端 PE 的路由策略。

     对于从对端的 ASBR 接收的带标签的 IPv4 路由，在向本 AS 的 PE 发布时，为其重新分配 MPLS 标签。
  3. 执行命令 **if-match mpls-label**，匹配带标签的 IPv4 路由。
  4. 执行命令 **apply mpls-label**，为 IPv4 路由分配标签。
  5. 执行命令 **quit**，退回系统视图。
  6. 执行命令 **route-policy** *policy-name2* **permit node** *node*，创建用于对端 ASBR 的路由策略。

     对于从本 AS 的 PE 接收的路由，在向对端 ASBR 发布时，分配 MPLS 标签。
  7. 执行命令 **apply mpls-label**，为 IPv4 路由分配标签。

- 应用路由策略

  路由策略应用在 ASBR 上。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** *ipv4-address* **route-policy** *policy-name1* **export**，配置向本端 PE 发布路由时应用的路由策略。
  4. 执行命令 **peer** *ipv4-address* **route-policy** *policy-name2* **export**，配置向对端 ASBR 发布路由时应用的路由策略。

----结束

2.8.4 PE 间建立 MP-EBGP 对等体关系

MP-EBGP 通过在 BGP 中引入扩展团体属性，使其能够在 PE 设备之间传播 VPNv4 路由。不同 AS 间的 PE 通常不是直连的，为了在它们之间建立 EBGP 连接，需要配置 PE 之间允许的最大跳数。

操作步骤

- 在 ASBR 或 PE 上配置将域内 PE 上用于 BGP 会话的 Loopback 接口地址发布给其他 AS 的 ASBR，进而发布给对端 PE。

  说明

  在跨域 VPN-OptionC 组网中，如果需要使用跨域的 TE 隧道传输流量，则必须在 PE 上执行以下配置，将用于 BGP 会话的 Loopback 接口地址发布给对端 PE。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **network** *ip-address* [*mask* | *mask-length*] [**route-policy** *route-policy-name*]，发布域内 PE 上用于 BGP 会话的 Loopback 接口地址。
- 在接入 CE 的 PE 上进行如下配置。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，指定对端 PE 为自己的 EBGP 对等体。
  4. 执行命令 **peer** *ipv4-address* **ebgp-max-hop** [*hop-count*]，配置建立 EBGP 对等体允许的最大跳数。

     不同 AS 间的 PE 通常不是直连的，为了在它们之间建立 EBGP 连接，需要配置 PE 之间允许的最大跳数；并且，应保证 PE 之间可达。
  5. 执行命令 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 子地址族视图。
  6. 执行命令 **peer** *ipv4-address* **enable**，使能与对端 PE 交换 VPNv4 路由的能力。
- （可选）配置路由反射器 RR（Route Reflector）。

  当使用路由反射器 RR 通告 VPNv4 路由的情况时，需要在 RR 上进行如下配置。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 子地址族视图。
  4. 执行命令 **peer** *ipv4-address* **enable**，使能与对端 RR 交换 VPNv4 路由的能力。
  5. 执行命令 **peer** *ipv4-address* **next-hop-invariable**，配置向 EBGP 对等体发送路由时不改变下一跳。

----结束

2.8.5 配置 CE 和 PE 间路由交换

PE 与 CE 之间的路由协议可以是：BGP、静态路由、IGP。

操作步骤

步骤 1 请参见配置 PE 和 CE 间路由交换。

----结束

2.8.6 检查配置结果

配置了跨域 VPN-OptionC 后，可以查看到所有 BGP 对等体关系的建立情况、PE 或 ASBR 上的 VPNv4 路由信息和 ASBR 上的 IPv4 路由标签信息。

前提条件

已经完成跨域 VPN-OptionC 功能的所有配置。

操作步骤

- 使用 **display bgp vpnv4 all peer** 命令在 PE 上检查 BGP 对等体关系的建立情况。
- 使用 **display bgp vpnv4 all routing-table** 命令在 PE 或 ASBR 上检查 VPN-IPv4 路由表。
- 使用 **display bgp routing-table label** 命令在 ASBR 上查看 IPv4 路由的标签信息。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 PE 上检查 VPN 路由表。

----结束

任务示例

在 PE 上执行命令 **display bgp vpnv4 all peer**，可以查看到 PE 间的 EBGP 对等体关系状态均为“Established”。

在 PE 和 ASBR 上执行命令 **display bgp vpnv4 all routing-table**，可以看到 PE 有 BGP VPNv4 路由和 BGP VPN 实例路由，ASBR 上没有。

在 ASBR 上执行命令 **display bgp routing-table label**，可以看到 IPv4 路由的标签信息。

在 PE 上执行命令 **display ip routing-table vpn-instance**，可以看到 PE 的 VPN 路由表中有到所有相关 CE 的 VPN 路由。

2.9 配置跨域 VPN-OptionC（方案二）

通过为带标签的公网 BGP 路由建立 LDP LSP，在不同 AS 的 PE 之间建立 Multihop 方式的 EBGP 连接，交换 VPNv4 路由。

2.9.1 建立配置任务

在配置跨域 VPN-OptionC（方案二）前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

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。

当每个 AS 都有大量的 VPN 路由需要交换时，可选择跨域 VPN-OptionC 方式，防止 ASBR 成为阻碍网络进一步扩展的瓶颈。实现跨域 VPN-OptionC 方式可以采用以下两种方式：

- 方案一：当本端 ASBR 从对端的 ASBR 学到对端 AS 域内的带标签 BGP 公网路由后，通过策略为该路由分配标签，发布给支持标签能力的 IBGP 邻居 PE，从而建立一条完整的公网 LSP。
- 方案二：在 PE 和 ASBR 之间不用配置 IBGP 邻居。当 ASBR 从对端的 ASBR 学到对端 AS 域的带标签 BGP 公网路由后，通过将本端 ASBR 上的 BGP 路由引入 IGP

协议中，并触发为带标签的公网 BGP 路由建立 LDP LSP，从而建立一条完整的公网 LSP。

当 ASBR 上需要接入大量的 PE 设备时，推荐采用 VPN-OptionC（方案二），该方式能够简化用户的配置。

前置任务

在配置 OptionC 方式跨域 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 基本能力和 MPLS LDP
- 在与 CE 相连的 PE 上配置 VPN 实例，并配置接口与 VPN 实例关联
- 在 CE 上配置接入 PE 的接口的 IP 地址
- 用于过滤带标签的公网 BGP 路由的 IP 前缀列表名称

数据准备

在配置 OptionC 方式跨域 VPN 之前，需准备以下数据。

序号	数据
1	PE 上配置 VPN 实例所需的数据，包括： ● VPN 实例名称 ●（可选）VPN 实例的描述信息 ● VPN 实例 IPv4 地址族的 RD、VPN Target 属性 ●（可选）控制 VPN 路由信息收发的路由策略 ●（可选）VPN 实例 IPv4 地址族中允许的最大路由数
2	PE 上接入 CE 的接口的 IP 地址
3	各 AS 的 AS 号
4	ASBR 之间接口的 IP 地址
5	ASBR 上使用的路由策略
6	PE 和 CE 之间的路由交换方式
7	（可选）用于过滤带标签的公网 BGP 路由的 IP 前缀列表名称

说明

跨域 OptionC 方式中，在 ASBR 之间不要使能 LDP。

因为，如果在 ASBR 之间的接口上使能 LDP，则会在 ASBR 之间建立 LDP Session。这样，ASBR 建立 Egress LSP 并向上游的 ASBR 发送 Mapping 消息，上游收到该 Mapping 消息后建立 Transit LSP。在 BGP 路由很大的情况下，在 ASBR 之间的接口上使能 LDP，会大量占用 LDP 标签。

2.9.2 ASBR 间建立 EBGP 对等体关系

ASBR 间建立 EBGP 对等体关系用于发布 PE 的 Loopback 接口路由。

操作步骤

- 请在 ASBR 上进行以下配置。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入连接对端 ASBR 的接口视图。
  3. 执行命令 **ip address** *ip-address* { *mask* | *mask-length* }，配置接口 IP 地址。
  4. 执行命令 **quit**，退回系统视图。
  5. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  6. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，将对端 ASBR 配置为 EBGP 对等体。
  7. （可选）执行命令 **peer** { *ipv4-address* | *group-name* } **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。

     通常情况下，EBGP 对等体之间必须具有直连的物理链路，如果不满足这一要求，则必须使用 **peer ebgp-max-hop** 命令允许它们之间经过多跳建立 TCP 连接。

----结束

2.9.3 将域内 PE 的路由发布给远端 PE

将域内 PE 的 Loopback 接口路由发布给远端 PE，用以建立 PE 之间的 MP-EBGP 关系。

操作步骤

- 将域内 PE 的 Loopback 地址发布给对端的 ASBR

  请在 ASBR 上进行以下配置。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **network** *ip-address* [*mask* | *mask-length*]，将域内 PE 的 Loopback 地址发布给对端的 ASBR。
  4. 执行命令 **quit**，退回系统视图。
- 将 BGP 路由引入到 IGP

  请在对端 ASBR 上进行以下配置。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **ospf** *process-id*，进入 OSPF 视图。
  3. 执行命令 **import-route bgp** [**cost** *cost*] [**route-policy** *route-policy-name*]，将 BGP 路由引入到 IGP。
  4. 执行命令 **quit**，退回系统视图。

----结束

2.9.4 使能标签 IPv4 路由交换能力

为了建立跨域的 BGP LSP，ASBR 之间需要使能交换标签 IPv4 路由。

操作步骤

- 创建路由策略

  在 ASBR 上进行如下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **route-policy** *route-policy-name* **permit node** *seq-number*，创建与对端 ASBR 发布路由时应用的路由策略。
  3. 执行命令 **apply mpls-label**，为 IPv4 路由分配标签。
  4. 执行命令 **quit**，退回系统视图。

- 应用路由策略

  路由策略应用在 ASBR 上。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** *ipv4-address* **route-policy** *route-policy-name* **export**，配置向对端 ASBR 发布路由时应用的路由策略。
  4. 执行命令 **quit**，退回系统视图。

- 使能标签 IPv4 路由交换

  在 ASBR 上进行如下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **interface** *interface-type interface-number*，进入连接对端 ASBR 的接口视图。
  3. 执行命令 **mpls**，使能 MPLS 能力。
  4. 执行命令 **quit**，退回系统视图。
  5. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  6. 执行命令 **peer** *ipv4-address* **label-route-capability** [**check-tunnel-reachable**]，配置与对端 ASBR 之间能够交换带标签的 IPv4 路由。
     - 如果使能 check-tunnel-reachable 功能，则当路由隧道不可达时向邻居发布 IPv4 单播路由，当隧道可达时发布标签路由。在 VPN 场景下，这样可以防止出现 PE 间建立 MP-EBGP 对等体成功而其中一段 LSP 建立失败，造成数据转发失败的情况。
     - 如果不使能 check-tunnel-reachable 功能，则不论引入路由隧道是否可达均发布标签路由。

----结束

2.9.5 为带标签的公网 BGP 路由建立 LDP LSP

通过在 ASBR 上使能 LDP 为 BGP 分标签，可以为通过 IP 前缀列表过滤的带标签的公网 BGP 路由建立 LDP LSP。

操作步骤

- 为通过 IP 前缀列表过滤的带标签的公网 BGP 路由建立 LDP LSP

  请在 ASBR 上进行以下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **mpls**，进入 MPLS 视图。

3. 执行命令 **lsp-trigger bgp-label-route** [**ip-prefix** *ip-prefix-name*]，为通过 IP 前缀列表过滤的带标签的公网 BGP 路由建立 LDP LSP。

----结束

2.9.6 PE 间建立 MP-EBGP 对等体关系

MP-EBGP 通过在 BGP 中引入扩展团体属性，使其能够在 PE 设备之间传播 VPNv4 路由。不同 AS 间的 PE 通常不是直连的，为了在它们之间建立 EBGP 连接，需要配置 PE 之间允许的最大跳数。

操作步骤

- 在 PE 上进行如下配置。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，指定对端 PE 为自己的 EBGP 对等体。
  4. 执行命令 **peer** *ipv4-address* **connect-interface** *interface-type interface-number ipv4-source-address*，指定发送 BGP 报文的源接口。
  5. 执行命令 **peer** *ipv4-address* **ebgp-max-hop** [*hop-count*]，配置建立 EBGP 对等体允许的最大跳数。
  6. 执行命令 **ipv4-family vpnv4**，进入 BGP-VPNv4 子地址族视图。
  7. 执行命令 **peer** *ipv4-address* **enable**，使能与对端 PE 交换 VPNv4 路由的能力。

----结束

2.9.7 配置 CE 和 PE 间路由交换

PE 与 CE 之间的路由协议可以是：BGP、静态路由、IGP。

操作步骤

- 配置 PE 侧
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
  4. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，将 CE 配置为 VPN 私网对等体。
  5. （可选）执行命令 **peer** { *ipv4-address* | *group-name* } **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。
  6. （可选）执行命令 **network** *ip-address mask*，发布到本地 CE 的直连路由。
  7. （可选）执行命令 **peer** *ip-address* **allow-as-loop** [*number*]，允许路由环路。
  8. （可选）执行命令 **peer** *ip-address* **substitute-as**，使能 BGP 的 AS 号替换功能。
- 配置 CE 侧
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。

3. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，将 PE 配置为对等体。
4. （可选）执行命令 **peer** { *ipv4-address* | *group-name* } **ebgp-max-hop** [*hop-count*]，配置 EBGP 连接的最大跳数。
5. 执行命令 **import-route** { **direct** | **static** | **rip** [*process-id*] | **ospf** *process-id* | **isis** *process-id* } [**med** *med* | **route-policy** *route-policy-name*]*，引入本站点的路由。

----结束

2.9.8 检查配置结果

配置了跨域 VPN-OptionC（方案二）后，可以查看到所有 BGP 对等体关系的建立情况、PE 上的 VPNv4 路由信息和 ASBR 上的 IPv4 路由标签信息。

前提条件

已经完成跨域 VPN-OptionC（方案二）功能的所有配置。

操作步骤

- 使用 **display bgp vpnv4 all peer** 命令在 PE 上检查 BGP 对等体关系的建立情况。
- 使用 **display bgp vpnv4 all routing-table** 命令在 PE 或 ASBR 上检查 VPN-IPv4 路由表。
- 使用 **display bgp routing-table label** 命令在 ASBR 上查看 IPv4 路由的标签信息。
- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 PE 上检查 VPN 路由表。
- 使用 **display mpls route-state** [**vpn-instance** *vpn-instance-name*] [{ **exclude** | **include** } { **idle** | **ready** | **settingup** } * | *destination-address mask-length*] [**verbose**]命令在 ASBR 上检查路由和 LSP 的对应情况。
- 使用 **display ip routing-table** 命令在 ASBR 上检查路由表。
- 使用 **display mpls lsp** [**vpn-instance** *vpn-instance-name*] [**protocol ldp**] [{ **exclude** | **include** } *ip-address mask-length*] [**outgoing-interface** *interface-type interface-number*] [**in-label** *in-label-value*] [**out-label** *out-label-value*] [**lsr-role** { **egress** | **ingress** | **transit** }] [**verbose**]命令在 ASBR 上检查 LDP LSP 的建立情况。

----结束

任务示例

在 PE 上执行命令 **display bgp vpnv4 all peer**，可以查看到 PE 间的 EBGP 对等体关系状态均为“Established”。

在 PE 和 ASBR 上执行命令 **display bgp vpnv4 all routing-table**，可以看到 PE 有 BGP VPNv4 路由和 BGP VPN 实例路由，ASBR 上没有。

在 ASBR 上执行命令 **display bgp routing-table label**，可以看到 IPv4 路由的标签信息。

在 PE 上执行命令 **display ip routing-table vpn-instance** *vpn-instance-name*，可以看到 PE 的 VPN 路由表中有到所有相关 CE 的 VPN 路由。

在 ASBR 上执行命令 **display mpls route-state verbose**，可以看到路由类型为“L”，即带标签的公网 BGP 路由。

在 ASBR 上执行命令 **display ip routing-table**，可以看到远端 PE 的路由为带标签的公网 BGP 路由：Routing Table 为“Public”，协议类型为“BGP”，标签值不为零。

```
[ASBR] display ip routing-table 4.4.4.9 verbose
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Table : Public
Summary Count : 1

Destination    : 4.4.4.9/32
     Protocol    : EBGP            Process ID    : 0
   Preference    : 255                  Cost    : 1
      NextHop    : 192.1.1.2       Neighbour: 192.1.1.2
        State    : Active Adv            Age    : 00h12m53s
          Tag    : 0               Priority    : low
     Label    : 15360             QoSInfo    : 0x0
   IndirectID: 0x0
 RelayNextHop    : 0.0.0.0         Interface    : GE2/0/0
     TunnelID    : 0x6002006           Flags    :  D
```

在 ASBR 上执行命令 **display mpls lsp**，可以看到 ASBR 和远端 PE 之间建立了一条 LDP LSP，并且在 PE 上可以看到到达对端 PE 的 LDP Ingress LSP。

```
[ASBR] display mpls lsp protocol ldp include 4.4.4.9 32 verbose
------------------------------------------------------------------------------
                LSP Information: LDP LSP
------------------------------------------------------------------------------
  No                  :  1
  VrfIndex            :
  Fec                 :  4.4.4.9/32
  Nexthop             :  192.1.1.2
  In-Label            :  1024
  Out-Label           :  NULL
  In-Interface        :  ----------
  Out-Interface       :  ----------
  LspIndex            :  13313
  Token               :  0x0
  FrrToken            :  0x0
  LsrType             :  Egress
  Outgoing token      :  0x6002006
  Label Operation     :  POPGO
  Mpls-Mtu            :  ------
  TimeStamp           :  15829sec
  Bfd-State           :  ---
  BGPKey              :  ---
```

2.10 配置 HoVPN

HoVPN 是具有层次化的 VPN 网络，由多个 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，以降低对 PE 设备的性能要求。

2.10.1 建立配置任务

在配置 HoVPN 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

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。

前置任务

在配置 HoVPN 之前，需完成以下任务：

配置基本 BGP/MPLS IP VPN

数据准备

在配置 HoVPN 之前，需准备以下数据。

序号	数据
1	UPE 和 SPE 的对应关系
2	向 UPE 发送缺省路由的 VPN 实例的名称

2.10.2 指定 UPE

配置 UPE 的前提是 UPE 和 SPE 之间建立了 VPNv4 邻居。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** { *ipv4-address* | *group-name* } **as-number** *as-number*，指定 UPE 作为自己的 BGP 对等体。

步骤 4 执行命令 **ipv4-family vpnv4** [**unicast**]，进入 BGP-VPNv4 子地址族视图。

步骤 5 执行命令 **peer** { *ipv4-address* | *group-name* } **enable**，使能对等体交换 BGP-VPNv4 路由信息。

步骤 6 执行命令 **peer** { *ipv4-address* | *group-name* } **upe**，将对等体指定为自己的 UPE。

----结束

2.10.3 发布 VPN 实例的缺省路由

SPE 向 UPE 发布一条下一跳地址为本地地址的缺省路由，用以指导 UPE 上的 VPN 报文转发。

背景信息

在 SPE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family vpnv4**，进入 BGP-VPNv4 子地址族视图。

步骤 4 执行命令 **peer** { *ipv4-address* | *group-name* } **default-originate vpn-instance** *vpn-instance-name*，向 UPE 发送指定 VPN 实例的缺省路由。

执行此命令后，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。

----结束

2.10.4 检查配置结果

配置了 HoVPN 后，可以查看到本端 CE 上没有到对端 CE 接口网段的路由，但有一条下一跳为 UPE 的缺省路由。

前提条件

已经完成 HoVPN 功能的所有配置。

操作步骤

- 使用 **display ip routing-table** 命令在 CE 上查看路由表。

----结束

任务示例

在下层 CE 上执行命令 **display ip routing-table**，下层 CE 上没有到对端 CE 接口网段的路由，但有一条下一跳为下层 PE（UPE）的缺省路由。

```
<CE> display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 5        Routes : 5

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        0.0.0.0/0   BGP     255  0           D   10.1.1.2        GigabitEthernet1/0/0
       10.1.1.0/24  Direct  0    0           D   10.1.1.1        GigabitEthernet1/0/0
       10.1.1.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0GigabitEthernet1/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

2.11 配置 Multi-VPN-Instance CE

通过在 CE 上配置 OSPF 多实例实现局域网不同业务的隔离。

2.11.1 建立配置任务

在配置 Multi-VPN-Instance CE 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

Multi-VPN-Instance CE 在局域网内使用，通过在 CE 设备上使用 OSPF 多实例实现不同业务的隔离。

一个 OSPF 进程只能属于一个 VPN 实例；一个 VPN 实例可以运行多个 OSPF 进程。

Multi-VPN-Instance CE 可以看作一种通过路由隔离实现业务隔离的组网方案，配置上并没有特殊之处，但需要禁止路由环路检查。

前置任务

在配置 Multi-VPN-Instance CE 之前，需完成以下任务：

- 在多实例 CE 及其接入的 PE 上配置 VPN 实例（每个业务配置一个 VPN 实例）。
- 配置局域网相关接口的链路层协议和网络层协议，将局域网接入到多实例 CE 上。每个业务使用一个接口接入多实例 CE。
- 在多实例 CE 的每个接口及 PE 接入多实例 CE 的接口上都绑定相应的 VPN 实例，并配置 IP 地址。

数据准备

在配置 Multi-VPN-Instance CE 之前，需准备以下数据。

序号	数据
1	各业务使用的 OSPF 进程所对应的 VPN 实例名称
2	各业务使用的 OSPF 进程的进程号和 Router ID
3	各 OSPF 进程发布的路由

2.11.2 在多实例 CE 接入的 PE 上配置 OSPF 多实例

不同的业务需要配置到不同的实例下，同时需要使用不同的 OSPF 进程号。

背景信息

在多实例 CE 接入的 PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** *process-id* [**router-id** *router-id*] **vpn-instance** *vpn-instance-name*，配置 OSPF 多实例。

不同的业务使用的 OSPF 进程号不同。*router-id* 可以相同，也可以不同。

步骤 3 执行命令 **area** *area-id*，进入 OSPF 区域视图。

步骤 4 执行命令 **network** *ip-address wildcard-mask*，发布接入多实例 CE 的接口的 IP 地址。

步骤 5 执行命令 **quit**，退回 OSPF 视图。

步骤 6 执行命令 **import-route bgp**，引入 BGP 路由。

步骤 7 执行命令 **quit**，退回系统视图。

步骤 8 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 9 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。

步骤 10 执行命令 **import-route ospf** *process-id*，引入 OSPF 多实例路由。

----结束

2.11.3 在多实例 CE 上配置 OSPF 多实例

在多实例 CE 上配置 OSPF 多实例的进程号需要与 PE 上的进程号一致。

背景信息

在多实例 CE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** *process-id* [**router-id** *router-id*] **vpn-instance** *vpn-instance-name*，配置 OSPF 多实例。

该进程号与 PE 上的进程号对应。

步骤 3 执行命令 **area** *area-id*，进入 OSPF 区域视图。

步骤 4 执行命令 **network** *ip-address wildcard-mask*，发布连接 PE 的接口的 IP 地址。

说明

如果多实例 CE 不是通过本进程的 OSPF 多实例学习局域网路由，还需要在本进程的 OSPF 实例中引入局域网的路由。

----结束

2.11.4 取消多实例 CE 上的路由环路检查

如果进行路由环路检查，则 CE 会丢弃来自 PE 的 DN 位置位的路由。

背景信息

在 MCE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ospf** *process-id* [**router-id** *router-id*] **vpn-instance** *vpn-instance-name*，进入 OSPF 多实例视图。

步骤 3 执行命令 **vpn-instance-capability simple**，不进行路由环路检查。

----结束

2.11.5 检查配置结果

配置了 Multi-VPN-Instance CE 后，可以查看多实例 CE 上的 VPN 路由表中对于每一种业务，CE 上都有到局域网的路由及到远端站点的路由。

前提条件

已经完成 Multi-VPN-Instance CE 功能的所有配置。

操作步骤

- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* [**verbose**]命令在多实例 CE 上查看 VPN 路由表。

----结束

任务示例

#配置成功后，在多实例 CE 执行命令 **display ip routing-table vpn-instance** 查看 VPN 路由表，可看到对于每一种业务，多实例 CE 上都有到局域网的路由及到远端站点的路由。

```
[MCE] display ip routing-table vpn-instance vpna
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpna
         Destinations : 8        Routes : 8
  Destination/Mask  Proto  Pre  Cost      Flags NextHop         Interface
       10.1.1.0/24 O_ASE 150 1            D 192.1.1.1      Pos1/0/0
       10.1.1.1/32 O_ASE 150 1            D 192.1.1.1      Pos1/0/0
       10.3.1.0/24  Direct 0    0             D  10.3.1.2        Pos3/0/0
       10.3.1.1/32  Direct 0    0             D  10.3.1.1        Pos3/0/0
       10.3.1.2/32  Direct 0    0             D  127.0.0.1       InLoopBack0Pos3/0/0
      192.1.1.0/24  Direct 0    0             D  192.1.1.2        Pos1/0/0
      192.1.1.1/32  Direct 0    0             D  192.1.1.1        Pos1/0/0
      192.1.1.2/32  Direct 0    0             D  127.0.0.1       InLoopBack0Pos1/0/0
```

2.12 配置 VPN 与 Internet 互联

一般 VPN 内的用户只能相互通信，不能与 Internet 用户通信，也不能接入 Internet。如果 VPN 的各个 site 需要访问 internet，需要配置 VPN 与 Internet 互联。

2.12.1 建立配置任务

在配置 VPN 与 Internet 互联前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在已搭建的 VPN 网络中增加一些软件配置，使 VPN 用户可以访问 Internet 资源。

前置任务

在配置 VPN 与 Internet 互联之前，需完成以下任务：

- 搭建 VPN 网络。

数据准备

在配置 VPN 与 Internet 互联之前，需准备以下数据。

序号	数据
1	VPN 实例名称
2	静态路由目的地址

2.12.2 CE 上配置静态路由

CE 上配置静态路由，用以指导 VPN 到 Internet 报文的转发。

背景信息

在 CE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } { *interface-type interface-number* [*nexthop-address*] | *nexthop-address* } [**preference** *preference* | **tag** *tag*]* [**description** *text*]，配置到公网目的地址的静态路由。

参数 *ip-address* 可以配置成公网目的 IP 地址，也可以配置成全零 0.0.0.0（即配置默认路由，其掩码也为全零）。出接口为与 PE 相连的接口；下一跳为 PE 上与本 CE 相连的接口的 IP 地址。

说明

如果 CE 与 PE 之间用以太网相连，则必须指定下一跳 IP 地址。

----结束

2.12.3 PE 上配置私网静态路由

PE 上配置静态路由，用以指导 VPN 到 Internet 报文的转发。

背景信息

在 PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static vpn-instance** *vpn-source-name destination-address* { *mask* | *mask-length* } *nexthop-address* **public** [**preference** *preference* | **tag** *tag*]* [**description** *text*]，配置从 VPN 用户到 Internet 的静态路由，并指定下一跳的地址为公网地址。

----结束

2.12.4 公网目的设备上配置到 VPN 用户的静态路由

配置到 VPN 用户的静态路由，用以指导从 Internet 返回 VPN 报文的转发。

背景信息

在 PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } { *interface-type interface-number* | **vpn-instance** *vpn-instance-name nexthop-address* | *nexthop-address* } [**preference** *preference* | **tag** *tag*]* [**description** *text*]，配置从公网到 VPN 用户的静态路由（下一跳为私网地址）。

说明

如果 CE 与 PE 之间用以太网相连，则必须指定下一跳。

----结束

2.12.5 检查配置结果

配置了 VPN 与 Internet 互联后，可以查看到 VPN 路由表中有到 CE 及公网目的设备的路由、公网目的设备上也有到 CE 的路由。

前提条件

已经完成 VPN 与 Internet 互联功能的所有配置。

操作步骤

- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令在 PE 上查看 VPN 路由表。
- 使用 **display ip routing-table** 命令在 CE 上及公网目的路由器上查看路由表。

----结束

任务示例

在 PE 上执行命令 **display ip routing-table vpn-instance**，可发现 VPN 路由表中有到 CE 及公网目的路由器的路由。

```
<Huawei> display ip routing-table vpn-instance vpn1
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpn1
         Destinations : 7        Routes : 7

  Destination/Mask  Proto  Pre  Cost      Flags NextHop         Interface

        0.0.0.0/0   Static 60   0           RD  100.1.1.2       Pos2/0/0
       10.1.1.0/24  Direct 0    0           D   10.1.1.2        Pos1/0/0
       10.1.1.1/32  Direct 0    0           D   10.1.1.1        Pos1/0/0
       10.1.1.2/32  Direct 0    0           D   127.0.0.1       Pos2/0/0
       10.2.1.0/24  BGP    255  0           RD  3.3.3.3         Pos2/0/0
       10.2.1.1/32  BGP    255  0           RD  3.3.3.3         Pos2/0/0
       10.2.1.2/32  BGP    255  0           RD  3.3.3.3         Pos2/0/0
      100.3.1.1/32  BGP    255  0           D   10.1.1.1        Pos1/0/0
```

在 CE 上执行命令 **display ip routing-table**，发现其有到公网目的路由器的路由，公网目的路由器也有到 CE 的路由。

```
<Huawei> display ip routing-table
```

```
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10

  Destination/Mask  Proto  Pre  Cost        Flags  NextHop         Interface

        1.1.1.1/32  Direct 0    0               D  127.0.0.1       LoopBack1
        2.2.2.2/32  OSPF   10   2               D  100.1.1.2       Pos2/0/0
        3.3.3.3/32  OSPF   10   3               D  100.1.1.2       Pos2/0/0
      100.1.1.0/24  Direct 0    0               D  100.1.1.1       Pos2/0/0
      100.1.1.1/32  Direct 0    0               D  127.0.0.1       Pos1/0/0
      100.1.1.2/32  Direct 0    0               D  100.1.1.2       Pos2/0/0
      100.2.1.0/24  OSPF   10   2               D  100.1.1.2       Pos2/0/0
      100.3.1.0/24  Static 60   0               D  10.1.1.1        Pos1/0/0
      127.0.0.0/8   Direct 0    0               D  127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct 0    0               D  127.0.0.1       InLoopBack0
```

使用 **ping** 命令检查 CE 与公网目的路由器之间的互通性。

```
<Huawei> ping 100.3.1.1
  PING 100.3.1.1: 56  data bytes, press CTRL_C to break
    Reply from 100.3.1.1: bytes=56 Sequence=1 ttl=254 time=62 ms
    Reply from 100.3.1.1: bytes=56 Sequence=2 ttl=254 time=62 ms
    Reply from 100.3.1.1: bytes=56 Sequence=3 ttl=254 time=62 ms
    Reply from 100.3.1.1: bytes=56 Sequence=4 ttl=254 time=62 ms
    Reply from 100.3.1.1: bytes=56 Sequence=5 ttl=254 time=62 ms
  --- 100.3.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 62/62/62 ms
```

2.13 配置私网 IP FRR

VPN site 中的多个 CE 接入到同一台 PE 上时，配置 IP FRR 特性，当 PE 与 CE 之间转发不通时，可以快速将流量切换到另一条 PE 与 CE 相连的链路上。

2.13.1 建立配置任务

在配置私网 IP FRR 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

私网 IP FRR 特性适用于私网中 IP 业务对于丢包、延时非常敏感的情形。配置了私网 IP FRR 特性后，当 PE 去往 CE 的主路由下一跳不可达时，流量快速切换到另一条 PE-CE 链路上，从而保证 IP 业务不中断。

前置任务

在配置私网 IP FRR 之前，需完成以下任务：

- 在路由器上配置路由协议，实现网络互通。
- 搭建 VPN 网络。
- 通过配置不同的度量值，生成两条不等价路由。

数据准备

在配置私网 IP FRR 之前，需要准备以下数据。

序号	数据
1	路由策略的名称
2	VPN 实例名称
3	备份路由的出接口
4	备份路由的下一跳

2.13.2 配置手动私网 IP FRR 功能

通过路由策略指定备份下一跳，当 PE 去往 CE 的主路由下一跳不可达时，私网流量可以走指定的备份下一跳 CE。

背景信息

在 PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **route-policy** *route-policy-name* { **permit** | **deny** } **node** *node*，创建路由策略的节点，并进入路由策略视图。

步骤 3 执行命令 **apply backup-interface** *interface-type interface-number*，设置备份出接口。

步骤 4 （可选）执行命令 **apply backup-nexthop** *ip-address*，设置备份下一跳。

对于 P2P 链路，可以不设置备份下一跳；而对于非 P2P 链路，必须设置备份下一跳。

步骤 5 执行命令 **quit**，退回系统视图。

步骤 6 执行命令 **ip vpn-instance** *vpn-instance-name*，进入 VPN 实例视图。

步骤 7 执行命令 **ipv4-family**，进入 VPN 实例 IPv4 地址族视图。

步骤 8 执行命令 **ip frr route-policy** *route-policy-name*，使能手动私网 IP FRR 功能。

----结束

2.13.3 检查配置结果

配置了私网 IP FRR 后，可以查看到 VPN 路由表中备份出接口和备份下一跳信息。

前提条件

已经完成私网 IP FRR 功能的所有配置。

操作步骤

- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* [*ip-address*] **verbose** 命令查看路由表中备份出接口和备份下一跳信息。

----结束

任务示例

在 PE 上执行命令 **display ip routing-table vpn-instance** *vpn-instance-name* [*ip-address*] **verbose**，可看到相应的 VPN 路由有备份出接口和下一跳。

```
<Huawei> display ip routing-table vpn-instance vpn1 10.5.1.0 verbose
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Table : vpn1
Summary Count : 1
Destination: 10.5.1.0/24
     Protocol: OSPF            Process ID: 1
   Preference: 10                    Cost: 3
      NextHop: 10.1.1.2         Neighbour: 0.0.0.0
        State: Active Adv             Age: 00h00m03s
          Tag: 0                 Priority: low
        Label: NULL               QoSInfo: 0x0
   IndirectID: 0x0
 RelayNextHop: 0.0.0.0          Interface: GigabitEthernet1/0/0
     TunnelID: 0x0                  Flags: D
    BkNextHop: 10.2.1.2       BkInterface: GigabitEthernet2/0/0
      BkLabel: NULL           SecTunnelID: 0x0
 BkPETunnelID: 0x0        BkPESecTunnelID: 0x0
 BkIndirectID: 0x0
```

2.14 配置 VPN FRR

在 CE 多归属组网中，配置 VPN FRR 可以保证 PE 设备发生故障时实现 VPN 业务端到端的快速切换。

2.14.1 建立配置任务

在配置 VPN FRR 前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

适用于 VPN 网络中对于丢包、延时非常敏感的业务。

前置任务

在配置 VPN FRR 之前，需完成以下任务：

- 在路由器上配置路由协议，实现网络互通。
- 通过配置不同的度量值，生成两条不等价路由。
- 搭建 VPN 网络。

注意

VPN 骨干网上创建 LSP 时，P 节点上不建议配置 **lsp-trigger** 命令，使用默认配置即可，否则容易导致 VPN FRR 回切失败。

数据准备

在配置 VPN FRR 之前，需要准备以下数据。

序号	数据
1	路由策略的名称
2	VPN 实例名称
3	备份路由的下一跳

2.14.2 配置手动 VPN FRR

通过路由策略指定备份下一跳，当 PE 之间转发不通时，VPN 流量可以走备份下一跳 PE。

背景信息

在 PE 上配置手动 VPN FRR 功能，需要先配置路由策略指定备份下一跳，然后在私网实例下使能 VPN FRR 功能时应用这个路由策略。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **route-policy** *route-policy-name* { **permit** | **deny** } **node** *node*，创建路由策略的节点，并进入路由策略视图。

步骤 3 执行命令 **apply backup-nexthop** *ip-address*，指定备份下一跳。

步骤 4 执行命令 **quit**，退回系统视图。

步骤 5 执行命令 **ip vpn-instance** *vpn-instance-name*，进入 VPN 实例视图。

步骤 6 执行命令 **ipv4-family**，进入 VPN 实例 IPv4 地址族视图。

步骤 7 执行命令 **vpn frr route-policy** *route-policy-name*，使能 VPN FRR 功能。

----结束

2.14.3 检查配置结果

配置了 VPN FRR 后，可以查看到 VPN 路由表中备份下一跳（PE）、备份隧道和备份标签值。

前提条件

已经完成 VPN FRR 功能的所有配置。

操作步骤

- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* [*ip-address*] **verbose** 命令查看路由表中备份下一跳（PE）、备份隧道和备份标签值。

----结束

任务示例

在配置了 VPN FRR 的 PE 上执行 **display ip routing-table vpn-instance** *vpn-instance-name ip-address* **verbose** 命令，可查看路由的备份下一跳 PE，备份隧道和标签。

2.15 配置路由反射器优化 VPN 骨干层

使用路由反射器，可以减少 PE 之间的 MP-IBGP 连接的数量，既减轻了 PE 的负担，也给维护和管理带来方便。

2.15.1 建立配置任务

在配置路由反射器优化 VPN 骨干层前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

BGP Speaker 从 IBGP 获得的路由不向它的 IBGP 对等体发布。为了让 PE 将自己接入的 VPN 的路由发布给同一个 AS 的 BGP VPNv4 对等体，该 PE 必须与所有对等体建立 IBGP 连接，以便直接交互 VPN 路由信息。即 MP-IBGP 对等体之间需要建立全连接关系。假设在一个 AS 中有 n 台 PE 设备（包括 ASBR），那么需要建立 n(n-1)/2 对 MP-IBGP 对等体。IBGP 对等体数目很多时，需要消耗大量网络资源。

利用 BGP 的路由反射 RR（Route Reflector）可以解决这一问题。在一个 AS 内选择一台设备作为 RR（反射 VPNv4 路由），其它 PE 和 ASBR 做为客户机（Client）。RR 可以是 P 路由、PE 设备、ASBR 设备或者其他设备。

使用路由反射器，可以减少 PE 之间的 MP-IBGP 连接的数量，既减轻了 PE 的负担，也给维护和管理带来方便。

前置任务

在配置路由反射优化 VPN 骨干层之前，需完成以下任务：

- 在 MPLS 骨干网上配置路由协议，实现骨干网设备的 IP 互通
- 在 RR 与所有作为客户机的 PE 之间建立隧道（LSP 或 GRE）

数据准备

在配置 BGP VPNv4 路由反射之前，需准备以下数据。

序号	数据
1	本地 AS 号、对等体的 AS 号

序号	数据
2	建立 TCP 连接使用的接口类型和编号
3	BGP 对等体组名称、对等体的地址

2.15.2 配置客户机 PE 与 RR 建立 MP-IBGP 连接

配置 PE 与 RR 建立 MP-IBGP 连接，以便反射 VPNv4 路由。

背景信息

在所有作为客户机的 PE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** *ipv4-address* **as-number** *as-number*，指定 RR 为 BGP 对等体。

步骤 4 执行命令 **peer** *ipv4-address* **connect-interface** *interface-type interface-number*，指定建立 TCP 连接的接口。该接口上的 IP 地址必须与 MPLS LSR-ID 相等，建议为 Loopback 接口。

步骤 5 执行命令 **ipv4-family vpnv4**，进入 BGP-VPNv4 地址族视图。

步骤 6 执行命令 **peer** *ipv4-address* **enable**，使能与 RR 交换 BGP 的 VPNv4 路由。

----结束

2.15.3 配置 RR 与其所有客户机 PE 建立 MP-IBGP 连接

配置 RR 与其所有客户机 PE 建立 MP-IBGP 连接，以便反射 VPNv4 路由。

背景信息

选择如下方案之一对 RR 进行配置。

操作步骤

- 配置与对等体组建立 MP-IBGP 连接

  将所有客户机 PE 都加入对等体组，并建立与对等体组的 MP-IBGP 连接

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **group** *group-name* [**internal**]，创建 IBGP 对等体组。
  4. 执行命令 **peer** *group-name* **connect-interface** *interface-type interface-number*，指定建立 TCP 连接的接口。该接口上的 IP 地址必须与 MPLS LSR-ID 相等，建议为 Loopback 接口。
  5. 执行命令 **ipv4-family vpnv4**，进入 BGP-VPNv4 地址族视图。

6. 执行命令 **peer** *group-name* **enable**，使能与对等体组交换 BGP 的 VPNv4 路由。
7. 执行命令 **peer** *ip-address* **group** *group-name*，向对等体组中加入对等体。

- 配置与每个对等体建立 MP-IBGP 连接

  在 RR 上重复进行如下步骤 1 ～ 6 的操作，使 RR 与所有的客户机 PE 建立 MP-IBGP 连接。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，指定客户机 PE 为 BGP 对等体。
  4. 执行命令 **peer** *ipv4-address* **connect-interface** *interface-type interface-number*，指定建立 TCP 连接的接口。该接口上的 IP 地址必须与 MPLS LSR-ID 相等，建议为 Loopback 接口。
  5. 执行命令 **ipv4-family vpnv4**，进入 BGP-VPNv4 地址族视图。
  6. 执行命令 **peer** *ipv4-address* **enable**，使能与客户机交换 BGP 的 VPNv4 路由。

----结束

2.15.4 配置 BGP-VPNv4 路由反射功能

使能 BGP-VPNv4 路由反射功能的前提是 RR 与其所有客户机 PE 建立 MP-IBGP 连接。

背景信息

在 RR 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family vpnv4**，进入 BGP-VPNv4 子地址族视图。

步骤 4 使能 RR 的 BGP-VPNv4 路由反射功能：

- 如果 RR 与其所有客户机 PE 建立 MP-IBGP 连接时使用对等体组，则执行命令 **peer** *group-name* **reflect-client**
- 如果 RR 与其所有客户机 PE 建立 MP-IBGP 连接时不是使用对等体组，则重复执行命令 **peer** *ipv4-address* **reflect-client**，使能反射所有客户机的 BGP-VPNv4 路由

步骤 5 执行命令 **undo policy vpn-target**，不对接收的 VPNv4 路由使能 VPN-Target 进行过滤。

步骤 6 （可选）执行命令 **rr-filter** { *extcomm-filter-number* | *extcomm-filter-name* }，配置路由反射器的反射策略。

只有路由目标扩展团体属性可以通过反射策略的 IBGP 路由才被反射。

----结束

2.15.5 检查配置结果

配置了路由反射器优化 VPN 骨干层后，可以在 RR 或客户机 PE 上查看 BGP VPNv4 对等体信息及 VPNv4 路由信息。

前提条件

已经完成路由反射器优化 VPN 骨干层功能的所有配置。

操作步骤

- 使用 **display bgp vpnv4 all peer** [[*ipv4-address*] **verbose**]命令在 RR 上或客户机 PE 上查看 BGP VPNv4 对等体信息。
- 使用 **display bgp vpnv4 all routing-table peer** *ipv4-address* { **advertised-routes** | **received-routes** }或者 **display bgp vpnv4 all routing-table statistics** 命令在 RR 上或客户机 PE 上查看从对等体接收的路由或发布给对等体的 VPNv4 路由信息。
- 使用 **display bgp vpnv4 all group** [*group-name*]命令在 RR 上查看 VPNv4 对等体组信息。

----结束

任务示例

如果配置成功，则：

- 在 RR 或客户机 PE 上执行 **display bgp vpnv4 all peer** 命令，可看到 RR 与所有客户机的 MP-IBGP 连接状态为“Established”。

```
<Huawei> display bgp vpnv4 all peer
 BGP local router ID : 1.1.1.9
 Local AS number : 100
 Total number of peers : 3                 Peers in established state : 3
  Peer            V    AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv
  2.2.2.9         4   100  2        4         0   00:00:31    Established  0
  3.3.3.9         4   100  3        5         0   00:01:23    Established  0
  Peer of IPv4-family for vpn instance :

 VPN-Instance vpna, router ID 1.1.1.9:
  10.1.1.1        4 65410       79       82     0 01:13:29  Established  0
```

- 在 RR 上或客户机 PE 上执行 **display bgp vpnv4 all routing-table peer** 命令，可看到 RR 和客户机之间能互相收发 VPNv4 路由信息。

```
<Huawei> display bgp vpnv4 all routing-table peer 2.2.2.9 received-routes
 BGP Local router ID is 1.1.1.9
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete


 Total Number of Routes: 1

 Route Distinguisher: 100:1


      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  1.1.1.1            2.2.2.9         0          100        0      i
```

- 如果使用对等体，在 RR 上执行 **display bgp vpnv4 all group** 命令，可查看对等体组的成员，且 RR 与对等体成员之间的 BGP 连接状态都为“Established”。

```
<Huawei> display bgp vpnv4 all group vpna
Group in VPNV4:

 BGP peer-group: vpna
 Remote AS: 100
 Authentication type configured: None
 Type : internal
 Configured hold timer value: 180
```

```
Keepalive timer value: 60
Connect-retry timer value: 32
Minimum route advertisement interval is 0 seconds
Connect-interface has been configured
PeerSession Members:
  2.2.2.2

Peer Preferred Value: 0
No routing policy is configured
Peer Members:
 Peer            V          AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv

 2.2.2.2         4         100       13       15     0 00:11:12 Established  0
```

2.16 配置路由反射器优化 VPN 接入层

当 PE 及其接入的多个 CE 位于同一个 AS 时，部署 BGP 路由反射器，可以减少 CE 之间的 IBGP 连接的数量，给维护和管理带来方便。

2.16.1 建立配置任务

在配置路由反射器优化 VPN 接入层前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

当 PE 及其接入的多个 CE 位于同一个 AS 时，可使用 PE 作为 RR（反射 VPN 实例路由），CE 设备做为客户机（Client），这样可减少 CE 之间的 IBGP 连接的数量，给维护和管理带来方便。

前置任务

在配置路由反射优化 VPN 接入层之前，需完成以下任务：

- 在 MPLS 骨干网上配置路由协议，实现骨干网设备的 IP 互通

数据准备

在配置路由反射器优化 VPN 接入层之前，需准备以下数据。

序号	数据
1	本地 AS 号、对等体的 AS 号
2	建立 TCP 连接使用的接口类型和编号
3	BGP 对等体组名称、对等体的地址

2.16.2 配置客户机 CE 与 RR 建立 IBGP 连接

配置客户机 CE 与 RR 建立 IBGP 连接，以便反射 VPNv4 路由。

背景信息

在所有作为客户机的 CE 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **peer** *ipv4-address* **as-number** *as-number*，指定 RR 为 BGP 对等体。

步骤 4 执行命令 **peer** *ipv4-address* **connect-interface** *interface-type interface-number*，指定建立 TCP 连接的接口。该接口的 IP 地址必须为本台设备的 MPLS LSR-ID，推荐用 Loopback 接口建立 TCP 连接。

----结束

2.16.3 配置 RR 与其所有客户机 CE 建立 MP-IBGP 连接

配置 RR 与其所有客户机 CE 建立 MP-IBGP 连接，以便反射 VPNv4 路由到所有客户机 CE。

背景信息

选择如下方案之一对 RR 进行配置。

操作步骤

- 配置与对等体组建立 MP-IBGP 连接
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*,进入 BGP-VPN 实例 IPv4 地址族视图。
  4. 执行命令 **group** *group-name* [**internal**]，创建 IBGP 对等体组。
  5. 执行命令 **peer** *group-name* **connect-interface** *interface-type interface-number*，指定建立 TCP 连接的接口。
  6. 执行命令 **peer** *ip-address* **group** *group-name*，向对等体组中加入对等体。
- 配置与每个对等体建立 MP-IBGP 连接

  重复执行如下步骤 1 ～ 5，为每一个客户机 CE 建立 MP-IBGP 连接。
  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **bgp** *as-number*，进入 BGP 视图。
  3. 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。
  4. 执行命令 **peer** *ipv4-address* **as-number** *as-number*，配置 BGP IPv4 VPN 实例的对等体。
  5. 执行命令 **peer** *ipv4-address* **connect-interface** *interface-type interface-number*，指定建立 TCP 连接的接口。

----结束

2.16.4 配置 BGP-VPN 实例路由反射功能

使能 BGP-VPNv4 路由反射功能的前提是 RR 与其所有客户机 CE 建立 MP-IBGP 连接。

背景信息

在 RR 上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **bgp** *as-number*，进入 BGP 视图。

步骤 3 执行命令 **ipv4-family vpn-instance** *vpn-instance-name*，进入 BGP-VPN 实例 IPv4 地址族视图。

步骤 4 使能 RR 的 BGP-VPN 实例 IPv4 地址族路由反射功能：

- 如果 RR 与其所有客户机 CE 建立 IBGP 连接时使用对等体组，则执行命令 **peer** *group-name* **reflect-client**
- 如果 RR 与其所有客户机 CE 建立 IBGP 连接时不是使用对等体组，则重复执行命令 **peer** *ipv4-address* **reflect-client**，使能反射所有客户机的 IPv4 VPN 路由

步骤 5 （可选）执行命令 **reflect between-clients**，使能客户机之间的路由反射。

缺省情况下，允许客户机之间的路由反射。

如果路由反射器的客户机已经是全连接的，可以使用 **undo reflect between-clients** 命令禁止客户间的反射，以便减少开销。

步骤 6 （可选）执行命令 **reflector cluster-id** *cluster-id*，配置路由反射器的集群 ID。

当一个集群里有多个路由反射器时，可以使用此命令给所有位于同一个集群内的路由反射器配置相同的 *cluster-id*，以避免路由环路。缺省情况下，使用 Router ID 作为集群 ID。

----结束

2.16.5 检查配置结果

配置了路由反射器优化 VPN 接入层后，可以在 RR 上查看 BGP VPN 实例对等体信息、从对等体接收的路由或发布给对等体的 VPNv4 路由信息。

前提条件

已经完成路由反射器优化 VPN 接入层功能的所有配置。

操作步骤

- 使用 **display bgp** [**vpnv4 vpn-instance** *vpn-instance-name*] **peer** [*ipv4-address*] **verbose** 命令在 RR 上查看 BGP VPN 实例对等体信息。
- 使用 **display bgp peer** [*ipv4-address*] **verbose** 命令在客户机 CE 上查看 BGP 对等体信息。

- 使用 **display bgp vpnv4 all routing-table peer** *ipv4-address* { **advertised-routes** | **received-routes** }或者 **display bgp vpnv4 all routing-table statistics** 命令在 RR 上查看从对等体接收的路由或发布给对等体的 VPNv4 路由信息。
- 使用 **display bgp routing-table peer** *ipv4-address* { **advertised-routes** | **received-routes** }或者 **display bgp routing-table statistics** 命令在客户机 CE 上查看从对等体接收的路由或发布给对等体的路由信息。
- 使用 **display bgp vpnv4 vpn-instance** *vpn-instance-name* **group** [*group-name*]命令在 RR 上查看 VPNv4 对等体组信息。
- 使用 **display bgp group** [*group-name*]命令在 CE 上查看 VPNv4 对等体组信息。

----结束

任务示例

如果配置成功，则：

- 在 RR 上执行 **display bgp vpnv4 all peer** 命令，可看到 RR 与所有客户机的 MP-IBGP 连接状态为“Established”。

```
<Huawei> display bgp vpnv4 all peer
 BGP local router ID : 1.1.1.9
 Local AS number : 100
 Total number of peers : 3                 Peers in established state : 3
  Peer          V    AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv
  2.2.2.9       4   100  2        4         0   00:00:31    Established  0
  3.3.3.9       4   100  3        5         0   00:01:23    Established  0
  Peer of IPv4-family for vpn instance :

 VPN-Instance vpna, router ID 1.1.1.9:
  10.1.1.1        4 65410       79       82     0 01:13:29  Established  0
```

- 在客户机 CE 上执行 **display bgp peer** 命令，可看到客户机 CE 与 RR 的 IBGP 连接状态为“Established”。

```
<Huawei> display bgp peer
 BGP Local router ID : 1.2.3.4
 local AS number : 10
 Total number of peers : 2                 Peers in established state : 1
  Peer            V    AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv

  1.1.1.1         4   100        0        0     0 00:00:07        Idle       0
  1.2.5.6         4   200       32       35     0 00:17:49 Established       0
```

- 在 RR 上执行 **display bgp vpnv4 all routing-table peer** 命令，可看到 RR 发布给客户机 CE 的路由信息或客户机发布给 RR 的路由信息。

```
<Huawei> display bgp vpnv4 all routing-table peer 2.2.2.9 received-routes
 BGP Local router ID is 1.1.1.9
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete


 Total Number of Routes: 1

 Route Distinguisher: 100:1

      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  1.1.1.1            2.2.2.9         0          100        0      i
```

- 在客户机 CE 上执行 **display bgp routing-table peer** *ipv4-address* { **advertised-routes** | **received-routes** 或者 **display bgp vpnv4 all routing-table statistics** 命令，可看到客户机发布给 RR 的路由信息或 RR 发布给客户机的路由信息。

```
<Huawei> display bgp routing-table peer 1.1.1.1 accepted-routes

 BGP Local router ID is 10.1.1.2
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete


 Total Number of Routes: 2
      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

   i  1.1.1.1/32         1.1.1.1         0          100        0      ?
 *>i  10.1.1.0/24        1.1.1.1         0          100        0      ?
<Huawei> display bgp vpnv4 all routing-table statistics

 Total number of routes from all PE: 4

 VPN-Instance vpn1, router ID 1.1.1.9:

 Total Number of Routes: 4

 VPN-Instance vpn2, router ID 1.1.1.9:

 Total Number of Routes: 0
```

- 如果使用对等体，在 RR 上执行 **display bgp vpnv4 all group** 命令，可查看对等体组的成员，且 RR 与对等体成员之间的 BGP 连接状态都为“Established”。

```
<Huawei> display bgp vpnv4 all group vpna
Group in VPNV4:

 BGP peer-group: vpna
 Remote AS: 100
 Authentication type configured: None
 Type : internal
 Configured hold timer value: 180
 Keepalive timer value: 60
 Connect-retry timer value: 32
 Minimum route advertisement interval is 0 seconds
 Connect-interface has been configured
 PeerSession Members:
   2.2.2.2

 Peer Preferred Value: 0
 No routing policy is configured
 Peer Members:
  Peer            V          AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv

  2.2.2.2         4         100       13       15     0 00:11:12 Established  0
```

2.17 维护 BGP/MPLS IP VPN

维护 BGP/MPLS IP VPN 包括查看 L3VPN 流量、监测网络连通性、复位 BGP 连接。

2.17.1 查看所有 IPv4 VPN 实例的综合路由统计信息

VPN 实例的综合路由统计信息就是所有 VPN 实例的路由统计信息之和。

操作步骤

- 执行命令 **display ip routing-table all-vpn-instance statistics**，查看所有 IPv4 VPN 实例的综合路由统计信息。

----结束

2.17.2 监控 BGP/MPLS IP VPN 的运行状态

监控 BGP/MPLS IP VPN 的运行状态包括：查看 VPN-instance 信息、查看 VPNv4 对等体信息、查看 BGP 对等体日志信息等。

背景信息

在日常维护工作中，可以在任意视图下选择执行以下命令，了解 BGP/MPLS IP VPN 的运行情况。

操作步骤

- 使用 **display ip routing-table vpn-instance** *vpn-instance-name* 命令查看与 VPN-instance 相关联的 IP 路由表。
- 使用 **display ip vpn-instance** [**verbose**] [*vpn-instance-name*]命令查看 VPN-instance 信息。
- 使用 **display bgp** [**vpnv4** { **all** | **vpn-instance** *vpn-instance-name* }] **routing-table label** 命令查看 BGP 路由表中的标签路由信息。
- 使用 **display bgp vpnv4** { **all** | **route-distinguisher** *route-distinguisher* | **vpn-instance** *vpn-instance-name* } **routing-table** *ipv4-address* [*mask* | *mask-length*]命令查看 BGP VPNv4 具体路由表项。
- 使用 **display bgp vpnv4** { **all** | **route-distinguisher** *route-distinguisher* | **vpn-instance** *vpn-instance-name* } **routing-table statistics** 命令查看 BGP VPNv4 路由表的统计信息。
- 使用 **display bgp vpnv4** { **all** | **route-distinguisher** *route-distinguisher* | **vpn-instance** *vpn-instance-name* } **routing-table** 命令查看 BGP VPNv4 路由表信息。
- 使用 **display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **group** [*group-name*] 命令查看 VPNv4 的 BGP 对等体组信息。
- 使用 **display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **peer** [[*ipv4-address*] **verbose**]命令查看 VPNv4 的 BGP 对等体信息。
- 使用 **display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **network** 命令查看 BGP 通过 network 方式发布的 VPNv4 路由信息。
- 使用 **display bgp vpnv4** { **all** | **vpn-instance** *vpn-instance-name* } **paths** [*as-regular-expression*]命令查看 BGP VPNv4 的 AS 路径信息。
- 使用 **display bgp vpnv4 vpn-instance** *vpn-instance-name* **peer** { *group-name* | *ipv4-address* } **log-info** 命令查看 VPN 实例的 BGP 对等体日志信息。

----结束

2.17.3 检测网络连通性/可达性

主要包括 ping 命令检查发送端到目的地址之间的网络是否连通和 tracert 命令查看数据包从发送端到目的地址所经过的设备。

操作步骤

- 使用 **ping** [**ip**] [**-a** *source-ip-address* | **-c** *count* | **-d** | **-f** | **-h** *ttl-value* | **-i** *interface-type interface-number* | **-m** *time* | **-n** | **-p** *pattern* | **-q** | **-r** | **-s** *packetsize* | **-t** *timeout* | **-tos** *tos-value* | **-v** | **-vpn-instance** *vpn-instance-name*] * *host* 命令检测从发送端到目的地址之间的网络是否连通。

- 使用 **tracert** [**-a** *source-ip-address* | **-f** *first-ttl* | **-m** *max-ttl* | **-p** *port* | **-q** *nqueries* | **-vpn-instance** *vpn-instance-name* | **-w** *timeout*] * *host* 命令查看数据包从发送端到目的地址所经过的网关。
- 使用 **ping lsp** [**-a** *source-ip* | **-c** *count* | **-exp** *exp-value* | **-h** *ttl-value* | **-m** *interval* | **-r** *reply-mode* | **-s** *packet-size* | **-t** *time-out* | **-v**] * **vpn-instance** *vpn-name* **remote** *remote-address mask-length* 命令检测 L3VPN LSP 链路的连通性。

----结束

任务示例

VPN 配置完成后，可以在 PE 设备上执行带**-vpn-instance** *vpn-instance-name* 参数的 **ping** 命令检查 PE 与属于同一 VPN 的 CE 设备之间的网络是否连通。如果 ping 不通，使用带**-vpn-instance** *vpn-instance-name* 参数的 **tracert** 命令分析网络什么地方发生了故障。

```
<Huawei> ping -vpn-instance vpna 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
```

如果 PE 上有多个绑定了同一个 VPN 的接口，**ping** 或 **tracert** 对端 PE 接入的 CE 时要指定源 IP 地址，即指定参数**-a** *source-ip-address*。如果不指定源 IP 地址，PE 将选择本设备上绑定该 VPN 的接口 IP 地址最小值作为 ICMP 报文的源地址。如果 CE 没有到被选中的 IPv4 地址的路由，返回的 ICMP 报文将被 CE 丢弃。

```
<Huawei> ping -a 202.38.160.243 -c 8 10.1.1.2
  PING 10.1.1.2: 56 data bytes, press CTRL_C to break
    Reply from 10.1.1.2: bytes=56 Sequence=1 ttl=255 time=32 ms
    Reply from 10.1.1.2: bytes=56 Sequence=2 ttl=255 time=32 ms
    Reply from 10.1.1.2: bytes=56 Sequence=3 ttl=255 time=32 ms
    Reply from 10.1.1.2: bytes=56 Sequence=4 ttl=255 time=32 ms
    Reply from 10.1.1.2: bytes=56 Sequence=5 ttl=255 time=32 ms
    Reply from 10.1.1.2: bytes=56 Sequence=6 ttl=255 time=32 ms
    Reply from 10.1.1.2: bytes=56 Sequence=7 ttl=255 time=32 ms
    Reply from 10.1.1.2: bytes=56 Sequence=8 ttl=255 time=32 ms
  --- 10.1.1.2 ping statistics ---
    8 packet(s) transmitted
    8 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 32/32/32 ms
```

2.17.4 清除 VPN 实例 IPv4 地址族的 BGP 统计信息

清除 VPN 实例 IPv4 地址族的 BGP 统计信息后，以前的信息将无法恢复，务必仔细确认。

操作步骤

- 在确认需要清除指定 VPN 实例 IPv4 地址族的 BGP 对等体振荡统计信息后，请在用户视图下执行 **reset bgp vpn-instance** *vpn-instance-name* **ipv4-family** [*ipv4-address*]**flap-info** 命令。

- 在确认需要清除指定 VPN 实例 IPv4 地址族的衰减统计信息后，请在用户视图下执行 **reset bgp vpn-instance** *vpn-instance-name* **ipv4-family dampening** [*ipv4-address* [*mask* | *mask-length*]]命令。

----结束

2.17.5 复位 BGP 连接

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。注意：复位 BGP 连接将导致 VPN 业务的中断。

背景信息

复位 BGP 连接将导致 VPN 业务中断，务必仔细确认是否必须执行复位 BGP 连接的操作。

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）。

操作步骤

- 在用户视图下，执行 **refresh bgp vpn-instance** *vpn-instance-name* **ipv4-family** { **all** | *ipv4-address* | **group** *group-name* | **internal** | **external** } **import** 命令来触发入方向 VPN 实例 IPv4 地址族 BGP 连接软复位，使新的配置生效。
- 在用户视图下，执行 **refresh bgp vpn-instance** *vpn-instance-name* **ipv4-family** { **all** | *ipv4-address* | **group** *group-name* | **internal** | **external** } **export** 命令来触发出方向 VPN 实例 IPv4 地址族 BGP 连接软复位，使新的配置生效。
- 在用户视图下，执行 **refresh bgp vpnv4** { **all** | *ipv4-address* | **group** *group-name* | **internal** | **external** } **import** 命令来触发入方向 BGP 的 VPNv4 连接软复位，使新的配置生效。
- 在用户视图下，执行 **refresh bgp vpnv4** { **all** | *ipv4-address* | **group** *group-name* | **internal** | **external** } **export** 命令来触发出方向 BGP 的 VPNv4 连接软复位，使新的配置生效。
- 在用户视图下，执行 **reset bgp vpn-instance** *vpn-instance-name* **ipv4-family** { *as-number* | *ipv4-address* | **group** *group-name* | **all** | **internal** | **external** }命令来复位指定 VPN 实例 IPv4 地址族的 BGP 连接，使新的配置生效。
- 在用户视图下，执行 **reset bgp vpnv4** { *as-number* | *ipv4-address* | **group** *group-name* | **all** | **internal** | **external** }命令来复位 BGP 的 VPNv4 连接，使新的配置生效。

----结束

2.18 配置举例

介绍 VPN 各种组网的配置举例。配置示例中包括组网需求、配置注意事项、配置思路、配置过程和配置文件。

2.18.1 配置 BGP/MPLS IP VPN 示例

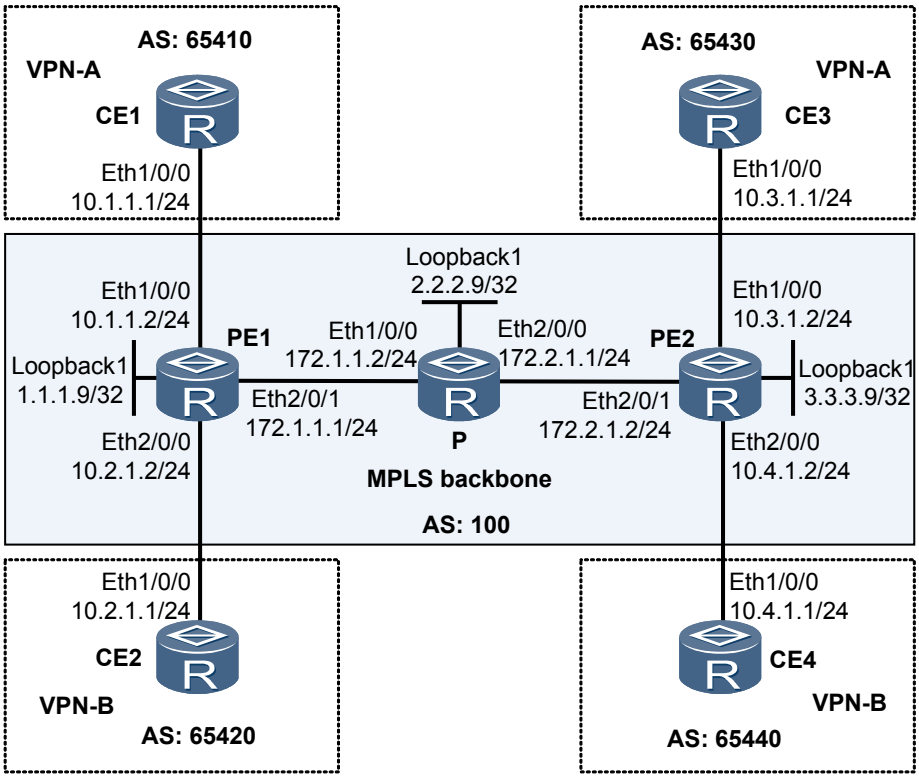
介绍基本的 BGP/MPLS IP VPN 配置过程，包括配置 MPLS LSP、VPNv4 对等体和 VPN 实例等。

组网需求

如图 2-2:

- CE1、CE3 属于 VPN-A。
- CE2、CE4 属于 VPN-B。
- VPN-A 使用的 VPN-target 属性为 111:1，VPN-B 为 222:2。
- 不同 VPN 用户之间不能互相访问。

图 2-2 BGP/MPLS IP VPN 组网图



配置思路

采用如下的思路配置 BGP/MPLS IP VPN：

1. 骨干网上配置 OSPF 实现 PE 之间的互通。
2. 配置 MPLS 基本能力和 MPLS LDP，建立 MPLS LSP。
3. PE 之间配置 MP-IBGP 交换 VPN 路由信息。
4. PE 上配置 VPN 实例，并把与 CE 相连的接口和相应的 VPN 实例绑定。
5. CE 与 PE 之间配置 EBGP 交换 VPN 路由信息。

数据准备

为完成此配置例，需准备如下的数据：

- PE 及 P 上的 MPLS LSR-ID
- VPN-A 与 VPN-B 的路由区分符 RD
- VPN-A 与 VPN-B 的收发路由属性 VPN-Target

操作步骤

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 PE1。

```
<Huawei> system-view
[Huawei] sysname PE1
[PE1] interface loopback 1
[PE1-LoopBack1] ip address 1.1.1.9 32
[PE1-LoopBack1] quit
[PE1] interface ethernet2/0/1
[PE1-Ethernet2/0/1] ip address 172.1.1.1 24
[PE1-Ethernet2/0/1] 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。

```
<Huawei> system-view
[Huawei] sysname P
[P] interface loopback 1
[P-LoopBack1] ip address 2.2.2.9 32
[P-LoopBack1] quit
[P] interface ethernet 1/0/0
[P-Ethernet1/0/0] ip address 172.1.1.2 24
[P-Ethernet1/0/0] quit
[P] interface ethernet 2/0/0
[P-Ethernet2/0/0] ip address 172.2.1.1 24
[P-Ethernet2/0/0] 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 PE2。

```
<Huawei> system-view
[Huawei] sysname PE2
[PE2] interface loopback 1
[PE2-LoopBack1] ip address 3.3.3.9 32
[PE2-LoopBack1] quit
[PE2] interface ethernet 2/0/1
[PE2-Ethernet2/0/1] ip address 172.2.1.2 24
[PE2-Ethernet2/0/1] 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，PE1 与 P、P 与 PE2 之间应能建立 OSPF 邻居关系，执行 **display ospf peer** 命令可以看到邻居状态为 Full。执行 **display ip routing-table** 命令可以看到 PE 之间学习到对方的 Loopback1路由。

以 PE1 的显示为例：

```
[PE1] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        1.1.1.9/32  Direct  0    0           D   127.0.0.1       InLoopBack0
        2.2.2.9/32  OSPF    10   2           D   172.1.1.2       Ethernet2/0/1
        3.3.3.9/32  OSPF    10   3           D   172.1.1.2       Ethernet2/0/1
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
      172.1.1.0/24  Direct  0    0           D   172.1.1.1       Ethernet2/0/1
      172.1.1.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
      172.2.1.0/24  OSPF    10   2           D   172.1.1.2       Ethernet2/0/1
[PE1] display ospf peer
          OSPF Process 1 with Router ID 1.1.1.9
                  Neighbors
 Area 0.0.0.0 interface 172.1.1.1(Ethernet2/0/1)'s neighbors
 Router ID: 2.2.2.9          Address: 172.1.1.2
   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 ]
```

步骤 2 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 PE1。

```
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
[PE1] interface ethernet 2/0/1
[PE1-Ethernet2/0/1] mpls
[PE1-Ethernet2/0/1] mpls ldp
[PE1-Ethernet2/0/1] 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 ethernet 1/0/0
[P-Ethernet1/0/0] mpls
[P-Ethernet1/0/0] mpls ldp
[P-Ethernet1/0/0] quit
[P] interface ethernet 2/0/0
[P-Ethernet2/0/0] mpls
[P-Ethernet2/0/0] mpls ldp
[P-Ethernet2/0/0] quit
```

配置 PE2。

```
[PE2] mpls lsr-id 3.3.3.9
[PE2] mpls
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface ethernet 2/0/1
```

```
[PE2-Ethernet2/0/1] mpls
[PE2-Ethernet2/0/1] mpls ldp
[PE2-Ethernet2/0/1] quit
```

上述配置完成后，PE1 与 P、P 与 PE2 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 项为“Operational”。执行 **display mpls ldp lsp** 命令，可以看到 LDP LSP 的建立情况。

以 PE1 的显示为例：

```
[PE1] display mpls ldp session
 LDP Session(s) in Public Network
 Codes: LAM(Label Advertisement Mode), SsnAge Unit(DDDD:HH:MM)
 A '*' before a session means the session is being deleted.
 ------------------------------------------------------------------------------
 PeerID            Status      LAM  SsnRole  SsnAge      KASent/Rcv
 ------------------------------------------------------------------------------
 2.2.2.9:0         Operational DU  Passive  0000:00:01  5/5
 ------------------------------------------------------------------------------
 TOTAL: 1 session(s) Found.
[PE1] display mpls ldp lsp
 LDP LSP Information
 -------------------------------------------------------------------------------
 DestAddress/Mask   In/OutLabel   UpstreamPeer   NextHop         OutInterface
 -------------------------------------------------------------------------------
 1.1.1.9/32         3/NULL        2.2.2.9        127.0.0.1       InLoop0
*1.1.1.9/32         Liberal
 2.2.2.9/32         NULL/3        -              172.1.1.2       Ethernet2/0/1
 2.2.2.9/32         1024/3        2.2.2.9        172.1.1.2       Ethernet2/0/1
 3.3.3.9/32         NULL/1025     -              172.1.1.2       Ethernet2/0/1
 3.3.3.9/32         1025/1025     2.2.2.9        172.1.1.2       Ethernet2/0/1
 -------------------------------------------------------------------------------
 TOTAL: 5 Normal LSP(s) Found.
 TOTAL: 1 Liberal LSP(s) Found.
 TOTAL: 0 Frr LSP(s) Found.
 A '*' before an LSP means the LSP is not established
 A '*' before a Label means the USCB or DSCB is stale
 A '*' before a UpstreamPeer means the session is in GR state
 A '*' before a NextHop means the LSP is FRR LSP
```

步骤 3 在 PE 之间建立 MP-IBGP 对等体关系

配置 PE1。

```
[PE1] bgp 100
[PE1-bgp] peer 3.3.3.9 as-number 100
[PE1-bgp] peer 3.3.3.9 connect-interface loopback 1
[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 PE2。

```
[PE2] bgp 100
[PE2-bgp] peer 1.1.1.9 as-number 100
[PE2-bgp] peer 1.1.1.9 connect-interface loopback 1
[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 vpnv4 all peer
BGP local router ID : 1.1.1.9
 Local AS number : 100
 Total number of peers : 1                 Peers in established state : 1
```

```
  Peer            V    AS  MsgRcvd  MsgSent    OutQ  Up/Down    State        PrefRcv
  3.3.3.9         4   100   12       18         0     00:09:38   Established  0
```

步骤 4 在 PE 设备上配置 VPN 实例，将 CE 接入 PE

配置 PE1。

```
[PE1] ip vpn-instance vpna
[PE1-vpn-instance-vpna] ipv4-family
[PE1-vpn-instance-vpna-af-ipv4] route-distinguisher 100:1
[PE1-vpn-instance-vpna-af-ipv4] vpn-target 111:1 both
[PE1-vpn-instance-vpna-af-ipv4] quit
[PE1-vpn-instance-vpna] quit
[PE1] ip vpn-instance vpnb
[PE1-vpn-instance-vpnb] ipv4-family
[PE1-vpn-instance-vpnb-af-ipv4] route-distinguisher 100:2
[PE1-vpn-instance-vpnb-af-ipv4] vpn-target 222:2 both
[PE1-vpn-instance-vpnb-af-ipv4] quit
[PE1-vpn-instance-vpnb] quit
[PE1] interface ethernet 1/0/0
[PE1-Ethernet1/0/0] ip binding vpn-instance vpna
[PE1-Ethernet1/0/0] ip address 10.1.1.2 24
[PE1-Ethernet1/0/0] quit
[PE1] interface ethernet 2/0/0
[PE1-Ethernet2/0/0] ip binding vpn-instance vpnb
[PE1-Ethernet2/0/0] ip address 10.2.1.2 24
[PE1-Ethernet2/0/0] quit
```

配置 PE2。

```
[PE2] ip vpn-instance vpna
[PE2-vpn-instance-vpna] ipv4-family
[PE2-vpn-instance-vpna-af-ipv4] route-distinguisher 200:1
[PE2-vpn-instance-vpna-af-ipv4] vpn-target 111:1 both
[PE2-vpn-instance-vpna-af-ipv4] quit
[PE2-vpn-instance-vpna] quit
[PE2] ip vpn-instance vpnb
[PE2-vpn-instance-vpnb] ipv4-family
[PE2-vpn-instance-vpnb-af-ipv4] route-distinguisher 200:2
[PE2-vpn-instance-vpnb-af-ipv4] vpn-target 222:2 both
[PE2-vpn-instance-vpnb-af-ipv4] quit
[PE2-vpn-instance-vpnb] quit
[PE2] interface ethernet 1/0/0
[PE2-Ethernet1/0/0] ip binding vpn-instance vpna
[PE2-Ethernet1/0/0] ip address 10.3.1.2 24
[PE2-Ethernet1/0/0] quit
[PE2] interface ethernet 2/0/0
[PE2-Ethernet2/0/0] ip binding vpn-instance vpnb
[PE2-Ethernet2/0/0] ip address 10.4.1.2 24
[PE2-Ethernet2/0/0] quit
```

按图 2-2 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 verbose** 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。

说明

当 PE 上有多个绑定了同一个 VPN 的接口，则使用 **ping -vpn-instance** 命令 ping 对端 PE 接入的 CE 时，要指定源 IP 地址，即要指定 **ping -vpn-instance** *vpn-instance-name* **-a** *source-ip-address dest-ip-address* 命令中的参数**-a** *source-ip-address*，否则可能 ping 不通。

以 PE1 和 CE1 为例：

```
[PE1] display ip vpn-instance verbose
  Total VPN-Instances configured : 2
 VPN-Instance Name and ID : vpna, 1
  Interfaces : Ethernet1/0/0
 Address family ipv4
```

```
  Create date : 2009/01/21 11:30:35
  Up time : 0 days, 00 hours, 05 minutes and 19 seconds
  Route Distinguisher : 100:1
  Export VPN Targets :  111:1
  Import VPN Targets :  111:1
  Label Policy : label per route
  The diffserv-mode Information is : uniform
  The ttl-mode Information is : pipe
  Log Interval : 5
 VPN-Instance Name and ID : vpnb, 2
  Interfaces : Ethernet2/0/0
 Address family ipv4
  Create date : 2009/01/21 11:31:18
  Up time : 0 days, 00 hours, 04 minutes and 36 seconds
  Route Distinguisher : 100:2
  Export VPN Targets :  222:2
  Import VPN Targets :  222:2
  Label Policy : label per route
  The diffserv-mode Information is : uniform
  The ttl-mode Information is : pipe
  Log Interval : 5
[PE1] ping -vpn-instance vpna 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
```

步骤 5 在 PE 与 CE 之间建立 EBGP 对等体关系，引入 VPN 路由

配置 CE1。

```
[CE1] bgp 65410
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] import-route direct
```

说明

另外 3 个 CE 设备（CE2 ~ CE4）配置与 CE1 设备配置类似，配置过程省略。

配置 PE1。

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpna
[PE1-bgp-vpna] peer 10.1.1.1 as-number 65410
[PE1-bgp-vpna] import-route direct
[PE1-bgp-vpna] quit
[PE1-bgp] ipv4-family vpn-instance vpnb
[PE1-bgp-vpnb] peer 10.2.1.1 as-number 65420
[PE1-bgp-vpnb] import-route direct
[PE1-bgp-vpnb] quit
```

说明

PE2 的配置与 PE1 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 PE1 与 CE1 的对等体关系为例：

```
[PE1] display bgp vpnv4 vpn-instance vpna peer
 BGP local router ID : 1.1.1.9
 Local AS number : 100
```

```
 VPN-Instance vpna, router ID 1.1.1.9:
 Total number of peers : 1           Peers in established state : 1
  Peer            V    AS  MsgRcvd  MsgSent    OutQ  Up/Down    State       PrefRcv
  10.1.1.1        4   65410  11     9          0     00:06:37   Established 1
```

步骤 6 检查配置结果

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 PE1 的显示为例：

```
[PE1] display ip routing-table vpn-instance vpna
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpna
         Destinations : 3        Routes : 3
Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface
     10.1.1.0/24    Direct 0    0         D     10.1.1.2        Ethernet1/0/0
     10.1.1.2/32    Direct 0    0         D     127.0.0.1       InLoopBack0
     10.3.1.0/24    IBGP   255  0         RD    3.3.3.9         Ethernet2/0/1
[PE1] display ip routing-table vpn-instance vpnb
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpnb
         Destinations : 3        Routes : 3
Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface
     10.2.1.0/24    Direct 0    0         D     10.2.1.2        Ethernet2/0/0
     10.2.1.2/32    Direct 0    0         D     127.0.0.1       InLoopBack0
     10.4.1.0/24    IBGP   255  0         RD    3.3.3.9         Ethernet2/0/1
```

同一 VPN 的 CE 能够相互 Ping 通，不同 VPN 的 CE 不能相互 Ping 通。

例如：CE1 能够 Ping 通 CE3（10.3.1.1/24），但不能 Ping 通 CE4（10.4.1.1/24）。

```
[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
```

----结束

配置文件

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpna
```

```
  ipv4-family
  route-distinguisher 100:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
ip vpn-instance vpnb
  ipv4-family
  route-distinguisher 100:2
  vpn-target 222:2 export-extcommunity
  vpn-target 222:2 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
#
mpls ldp
#
interface Ethernet1/0/0
 ip binding vpn-instance vpna
 ip address 10.1.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip binding vpn-instance vpnb
 ip address 10.2.1.2 255.255.255.0
#
interface Ethernet2/0/1
 ip address 172.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 3.3.3.9 as-number 100
 peer 3.3.3.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 3.3.3.9 enable
#
 ipv4-family vpnv4
  policy vpn-target
  peer 3.3.3.9 enable
 #
 ipv4-family vpn-instance vpna
  peer 10.1.1.1 as-number 65410
  import-route direct
#
 ipv4-family vpn-instance vpnb
  peer 10.2.1.1 as-number 65420
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 172.1.1.0 0.0.0.255
  network 1.1.1.9 0.0.0.0
#
return
```

- P 的配置文件

```
#
 sysname P
#
 mpls lsr-id 2.2.2.9
 mpls
#
mpls ldp
#
interface Ethernet1/0/0
 ip address 172.1.1.2 255.255.255.0
```

```
 mpls
 mpls ldp
#
interface Ethernet2/0/0
 ip address 172.2.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 172.1.1.0 0.0.0.255
  network 172.2.1.0 0.0.0.255
  network 2.2.2.9 0.0.0.0
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpna
  ipv4-family
  route-distinguisher 200:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
ip vpn-instance vpnb
  ipv4-family
  route-distinguisher 200:2
  vpn-target 222:2 export-extcommunity
  vpn-target 222:2 import-extcommunity
#
 mpls lsr-id 3.3.3.9
 mpls
#
mpls ldp
#
interface Ethernet1/0/0
 ip binding vpn-instance vpna
 ip address 10.3.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip binding vpn-instance vpnb
 ip address 10.4.1.2 255.255.255.0
#
interface Ethernet2/0/1
 ip address 172.2.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
#
bgp 100
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.9 enable
 #
 ipv4-family vpn-instance vpna
  peer 10.3.1.1 as-number 65430
```

```
  import-route direct
 #
 ipv4-family vpn-instance vpnb
  peer 10.4.1.1 as-number 65440
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 172.2.1.0 0.0.0.255
  network 3.3.3.9 0.0.0.0
#
return
```

- CE1 的配置文件

```
#
 sysname CE1
#
interface Ethernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
bgp 65410
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface Ethernet1/0/0
 ip address 10.2.1.1 255.255.255.0
#
bgp 65420
 peer 10.2.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
#
return
```

- CE3 的配置文件

```
#
 sysname CE3
#
interface Ethernet1/0/0
 ip address 10.3.1.1 255.255.255.0
#
bgp 65430
 peer 10.3.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.3.1.2 enable
#
return
```

- CE4 的配置文件

```
#
 sysname CE4
#
interface Ethernet1/0/0
```

```
 ip address 10.4.1.1 255.255.255.0
#
bgp 65440
 peer 10.4.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.4.1.2 enable
#
return
```

2.18.2 配置 BGP AS 号替换示例

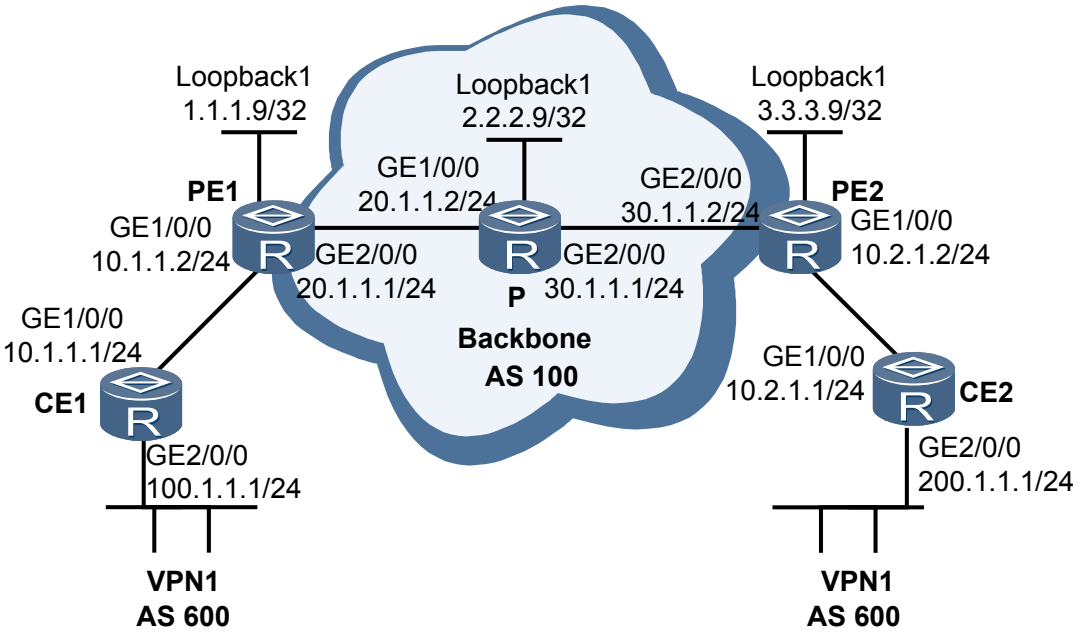
当 VPN site AS 号相同，PE 与 CE 之间采用 EBGP 连接时，需要在 VPN site 接入的 PE 设备上使能 AS 号替换功能，否则会因为 AS 相同，EBGP 协议会丢弃 AS 号相同的路由。

组网需求

如图 2-3，CE1 和 CE2 属于同一个 VPN，分别接入 PE1 和 PE2。

CE1 和 CE2 复用 AS 号 600。

图 2-3 BGP AS 号替换组网图



配置思路

本例配置主要思路是：

1. 骨干网上运行 IGP 协议实现 PE 之间及 PE 与 P 之间的互通。
2. PE 之间要建立 MPLS LDP LSP；PE 上还要创建 VPN 实例，接入 CE。
3. PE 和 CE 之间建立 EBGP 关系，将 CE 路由引入到 PE 中。

4. 在 PE 上配置 BGP 的 AS 号替换功能。

数据准备

为完成此配置例，需准备如下的数据：

- PE 及 P 上的 MPLS LSR-ID
- PE1 和 PE2 上创建的 VPN 实例
- CE1 和 CE2 用相同 AS 号（但与骨干网 AS 号不同）

操作步骤

步骤 1 配置基本 BGP/MPLS IP VPN

包括以下配置：

- 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 VPNv4 路由；
- 在 PE2 上配置 VPN1 的 VPN 实例，接入 CE2；
- 在 PE1 上配置 VPN1 的 VPN 实例，接入 CE1；
- PE1 和 CE1、PE2 和 CE2 之间配置 BGP，将 CE 的路由引入 PE。

完成上述配置后，在 CE2 上执行 **display ip routing-table** 命令，可以看到 CE2 能够学到 CE1 接入 PE1 的接口所在网段（10.1.1.0/24）的路由，但没有到达 CE1 内部 VPN（100.1.1.0/24）的路由。CE1 上也存在同样的现象。

```
[CE2] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9

Destination/Mask  Proto  Pre  Cost      Flags NextHop         Interface

     10.1.1.0/24  EBGP   255  0           D   10.2.1.2        GigabitEthernet1/0/0
     10.1.1.1/32  EBGP   255  0           D   10.2.1.2        GigabitEthernet1/0/0
     10.2.1.0/24  Direct 0    0           D   10.2.1.1        GigabitEthernet1/0/0
     10.2.1.2/32  Direct 0    0           D   10.2.1.2        GigabitEthernet1/0/0
     10.2.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
     127.0.0.0/8  Direct 0    0           D   127.0.0.1       InLoopBack0
    127.0.0.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
    200.1.1.0/24  Direct 0    0           D   200.1.1.1       GigabitEthernet2/0/0
    200.1.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
```

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 PE2 的显示为例：

```
[PE2] display ip routing-table vpn-instance vpn1
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpn1
         Destinations : 8        Routes : 8

Destination/Mask  Proto  Pre  Cost      Flags NextHop         Interface

     10.1.1.0/24  EBGP    255  0          RD   1.1.1.9         GigabitEthernet2/0/0
     10.1.1.1/32  EBGP    255  0          RD   1.1.1.9         GigabitEthernet2/0/0
     10.1.1.2/32  EBGP    255  0          RD   1.1.1.9         GigabitEthernet2/0/0
     10.2.1.0/24  Direct  0    0          D    10.2.1.2        GigabitEthernet1/0/0
     10.2.1.1/32  Direct  0    0          D    10.2.1.1        GigabitEthernet1/0/0
     10.2.1.2/32  Direct  0    0          D    127.0.0.1       InLoopBack0
```

```
    100.1.1.0/24  EBGP    255  0         RD      1.1.1.9         GigabitEthernet2/0/0
    200.1.1.0/24  EBGP    255  0         D       10.2.1.1        GigabitEthernet1/0/0
```

在 CE2 上执行 **display bgp routing-table peer received-routes** 命令，可以看到 CE2 没有接收 100.1.1.0/24 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, > - 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?
```

步骤 2 配置 BGP 的 AS 号替换功能

在 PE 上配置 BGP 的 AS 号替换功能。

以 PE2 上的配置为例。

```
[PE2] bgp 100
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] peer 10.2.1.1 substitute-as
```

再次查看 CE2 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: 6
 BGP Local router ID is 10.2.1.1
 Status codes: * - valid, > - 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?
 *>   10.1.1.2/32      10.2.1.2                           0       100 100?
 *    10.2.1.0/24      10.2.1.2       0                   0       100?
 *    10.2.1.1/32      10.2.1.2       0                   0       100?
 *>   100.1.1.0/24     10.2.1.2                           0       100 100?
[CE2] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10        Routes : 10
Destination/Mask  Proto  Pre  Cost       Flags  NextHop         Interface
     10.1.1.0/24  EBGP    255  0          D      10.2.1.2        GigabitEthernet1/0/0
     10.1.1.1/32  EBGP    255  0          D      10.2.1.2        GigabitEthernet1/0/0
     10.2.1.0/24  Direct 0    0          D      10.2.1.1        GigabitEthernet1/0/0
     10.2.1.1/32  Direct 0    0          D      127.0.0.1       InLoopBack0
     10.2.1.2/32  Direct 0    0          D      10.2.1.2        GigabitEthernet1/0/0
    100.1.1.1/24 EBGP    255  0          D      10.2.1.2        GigabitEthernet1/0/0
    127.0.0.0/8   Direct 0    0          D      127.0.0.1       InLoopBack0
    127.0.0.1/32  Direct 0    0          D      127.0.0.1       InLoopBack0
    200.1.1.0/24  Direct 0    0          D      127.0.0.1       GigabitEthernet2/0/0
    200.1.1.1/32  Direct 0    0          D      127.0.0.1       InLoopBack0
```

在 PE1 上也配置 BGP 的 AS 号替换功能后，CE1 和 CE2 的 GigabitEthernet 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
```

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 100.1.1.1 255.255.255.0
#
bgp 600
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
#
return
```

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip binding vpn-instance vpn1
 ip address 10.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 20.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 3.3.3.9 as-number 100
 peer 3.3.3.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 3.3.3.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
```

```
  peer 3.3.3.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.1.1.1 as-number 600
  peer 10.1.1.1 substitute-as
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 1.1.1.9 0.0.0.0
  network 20.1.1.0 0.0.0.255
#
return
```

- P 的配置文件

```
#
 sysname P
#
 mpls lsr-id 2.2.2.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 20.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip address 30.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 2.2.2.9 0.0.0.0
  network 20.1.1.0 0.0.0.255
  network 30.1.1.0 0.0.0.255
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 3.3.3.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip binding vpn-instance vpn1
 ip address 10.2.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 30.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
```

```
#
bgp 100
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.2.1.1 as-number 600
  peer 10.2.1.1 substitute-as
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 3.3.3.9 0.0.0.0
  network 30.1.1.0 0.0.0.255
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface GigabitEthernet1/0/0
 ip address 10.2.1.1 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 200.1.1.1 255.255.255.0
#
bgp 600
 peer 10.2.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
#
return
```

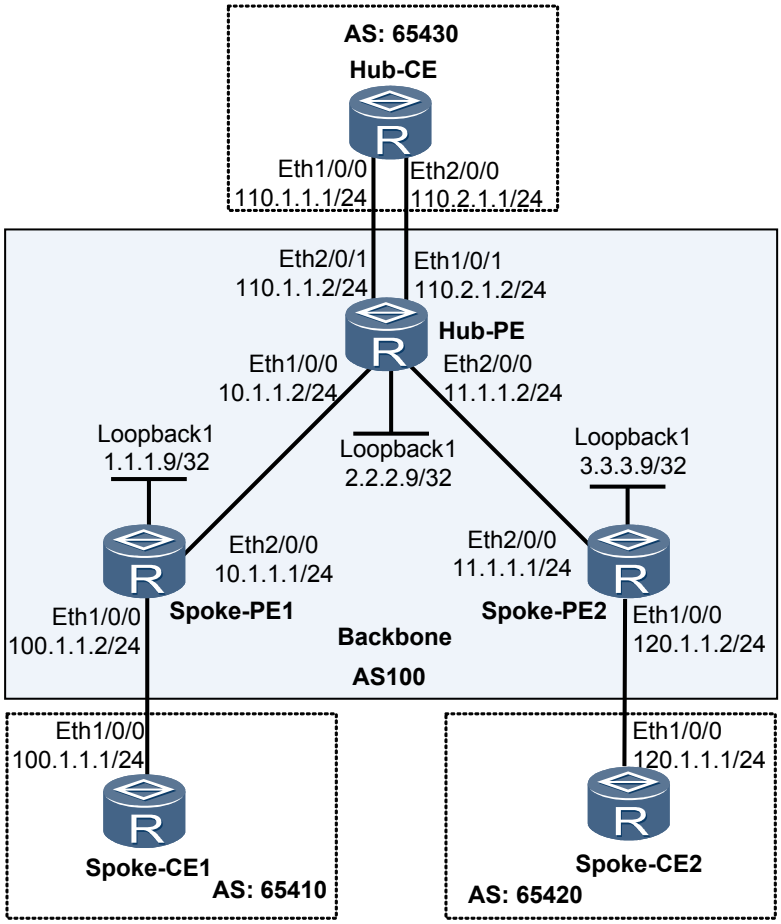
2.18.3 配置 Hub and Spoke 示例

Hub and Spoke 是在 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。

组网需求

如图 2-4，Spoke-CE 之间的通信通过中心站点 Hub-CE 控制，即 Spoke-CE 之间的流量经过 Hub-CE 转发，而不是只经过 Hub-PE 转发。

图 2-4 Hub and Spoke 组网图



配置思路

本例配置主要思路是：

1. Hub-PE 与 Spoke-PE 建立 MP-IBGP 对等体关系。（Spoke-PE 之间不建立 MP-IBGP 对等体关系，不交换 VPN 路由信息）
2. 在 Spoke-PE 上创建一个 VPN 实例，其 Import Target 属性与 Export Target 属性设为不同。
3. 在 Hub-PE 上创建两个 VPN 实例（vpn_in, vpn_out）。其中，vpn_in 接收的 VPN-target 团体属性为两个 Spoke-PE 发布的 VPN-target 团体属性值；vpn_out 发布的 VPN-target 团体属性值与接收的 VPN-target 团体属性不同，且为两个 Spoke-PE 接收的 VPN-target 团体属性值。
4. CE 和 PE 之间使用 EBGP。
5. Hub-PE 上配置允许接收 AS 重复 1 次的路由。

数据准备

为完成此配置例，需准备如下的数据：

- PE 上的 MPLS LSR ID

- Hub-PE 和 Spoke-PE 的 VPN 实例名、路由标志（Route-Distinguisher）及 VPN-Target

操作步骤

步骤 1 在骨干网上配置 IGP 协议，实现骨干网 Hub-PE 和 Spoke-PE 的互通

本例中采用 OSPF，具体配置过程略。

配置完成后，Hub-PE 和 Spoke-PE 之间应能建立 OSPF 邻居关系，执行 **display ospf peer** 命令可以看到邻居状态为 Full。执行 **display ip routing-table** 命令可以看到 Hub-PE 和 Spoke-PE 之间学习到对方的 Loopback 路由。

步骤 2 在骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

具体配置过程略。

配置完成后，Hub-PE 和 Spoke-PE 之间应该建立起 LDP 对等体关系，在各路由器上执行 **display mpls ldp session** 命令可以看到显示结果中 Session State 项为“Operational”。

步骤 3 在各 PE 设备上配置 VPN 实例，将 CE 接入 PE

说明

Hub-PE 上其中一个 VPN 的 Import-Target 列表必须包含所有 Spoke-PE 的 Export-Target 属性。

Hub-PE 上另一个 VPN 的 Export-Target 列表必须包含所有 Spoke-PE 的 Import-Targetsh 属性。

配置 Spoke-PE1。

```
<Spoke-PE1> system-view
[Spoke-PE1] ip vpn-instance vpna
[Spoke-PE1-vpn-instance-vpna] ipv4-family
[Spoke-PE1-vpn-instance-vpna-af-ipv4] route-distinguisher 100:1
[Spoke-PE1-vpn-instance-vpna-af-ipv4] vpn-target 100:1 export-extcommunity
[Spoke-PE1-vpn-instance-vpna-af-ipv4] vpn-target 200:1 import-extcommunity
[Spoke-PE1-vpn-instance-vpna-af-ipv4] quit
[Spoke-PE1-vpn-instance-vpna] quit
[Spoke-PE1] interface ethernet 1/0/0
[Spoke-PE1-Ethernet1/0/0] ip binding vpn-instance vpna
[Spoke-PE1-Ethernet1/0/0] ip address 100.1.1.2 24
[Spoke-PE1-Ethernet1/0/0] quit
```

配置 Spoke-PE2。

```
<Spoke-PE2> system-view
[Spoke-PE2] ip vpn-instance vpna
[Spoke-PE2-vpn-instance-vpna] ipv4-family
[Spoke-PE2-vpn-instance-vpna-af-ipv4] route-distinguisher 100:3
[Spoke-PE2-vpn-instance-vpna-af-ipv4] vpn-target 100:1 export-extcommunity
[Spoke-PE2-vpn-instance-vpna-af-ipv4] vpn-target 200:1 import-extcommunity
[Spoke-PE2-vpn-instance-vpna-af-ipv4] quit
[Spoke-PE2-vpn-instance-vpna] quit
[Spoke-PE2] interface ethernet 1/0/0
[Spoke-PE2-Ethernet1/0/0] ip binding vpn-instance vpna
[Spoke-PE2-Ethernet1/0/0] ip address 120.1.1.2 24
[Spoke-PE2-Ethernet1/0/0] quit
```

配置 Hub-PE。

```
<Hub-PE> system-view
[Hub-PE] ip vpn-instance vpn_in
[Hub-PE-vpn-instance-vpn_in] ipv4-family
[Hub-PE-vpn-instance-vpn_in-af-ipv4] route-distinguisher 100:21
[Hub-PE-vpn-instance-vpn_in-af-ipv4] vpn-target 100:1 import-extcommunity
[Hub-PE-vpn-instance-vpn_in-af-ipv4] quit
[Hub-PE-vpn-instance-vpn_in] quit
```

```
[Hub-PE] ip vpn-instance vpn_out
[Hub-PE-vpn-instance-vpn_out] ipv4-family
[Hub-PE-vpn-instance-vpn_out-af-ipv4] route-distinguisher 100:22
[Hub-PE-vpn-instance-vpn_out-af-ipv4] vpn-target 200:1 export-extcommunity
[Hub-PE-vpn-instance-vpn_out-af-ipv4] quit
[Hub-PE-vpn-instance-vpn_out] quit
[Hub-PE] interface ethernet 2/0/1
[Hub-PE-Ethernet2/0/1] ip binding vpn-instance vpn_in
[Hub-PE-Ethernet2/0/1] ip address 110.1.1.2 24
[Hub-PE-Ethernet2/0/1] quit
[Hub-PE] interface ethernet 1/0/1
[Hub-PE-Ethernet1/0/1] ip binding vpn-instance vpn_out
[Hub-PE-Ethernet1/0/1] ip address 110.2.1.2 24
[Hub-PE-Ethernet1/0/1] quit
```

按图 2-4 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 verbose** 命令可以看到 VPN 实例的配置情况。各 PE 能用 **ping -vpn-instance** *vpn-name ip-address* 命令 ping 通自己接入的 CE。

说明

当 PE 上有多个绑定了同一个 VPN 的接口，则使用 **ping -vpn-instance** 命令 ping 对端 PE 接入的 CE 时，要指定源 IP 地址，即要指定 **ping -vpn-instance** *vpn-instance-name* **-a** *source-ip-address dest-ip-address* 命令中的参数**-a** *source-ip-address*，否则可能 ping 不通。

步骤 4 在 PE 与 CE 之间建立 EBGP 对等体关系，引入 VPN 路由

说明

Hub-PE 上需要配置允许 AS 号重复一次，以接收 Hub-CE 发布的路由。

Spoke-PE 上不需要配置允许 AS 号重复一次，因为路由器接收 IBGP 对等体发布的路由时并不检查其中的 AS-PATH 属性。

配置 Spoke-CE1。

```
[Spoke-CE1] bgp 65410
[Spoke-CE1-bgp] peer 100.1.1.2 as-number 100
[Spoke-CE1-bgp] import-route direct
[Spoke-CE1-bgp] quit
```

配置 Spoke-PE1。

```
[Spoke-PE1] bgp 100
[Spoke-PE1-bgp] ipv4-family vpn-instance vpna
[Spoke-PE1-bgp-vpna] peer 100.1.1.1 as-number 65410
[Spoke-PE1-bgp-vpna] quit
[Spoke-PE1-bgp] quit
```

配置 Spoke-CE2。

```
[Spoke-CE2] bgp 65420
[Spoke-CE2-bgp] peer 120.1.1.2 as-number 100
[Spoke-CE2-bgp] import-route direct
[Spoke-CE2-bgp] quit
```

配置 Spoke-PE2。

```
[Spoke-PE2] bgp 100
[Spoke-PE2-bgp] ipv4-family vpn-instance vpna
[Spoke-PE2-bgp-vpna] peer 120.1.1.1 as-number 65420
[Spoke-PE2-bgp-vpna] quit
[Spoke-PE2-bgp] quit
```

配置 Hub-CE。

```
[Hub-CE] bgp 65430
```

```
[Hub-CE-bgp] peer 110.1.1.2 as-number 100
[Hub-CE-bgp] peer 110.2.1.2 as-number 100
[Hub-CE-bgp] import-route direct
[Hub-CE-bgp] quit
```

配置 Hub-PE。

```
[Hub-PE] bgp 100
[Hub-PE-bgp] ipv4-family vpn-instance vpn_in
[Hub-PE-bgp-vpn_in] peer 110.1.1.1 as-number 65430
[Hub-PE-bgp-vpn_in] quit
[Hub-PE-bgp] ipv4-family vpn-instance vpn_out
[Hub-PE-bgp-vpn_out] peer 110.2.1.1 as-number 65430
[Hub-PE-bgp-vpn_out] peer 110.2.1.1 allow-as-loop 1
[Hub-PE-bgp-vpn_out] quit
[Hub-PE-bgp] quit
```

配置完成后，在各 PE 设备上执行 **display bgp vpnv4 all 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。

步骤 5 在 PE 之间建立 MP-IBGP 对等体关系

配置 Spoke-PE1。

```
[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 1
[Spoke-PE1-bgp] ipv4-family vpnv4
[Spoke-PE1-bgp-af-vpnv4] peer 2.2.2.9 enable
[Spoke-PE1-bgp-af-vpnv4] quit
```

配置 Spoke-PE2。

```
[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 1
[Spoke-PE2-bgp] ipv4-family vpnv4
[Spoke-PE2-bgp-af-vpnv4] peer 2.2.2.9 enable
[Spoke-PE2-bgp-af-vpnv4] 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 1
[Hub-PE-bgp] peer 3.3.3.9 as-number 100
[Hub-PE-bgp] peer 3.3.3.9 connect-interface loopback 1
[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
```

配置完成后，在各 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。

步骤 6 检查配置结果

完成上述配置后，Spoke-CE 之间可以相互 Ping 通，使用 Tracert 可以看到 Spoke-CE 之间的流量经过 Hub-CE 转发，也可以通过 Ping 结果中的 TTL 值推算 Spoke-CE 之间经过的转发设备数目。

以 Spoke-CE1 的显示为例：

```
[Spoke-CE1] 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=250 time=80 ms
    Reply from 120.1.1.1: bytes=56 Sequence=2 ttl=250 time=129 ms
    Reply from 120.1.1.1: bytes=56 Sequence=3 ttl=250 time=132 ms
```

```
    Reply from 120.1.1.1: bytes=56 Sequence=4 ttl=250 time=92 ms
    Reply from 120.1.1.1: bytes=56 Sequence=5 ttl=250 time=126 ms
  --- 120.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 80/111/132 ms
[Spoke-CE1] tracert 120.1.1.1
 traceroute to  120.1.1.1(120.1.1.1), max hops: 30 ,packet length: 40
1 100.1.1.2 8 ms  2 ms  2 ms
 2 110.2.1.2  3 ms  2 ms  2 ms
 3 110.2.1.1  3 ms  2 ms  2 ms
 4 110.1.1.2  3 ms  2 ms  2 ms
 5 120.1.1.2  6 ms  6 ms  6 ms
 6 120.1.1.1  6 ms  6 ms  6 ms
```

在 Spoke-CE 上执行 **display bgp routing-table** 命令，可以看到去往对端 Spoke-CE 的 BGP 路由的 AS 路径中存在重复的 AS 号。

以 Spoke-CE1 的显示为例：

```
[Spoke-CE1] display bgp routing-table
 Total Number of Routes: 6
 BGP Local router ID is 100.1.1.1
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
     Network         NextHop       MED    LocPrf    PrefVal Path/Ogn
 *>  100.1.1.0/24    0.0.0.0         0                 0      ?
 *                   100.1.1.2       0                 0      100?
 *>  100.1.1.1/32    0.0.0.0         0                 0      ?
 *>  110.1.1.0/24    100.1.1.2                         0      100 65430?
 *>  110.2.1.0/24    100.1.1.2                         0      100?
 *>  120.1.1.0/24    100.1.1.2                         0      100 65430 100?
```

----结束

配置文件

- Spoke-CE1 的配置文件

```
#
 sysname Spoke-CE1
#
interface Ethernet1/0/0
 ip address 100.1.1.1 255.255.255.0
#
bgp 65410
 peer 100.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 100.1.1.2 enable
#
return
```

- Spoke-PE1 的配置文件

```
#
 sysname Spoke-PE1
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 100:1
  vpn-target 100:1 export-extcommunity
  vpn-target 200:1 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
```

```
#
mpls ldp
#
interface Ethernet1/0/0
 ip binding vpn-instance vpna
 ip address 100.1.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 10.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 2.2.2.9 as-number 100
 peer 2.2.2.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 2.2.2.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 2.2.2.9 enable
#
 ipv4-family vpn-instance vpna
  peer 100.1.1.1 as-number 65410
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 10.1.1.0 0.0.0.255
  network 1.1.1.9 0.0.0.0
#
return
```

- Spoke-PE2 的配置文件

```
#
 sysname Spoke-PE2
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 100:3
  vpn-target 100:1 export-extcommunity
  vpn-target 200:1 import-extcommunity
#
 mpls lsr-id 3.3.3.9
 mpls
#
mpls ldp
#
interface Ethernet1/0/0
 ip binding vpn-instance vpna
 ip address 120.1.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip address 11.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
#
bgp 100
 peer 2.2.2.9 as-number 100
 peer 2.2.2.9 connect-interface LoopBack1
 #
 ipv4-family unicast
```

```
  undo synchronization
  peer 2.2.2.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 2.2.2.9 enable
#
 ipv4-family vpn-instance vpna
  peer 120.1.1.1 as-number 65420
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 3.3.3.9 0.0.0.0
  network 11.1.1.0 0.0.0.255
#
return
```

- Spoke-CE2 的配置文件

```
#
 sysname Spoke-CE2
#
interface Ethernet1/0/0
 ip address 120.1.1.1 255.255.255.0
#
bgp 65420
 peer 120.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 120.1.1.2 enable
#
return
```

- Hub-CE 的配置文件

```
#
 sysname Hub-CE
#
interface Ethernet1/0/0
 ip address 110.1.1.1 255.255.255.0
#
interface Ethernet2/0/0
 ip address 110.2.1.1 255.255.255.0
#
bgp 65430
 peer 110.1.1.2 as-number 100
 peer 110.2.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 110.2.1.2 enable
  peer 110.1.1.2 enable
#
return
```

- Hub-PE 的配置文件

```
#
 sysname Hub-PE
#
ip vpn-instance vpn_in
 ipv4-family
  route-distinguisher 100:21
  vpn-target 100:1 import-extcommunity
#
ip vpn-instance vpn_out
 ipv4-family
  route-distinguisher 100:22
```

```
  vpn-target 200:1 export-extcommunity
#
 mpls lsr-id 2.2.2.9
 mpls
#
mpls ldp
#
interface Ethernet1/0/0
 ip address 10.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface Ethernet2/0/0
 ip address 11.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface Ethernet2/0/1
 ip binding vpn-instance vpn_in
 ip address 110.1.1.2 255.255.255.0
#
interface Ethernet1/0/1
 ip binding vpn-instance vpn_out
 ip address 110.2.1.2 255.255.255.0
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
bgp 100
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 peer 3.3.3.9 as-number 100
 peer 3.3.3.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
  peer 3.3.3.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.9 enable
  peer 3.3.3.9 enable
#
 ipv4-family vpn-instance vpn_in
  peer 110.1.1.1 as-number 65430
  import-route direct
#
 ipv4-family vpn-instance vpn_out
  peer 110.2.1.1 as-number 65430
  peer 110.2.1.1 allow-as-loop
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 2.2.2.9 0.0.0.0
  network 10.1.1.0 0.0.0.255
  network 11.1.1.0 0.0.0.255
#
return
```

2.18.4 配置 OptionA 方式跨域 VPN 示例

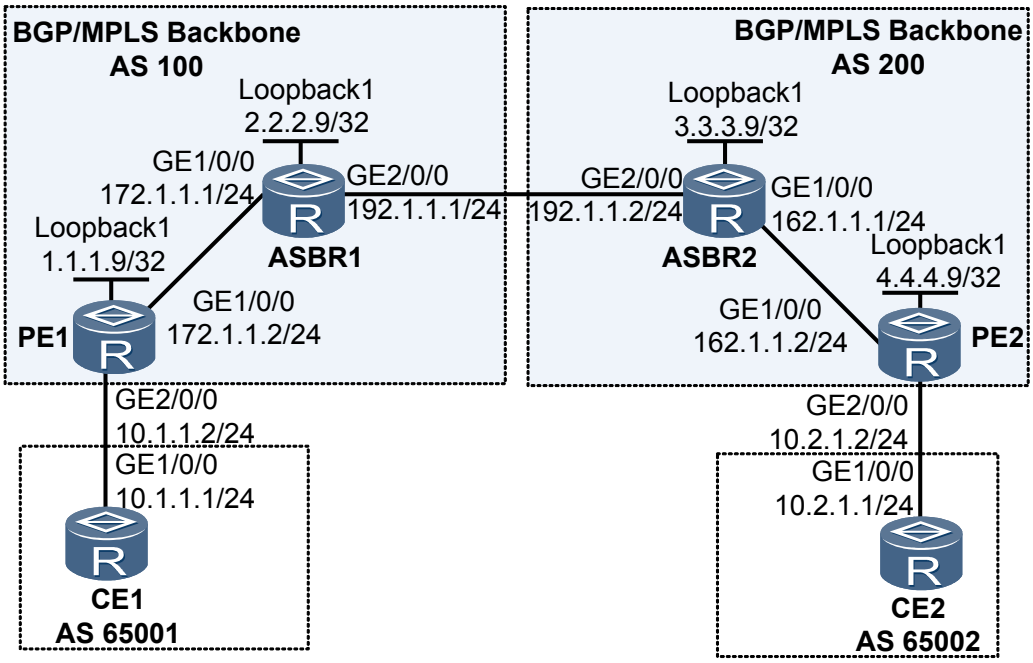
通过在 ASBR 上配置 VPN 实例，实现 VRF-to-VRF 方式管理 VPN 路由的 OptionA 方案。

组网需求

如图 2-5，CE1 和 CE2 属于同一个 VPN。CE1 通过 AS100 的 PE1 接入，CE2 通过 AS200 的 PE2 接入。

采用 OptionA 方式实现跨域的 BGP/MPLS IP VPN，即采用 VRF-to-VRF 方式管理 VPN 路由。

图 2-5 跨域 VPN 组网图



配置思路

本例配置主要思路是：

1. PE 与 CE 之间建立 EBGP 对等体关系；PE 与 ASBR 之间建立 MP-IBGP 对等体关系。
2. 在两个 ASBR 上创建 VPN 实例，并将此实例绑定到连接另一 ASBR 的接口，并在 ASBR 之间建立 EBGP 对等体关系。

数据准备

为完成此配置例，需准备如下的数据：

- PE 及 ASBR 上的 MPLS LSR-ID
- PE 及 ASBR 的 VPN 实例名、路由标志 RD 和 VPN-Target

操作步骤

步骤 1 在 AS100 和 AS200 的 MPLS 骨干网上分别配置 IGP 协议，实现各自骨干网 ASBR 和 PE 之间的互通。

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，同一 AS 的 ASBR 与 PE 之间应能建立 OSPF 邻居关系，执行 **display ospf peer** 命令可以看到邻居状态为 Full。

同一 AS 的 ASBR 和 PE 能学习到对方的 Loopback 地址，并能够互相 ping 通。

步骤 2 在 AS100 和 AS200 的 MPLS 骨干网上分别配置 MPLS 基本能力和 MPLS LDP，建立 MPLS LDP LSP。

配置 PE1 的 MPLS 基本能力，并在与 ASBR1 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 gigabitethernet1/0/0
[PE1-GigabitEthernet1/0/0] mpls
[PE1-GigabitEthernet1/0/0] mpls ldp
[PE1-GigabitEthernet1/0/0] quit
```

配置 ASBR1 的 MPLS 基本能力，并在与 PE1 相连的接口上使能 LDP。

```
<ASBR1> system-view
[ASBR1] mpls lsr-id 2.2.2.9
[ASBR1] mpls
[ASBR1-mpls] quit
[ASBR1] mpls ldp
[ASBR1-mpls-ldp] quit
[ASBR1] interface gigabitethernet1/0/0
[ASBR1-GigabitEthernet1/0/0] mpls
[ASBR1-GigabitEthernet1/0/0] mpls ldp
[ASBR1-GigabitEthernet1/0/0] quit
```

配置 ASBR2 的 MPLS 基本能力，并在与 PE2 相连的接口上使能 LDP。

```
<ASBR2> system-view
[ASBR2] mpls lsr-id 3.3.3.9
[ASBR2] mpls
[ASBR2-mpls] quit
[ASBR2] mpls ldp
[ASBR2-mpls-ldp] quit
[ASBR2] interface gigabitethernet1/0/0
[ASBR2-GigabitEthernet1/0/0] mpls
[ASBR2-GigabitEthernet1/0/0] mpls ldp
[ASBR2-GigabitEthernet1/0/0] quit
```

配置 PE2 的 MPLS 基本能力，并在与 ASBR2 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 gigabitethernet1/0/0
[PE2-GigabitEthernet1/0/0] mpls
[PE2-GigabitEthernet1/0/0] mpls ldp
[PE2-GigabitEthernet1/0/0] quit
```

上述配置完成后，同一 AS 的 PE 和 ASBR 之间应该建立起 LDP 对等体，在各 PE 或者 ASBR 上执行 **display mpls ldp session** 命令可以看到显示结果中 Session State 项为“Operational”。

以 PE1 为例：

```
[PE1] display mpls ldp session
 LDP Session(s) in Public Network
 Codes: LAM(Label Advertisement Mode), SsnAge Unit(DDDD:HH:MM)
 A '*' before a session means the session is being deleted.
------------------------------------------------------------------------------
 PeerID             Status      LAM  SsnRole  SsnAge      KASent/Rcv
------------------------------------------------------------------------------
2.2.2.9:0          Operational  DU   Passive  0000:00:02  9/9
------------------------------------------------------------------------------
 TOTAL: 1 session(s) Found.
```

步骤 3 为 AS100 和 AS200 分别配置基本 BGP/MPLS IP VPN。

说明

同一 AS 内的 ASBR 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 CE1。

```
<CE1> system-view
[CE1] interface gigabitethernet 1/0/0
[CE1-GigabitEthernet1/0/0] ip address 10.1.1.1 24
[CE1-GigabitEthernet1/0/0] quit
[CE1] bgp 65001
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] import-route direct
[CE1-bgp] quit
```

配置 PE1 与 CE1 建立 EBGP 对等体关系。

```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] ipv4-family
[PE1-vpn-instance-vpn1-af-ipv4] route-distinguisher 100:1
[PE1-vpn-instance-vpn1-af-ipv4] vpn-target 1:1 both
[PE1-vpn-instance-vpn1-af-ipv4] quit
[PE1-vpn-instance-vpn1] quit
[PE1] interface gigabitethernet 2/0/0
[PE1-GigabitEthernet2/0/0] ip binding vpn-instance vpn1
[PE1-GigabitEthernet2/0/0] ip address 10.1.1.2 24
[PE1-GigabitEthernet2/0/0] quit
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] peer 10.1.1.1 as-number 65001
[PE1-bgp-vpn1] quit
[PE1-bgp] quit
```

配置 PE1 与 ASBR1 建立 MP-IBGP 对等体关系。

```
[PE1] bgp 100
[PE1-bgp] peer 2.2.2.9 as-number 100
[PE1-bgp] peer 2.2.2.9 connect-interface loopback 1
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 2.2.2.9 enable
[PE1-bgp-af-vpnv4] quit
```

配置 ASBR1 与 PE1 建立 MP-IBGP 对等体关系。

```
[ASBR1] bgp 100
[ASBR1-bgp] peer 1.1.1.9 as-number 100
[ASBR1-bgp] peer 1.1.1.9 connect-interface loopback 1
[ASBR1-bgp] ipv4-family vpnv4
[ASBR1-bgp-af-vpnv4] peer 1.1.1.9 enable
```

```
[ASBR1-bgp-af-vpnv4] quit
[ASBR1-bgp] quit
```

说明

CE2、PE2、ASBR2 上的配置分别与 CE1、PE1、ASBR1 类似，此处不再详述。

配置完成后，在 PE 设备上执行 **display bgp vpnv4 vpn-instance** *vpn-instancename* **peer** 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。执行 **display bgp vpnv4 all peer** 命令，可以看到 PE 与 CE 之间、PE 与 ASBR 之间的 BGP 对等体关系已建立，并达到 Established 状态。

以 PE1 的显示为例：

```
[PE1] display bgp vpnv4 vpn-instance vpn1 peer
 BGP local router ID : 1.1.1.9
 Local AS number : 100
 VPN-Instance vpn1, router ID 1.1.1.9:
 Total number of peers : 1                 Peers in established state : 1
  Peer      V  AS  MsgRcvd  MsgSent  OutQ  Up/Down     State       PrefRcv
  10.1.1.1  4 65001       10         10     0 00:07:10 Established       2
[PE1] display bgp vpnv4 all peer
BGP local router ID : 1.1.1.9
 Local AS number : 100
 Total number of peers : 2                  Peers in established state : 2
  Peer            V    AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv
  2.2.2.9         4   100        3        7     0 00:01:36 Established       0
  Peer of IPv4-family for vpn instance :

 VPN-Instance vpn1, router ID 1.1.1.9:
  10.1.1.1        4 65001       13       13     0 00:04:00 Established       2
```

步骤 4 配置 VRF-to-VRF 方式的跨域 VPN

配置 ASBR1：创建 VPN 实例，并将此实例绑定到连接 ASBR2 的接口（ASBR1 认为 ASBR2 是自己的 CE）。

```
[ASBR1] ip vpn-instance vpn1
[ASBR1-vpn-instance-vpn1] ipv4-family
[ASBR1-vpn-instance-vpn1-af-ipv4] route-distinguisher 100:2
[ASBR1-vpn-instance-vpn1-af-ipv4] vpn-target 1:1 both
[ASBR1-vpn-instance-vpn1-af-ipv4] quit
[ASBR1-vpn-instance-vpn1] quit
[ASBR1] interface gigabitethernet 2/0/0
[ASBR1-GigabitEthernet2/0/0] ip binding vpn-instance vpn1
[ASBR1-GigabitEthernet2/0/0] ip address 192.1.1.1 24
[ASBR1-GigabitEthernet2/0/0] quit
```

配置 ASBR2：创建 VPN 实例，并将此实例绑定到连接 ASBR1 的接口（ASBR2 认为 ASBR1 是自己的 CE）。

```
[ASBR2] ip vpn-instance vpn1
[ASBR2-vpn-instance-vpn1] ipv4-family
[ASBR2-vpn-instance-vpn1-af-ipv4] route-distinguisher 200:2
[ASBR2-vpn-instance-vpn1-af-ipv4] vpn-target 2:2 both
[ASBR2-vpn-instance-vpn1-af-ipv4] quit
[ASBR2-vpn-instance-vpn1] quit
[ASBR2] interface gigabitethernet 2/0/0
[ASBR2-GigabitEthernet2/0/0] ip binding vpn-instance vpn1
[ASBR2-GigabitEthernet2/0/0] ip address 192.1.1.2 24
[ASBR2-GigabitEthernet2/0/0] quit
```

配置 ASBR1 与 ASBR2 建立 EBGP 对等体关系。

```
[ASBR1] bgp 100
[ASBR1-bgp] ipv4-family vpn-instance vpn1
[ASBR1-bgp-vpn1] peer 192.1.1.2 as-number 200
[ASBR1-bgp-vpn1] import-route direct
```

```
[ASBR1-bgp-vpn1] quit
[ASBR1-bgp] quit
```

配置 ASBR2 与 ASBR1 建立 EBGP 对等体关系。

```
[ASBR2] bgp 200
[ASBR2-bgp] ipv4-family vpn-instance vpn1
[ASBR2-bgp-vpn1] peer 192.1.1.1 as-number 100
[ASBR2-bgp-vpn1] import-route direct
[ASBR2-bgp-vpn1] quit
[ASBR2-bgp] quit
```

配置完成后，在 ASBR 上执行 **display bgp vpnv4 vpn-instance peer** 命令，可以看到 ASBR 间的 BGP 对等体关系已建立，并达到 Established 状态。

步骤 5 检查配置结果

上述配置完成后，CE 之间能学习到对方的接口路由，CE1 和 CE2 能够相互 ping 通。

以 CE1 的显示为例：

```
[CE1] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7
Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface
       10.1.1.0/24  Direct 0    0             D  10.1.1.1        GigabitEthernet1/0/0
       10.1.1.1/32  Direct 0    0             D  127.0.0.1       InLoopBack0
     10.2.1.0/24  EBGP    255  0             D  10.1.1.2        GigabitEthernet1/0/0
      127.0.0.0/8   Direct 0    0             D  127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct 0    0             D  127.0.0.1       InLoopBack0
      192.1.1.0/24  EBGP    255  0             D  10.1.1.2        GigabitEthernet1/0/0
      192.1.1.2/32  EBGP    255  0             D  10.1.1.2        GigabitEthernet1/0/0
[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=251 time=119 ms
    Reply from 10.2.1.1: bytes=56 Sequence=2 ttl=251 time=141 ms
    Reply from 10.2.1.1: bytes=56 Sequence=3 ttl=251 time=136 ms
    Reply from 10.2.1.1: bytes=56 Sequence=4 ttl=251 time=113 ms
    Reply from 10.2.1.1: bytes=56 Sequence=5 ttl=251 time=78 ms
  --- 10.2.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 78/117/141 ms
```

在 ASBR 上执行 **display ip routing-table vpn-instance** 命令，可以看到 ASBR 上为 VPN 维护的路由表。

```
[ASBR1] display ip routing-table vpn-instance vpn1
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpn1
         Destinations : 7        Routes : 7
Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface
       10.1.1.0/24  IBGP    255  0             RD  1.1.1.9         GigabitEthernet1/0/0
       10.1.1.1/32  IBGP    255  0             RD  1.1.1.9         GigabitEthernet1/0/0
       10.2.1.0/24  IBGP    255  0              D  192.1.1.2       GigabitEthernet2/0/0
       10.2.1.1/32  IBGP    255  0              D  192.1.1.2       GigabitEthernet2/0/0
      192.1.1.0/24  Direct 0    0             D  192.1.1.1       GigabitEthernet2/0/0
      192.1.1.1/32  Direct 0    0             D  127.0.0.1       InLoopBack0
      192.1.1.2/32  Direct 0    0             D  192.1.1.2       GigabitEthernet2/0/0
```

在 ASBR 上执行 **display bgp vpnv4 all routing-table** 命令，可以看到 ASBR 上的 VPNv4 路由。

```
[ASBR1] display bgp vpnv4 all routing-table
 Local AS number : 100
```

```
BGP Local router ID is 2.2.2.9
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
 Total number of routes from all PE: 5
 Route Distinguisher: 100:1

      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  10.1.1.0/24        1.1.1.9         0          100        0      ?

 Route Distinguisher: 100:2

      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>   10.2.1.0/24        192.1.1.2                             0      200?
 *>   192.1.1.0          0.0.0.0         0                     0      ?
 *                       192.1.1.2       0                     0      200?
 *>   192.1.1.1/32       0.0.0.0         0                     0      ?

      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  10.1.1.0/24        1.1.1.9         0          100        0      ?
 *>   10.2.1.0/24        192.1.1.2                             0      200?
 *>   192.1.1.0          0.0.0.0         0                     0      ?
                         192.1.1.2       0                     0      200?
 *>   192.1.1.1/32       0.0.0.0         0                     0      ?
```

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
bgp 65001
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
#
return
```

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
```

```
 ip address 172.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.1.1.2 255.255.255.0
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 2.2.2.9 as-number 100
 peer 2.2.2.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 2.2.2.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 2.2.2.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.1.1.1 as-number 65001
#
ospf 1
 area 0.0.0.0
  network 1.1.1.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
return
```

- ASBR1 的配置文件

```
#
 sysname ASBR1
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:2
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 2.2.2.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 172.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 192.1.1.1 255.255.255.0
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
bgp 100
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
```

```
  peer 1.1.1.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 192.1.1.2 as-number 200
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 2.2.2.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
return
```

- ASBR2 的配置文件

```
#
 sysname ASBR2
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 200:2
  vpn-target 2:2 export-extcommunity
  vpn-target 2:2 import-extcommunity
#
 mpls lsr-id 3.3.3.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 162.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 192.1.1.2 255.255.255.0
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
#
bgp 200
 peer 4.4.4.9 as-number 200
 peer 4.4.4.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 4.4.4.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 4.4.4.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 192.1.1.1 as-number 100
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 3.3.3.9 0.0.0.0
  network 162.1.1.0 0.0.0.255
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 200:1
```

```
  vpn-target 2:2 export-extcommunity
  vpn-target 2:2 import-extcommunity
#
 mpls lsr-id 4.4.4.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 162.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.2.1.2 255.255.255.0
#
interface LoopBack1
 ip address 4.4.4.9 255.255.255.255
#
bgp 200
 peer 3.3.3.9 as-number 200
 peer 3.3.3.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 3.3.3.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 3.3.3.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.2.1.1 as-number 65002
#
ospf 1
 area 0.0.0.0
  network 4.4.4.9 0.0.0.0
  network 162.1.1.0 0.0.0.255
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface GigabitEthernet1/0/0
 ip address 10.2.1.1 255.255.255.0
#
bgp 65002
 peer 10.2.1.2 as-number 200
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
#
return
```

2.18.5 配置 OptionB 方式跨域 VPN 示例

通过在 ASBR 间建立单跳的 MP-EBGP 对等体，实现跨域的 VPN OptionB 方案。

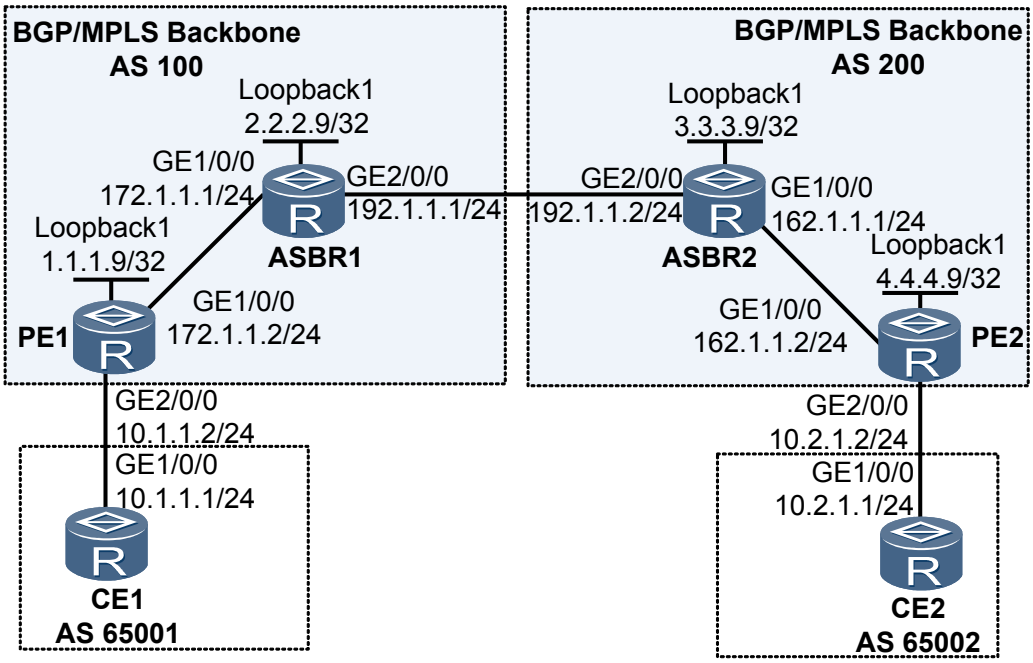
组网需求

如图 2-6，CE1 和 CE2 属于同一个 VPN。CE1 通过 AS100 的 PE1 接入，CE2 通过 AS200 的 PE2 接入。

采用 OptionB 方式实现跨域的 BGP/MPLS IP VPN：

- ASBR1 与 ASBR2 间通过 MP-EBGP 交换 VPNv4 路由；
- ASBR 不对接收的 VPNv4 路由进行 VPN-Target 过滤。

图 2-6 跨域 VPN 组网图



配置思路

本例配置主要思路是：

1. 在骨干网上运行 IGP 协议实现同一 AS 的 ASBR 与 PE 之间的互通，并且同一 AS 的 ASBR 与 PE 之间要建立 MPLS LDP LSP。
2. PE 与 CE 之间建立 EBGP 对等体关系；PE 与 ASBR 之间建立 MP-IBGP 对等体关系。
3. 在 PE 上需配置 VPN 实例（在 ASBR 上无需配置 VPN 实例）。
4. 在 ASBR 上与另一 ASBR 相连接口上分别使能 MPLS，且 ASBR 之间建立 MP-EBGP 对等体关系，并且不对接收的 VPNv4 路由进行 VPN-target 过滤。

数据准备

为完成此配置例，需准备如下的数据：

- PE 及 ASBR 上的 MPLS LSR-ID
- PE1 和 PE2 上创建的 VPN 实例的名称、RD 和 VPN-Target

操作步骤

步骤 1 在 AS100 和 AS200 的 MPLS 骨干网上分别配置 IGP 协议，实现各自骨干网 PE 之间的互通

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，同一 AS 的 ASBR 与 PE 之间应能建立 OSPF 邻居关系，执行 **display ospf peer** 命令可以看到邻居状态为 Full。

同一 AS 的 ASBR 和 PE 能学习到对方的 Loopback 地址，并能够互相 ping 通。

步骤 2 在 AS100 和 AS200 的 MPLS 骨干网上分别配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置步骤请参见**配置 OptionA 方式跨域 VPN 示例**，此处不再赘述。

步骤 3 为 AS100 和 AS200 配置基本 BGP/MPLS IP VPN

说明

PE1 和 PE2 上的 VPN 实例的 VPN-Target 需匹配。

具体配置步骤请参见后面的配置文件，此处不再赘述。

步骤 4 配置跨域 VPN-OptionB 方式

配置 ASBR1：在与 ASBR2 相连的接口 GigabitEthernet2/0/0 上使能 MPLS。

```
<ASBR1> system-view
[ASBR1] interface gigabitethernet 2/0/0
[ASBR1-GigabitEthernet2/0/0] ip address 192.1.1.1 24
[ASBR1-GigabitEthernet2/0/0] mpls
[ASBR1-GigabitEthernet2/0/0] quit
```

配置 ASBR1：与 ASBR2 建立 MP-EBGP 对等体关系，并且不对接收的 VPNv4 路由进行 VPN-target 过滤，并且使能 ASBR1 按下一跳分标签。

```
[ASBR1] bgp 100
[ASBR1-bgp] peer 192.1.1.2 as-number 200
[ASBR1-bgp] ipv4-family vpnv4
[ASBR1-bgp-af-vpnv4] peer 192.1.1.2 enable
[ASBR1-bgp-af-vpnv4] undo policy vpn-target
[ASBR1-bgp-af-vpnv4] apply-label per-nexthop
[ASBR1-bgp-af-vpnv4] quit
[ASBR1-bgp] quit
```

说明

ASBR2 的配置与 ASBR1 类似，此处不再详述。

步骤 5 检查配置结果

上述配置完成后，CE 之间能学习到对方的接口路由，CE1 和 CE2 能够相互 ping 通。

以 CE1 的显示为例：

```
<CE1> display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 5        Routes : 5
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
       10.1.1.0/24  Direct  0    0           D   10.1.1.1        GigabitEthernet1/0/0
       10.1.1.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
       10.2.1.0/24  EBGP    255  0           D   10.1.1.2        GigabitEthernet1/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
<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=252 time=120 ms
    Reply from 10.2.1.1: bytes=56 Sequence=2 ttl=252 time=73 ms
    Reply from 10.2.1.1: bytes=56 Sequence=3 ttl=252 time=111 ms
    Reply from 10.2.1.1: bytes=56 Sequence=4 ttl=252 time=86 ms
    Reply from 10.2.1.1: bytes=56 Sequence=5 ttl=252 time=110 ms
  --- 10.2.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 73/100/120 ms
```

在 ASBR 上执行 **display bgp vpnv4 all routing-table** 命令，可以看到 ASBR 上的 VPNv4 路由。

以 ASBR1 的显示为例：

```
[ASBR1] display bgp vpnv4 all routing-table
 Local AS number : 100
BGP Local router ID is 2.2.2.9
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
 Total number of routes from all PE: 3
 Route Distinguisher: 100:1
      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn
 *>i  10.1.1.0/24        1.1.1.9         0          100        0      ?
 *>i  10.1.1.1/32        1.1.1.9         0          100        0      ?
Route Distinguisher: 200:1
      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn
 *>   10.2.1.0/24        192.1.1.2                             0      200?
```

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
bgp 65001
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
return
```

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  apply-label per-instance
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
#
mpls ldp
#
```

```
interface GigabitEthernet1/0/0
 ip address 172.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.1.1.2 255.255.255.0
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 2.2.2.9 as-number 100
 peer 2.2.2.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 2.2.2.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 2.2.2.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.1.1.1 as-number 65001
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 1.1.1.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
return
```

- ASBR1 的配置文件

```
#
 sysname ASBR1
#
 mpls lsr-id 2.2.2.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 172.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip address 192.1.1.1 255.255.255.0
 mpls
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
bgp 100
 peer 192.1.1.2 as-number 200
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 192.1.1.2 enable
  peer 1.1.1.9 enable
 #
 ipv4-family vpnv4
  undo policy vpn-target
  apply-label per-nexthop
  peer 1.1.1.9 enable
```

```
  peer 192.1.1.2 enable
#
ospf 1
 area 0.0.0.0
  network 2.2.2.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
return
```

- ASBR2 的配置文件

```
#
 sysname ASBR2
#
 mpls lsr-id 3.3.3.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 162.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip address 192.1.1.2 255.255.255.0
 mpls
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
#
bgp 200
 peer 192.1.1.1 as-number 100
 peer 4.4.4.9 as-number 200
 peer 4.4.4.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 192.1.1.1 enable
  peer 4.4.4.9 enable
 #
 ipv4-family vpnv4
  undo policy vpn-target
  apply-label per-nexthop
  peer 4.4.4.9 enable
  peer 192.1.1.1 enable
#
ospf 1
 area 0.0.0.0
  network 3.3.3.9 0.0.0.0
  network 162.1.1.0 0.0.0.255
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 200:1
  apply-label per-instance
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 4.4.4.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
```

```
 ip address 162.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.2.1.2 255.255.255.0
#
interface LoopBack1
 ip address 4.4.4.9 255.255.255.255
#
bgp 200
 peer 3.3.3.9 as-number 200
 peer 3.3.3.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 3.3.3.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 3.3.3.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.2.1.1 as-number 65002
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 4.4.4.9 0.0.0.0
  network 162.1.1.0 0.0.0.255
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface GigabitEthernet1/0/0
 ip address 10.2.1.1 255.255.255.0
#
bgp 65002
 peer 10.2.1.2 as-number 200
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
#
return
```

2.18.6 配置 OptionC 方式跨域 VPN 示例

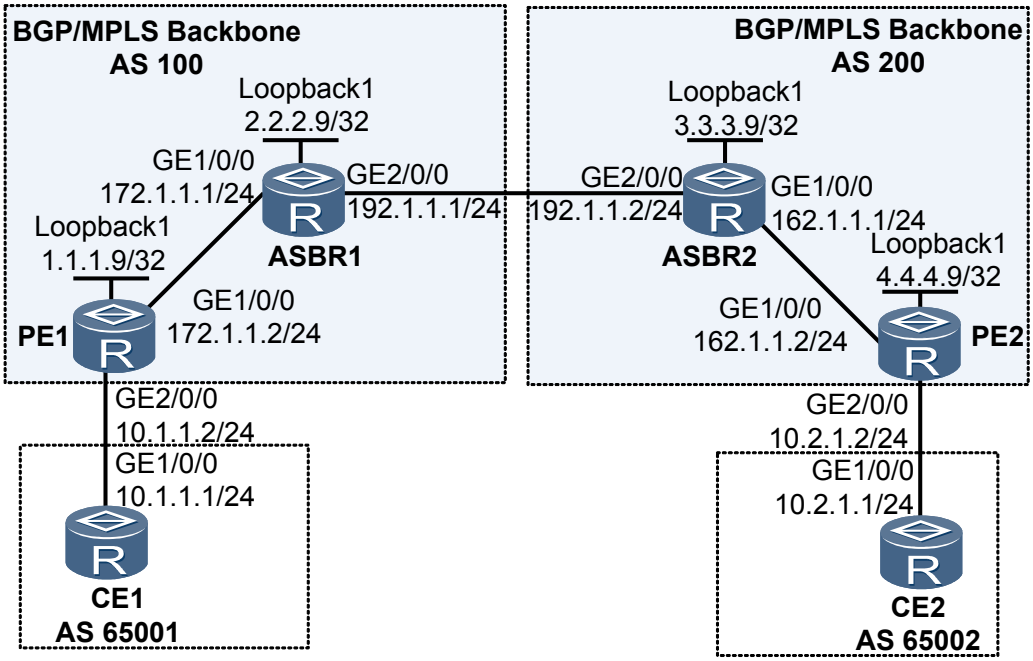
通过在不同 AS 的 PE 之间建立多跳的 MP-EBGP 对等体，实现跨域的 VPN OptionC 方案。

组网需求

如图 2-7，CE1 和 CE2 属于同一个 VPN。CE1 通过 AS100 的 PE1 接入，CE2 通过 AS200 的 PE2 接入。

采用 OptionC 方式实现跨域的 BGP/MPLS IP VPN。

图 2-7 跨域 VPN 组网图



配置思路

本例配置主要思路是：

1. 在不同 AS 间的 PE 间建立 MP-EBGP 对等体关系，并配置 PE 之间的最大跳数。
2. 在 ASBR 上配置路由策略：对从本 AS 的 PE 接收的带有 MPLS Token 的 Loopback 路由，在向对端 ASBR 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。
3. PE 与本 AS 的 ASBR 之间能够交换带标签的 IPv4 路由。
4. ASBR 与对端 ASBR 之间能够交换带标签的 IPv4 路由。

数据准备

为完成此配置例，需准备如下的数据：

- PE 及 ASBR 上的 MPLS LSR-ID
- PE 上创建的 VPN 实例名、路由标志 RD 及收发路由属性 VPN-Target
- ASBR 上配置的路由策略

操作步骤

步骤 1 在 AS100 和 AS200 的 MPLS 骨干网上分别配置 IGP 协议，实现各自骨干网内部 PE 和 ASBR 的互通

本例中 IGP 协议采用 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，同一 AS 的 ASBR 与 PE 之间应能建立 OSPF 邻居关系，执行 **display ospf peer** 命令可以看到邻居状态为 Full。

以 PE1 为例：

```
<PE1> display ospf peer
         OSPF Process 1 with Router ID 1.1.1.9
                  Neighbors
 Area 0.0.0.0 interface 172.1.1.2(GigabitEthernet1/0/0)'s neighbors
 Router ID: 2.2.2.9          Address: 172.1.1.1
   State: Full  Mode:Nbr is  Master  Priority: 1
   DR: None   BDR: None   MTU: 0
   Dead timer due in 31  sec
   Neighbor is up for 00:28:11
   Authentication Sequence: [ 0 ]
```

同一 AS 的 ASBR 和 PE 能学习到对方的 Loopback1 的 IP 地址，并能够互相 ping 通。

步骤 2 在 AS100 和 AS200 的 MPLS 骨干网上分别配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置步骤请参见**配置 OptionA 方式跨域 VPN 示例**，此处不再赘述。

步骤 3 为 AS100 和 AS200 分别配置 IPv4 地址族的 IBGP 对等体关系

具体配置请参见后面的配置文件，此处不赘述。

步骤 4 在 PE 上配置 VPN 实例，并接入 CE

具体配置请参见后面的配置文件，此处不再赘述。

说明

PE1 的 VPN 实例的 import VPN-Target 需要匹配 PE2 的 VPN 实例的 export VPN-Target；PE2 的 VPN 实例的 import VPN-Target 需要匹配 PE1 的 VPN 实例的 export VPN-Target。

步骤 5 配置标签 IPv4 路由交换

配置 PE1：使能与 ASBR1 交换标签 IPv4 路由的能力。

```
[PE1] bgp 100
[PE1-bgp] peer 2.2.2.9 label-route-capability
[PE1-bgp] quit
```

配置 ASBR1：在与 ASBR2 相连的接口 GigabitEthernet2/0/0 上使能 MPLS。

```
[ASBR1] interface gigabitethernet 2/0/0
[ASBR1-GigabitEthernet2/0/0] ip address 192.1.1.1 24
[ASBR1-GigabitEthernet2/0/0] mpls
[ASBR1-GigabitEthernet2/0/0] quit
```

配置 ASBR1：创建路由策略。

```
[ASBR1] route-policy policy1 permit node 1
[ASBR1-route-policy] apply mpls-label
[ASBR1-route-policy] quit
[ASBR1] route-policy policy2 permit node 1
[ASBR1-route-policy] if-match mpls-label
[ASBR1-route-policy] apply mpls-label
[ASBR1-route-policy] quit
```

配置 ASBR1：对向 PE1 发布的路由应用路由策略，并使能与 PE1 交换标签 IPv4 路由的能力。

```
[ASBR1] bgp 100
[ASBR1-bgp] peer 1.1.1.9 route-policy policy2 export
[ASBR1-bgp] peer 1.1.1.9 label-route-capability
```

#配置 ASBR1：对向 ASBR2 发布的路由应用路由策略，并使能与 ASBR2 交换标签 IPv4 路由的能力。

```
[ASBR1-bgp] peer 192.1.1.2 as-number 200
[ASBR1-bgp] peer 192.1.1.2 route-policy policy1 export
[ASBR1-bgp] peer 192.1.1.2 label-route-capability
[ASBR1-bgp] quit
```

#配置 ASBR1：将 PE1 的带有 MPLS token 的 Loopback 路由发布给 ASBR2，进而发布给 PE2。

```
[ASBR1] route-policy policy3 permit node 1
[ASBR1-route-policy] if-match mpls-token
[ASBR1-route-policy] quit
[ASBR1] bgp 100
[ASBR1-bgp] network 1.1.1.9 32 route-policy policy3
[ASBR1-bgp] quit
```

说明

PE2、ASBR2 上的配置分别与 PE1、ASBR1 类似，此处不再详述。

步骤 6 PE1 与 PE2 建立 MP-EBGP 对等体关系

#配置 PE1。

```
[PE1] bgp 100
[PE1-bgp] peer 4.4.4.9 as-number 200
[PE1-bgp] peer 4.4.4.9 connect-interface LoopBack 1
[PE1-bgp] peer 4.4.4.9 ebgp-max-hop 10
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 4.4.4.9 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

#配置 PE2。

```
[PE2] bgp 200
[PE2-bgp] peer 1.1.1.9 as-number 100
[PE2-bgp] peer 1.1.1.9 connect-interface LoopBack 1
[PE2-bgp] peer 1.1.1.9 ebgp-max-hop 10
[PE2-bgp] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 1.1.1.9 enable
[PE2-bgp-af-vpnv4] quit
[PE2-bgp] quit
```

步骤 7 检查配置结果

上述配置完成后，CE 之间能学习到对方的接口路由，CE1 和 CE2 能够相互 ping 通。

以 CE1 的显示为例：

```
[CE1] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 5        Routes : 5
Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface
       10.1.1.0/24  Direct 0    0           D  10.1.1.1        GigabitEthernet1/0/0
       10.1.1.1/32  Direct 0    0           D  127.0.0.1       InLoopBack0
       10.2.1.0/24  EBGP   255  0           D  10.1.1.2        GigabitEthernet1/0/0
      127.0.0.0/8   Direct 0    0           D  127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct 0    0           D  127.0.0.1       InLoopBack0
[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=252 time=102 ms
    Reply from 10.2.1.1: bytes=56 Sequence=2 ttl=252 time=89 ms
    Reply from 10.2.1.1: bytes=56 Sequence=3 ttl=252 time=106 ms
    Reply from 10.2.1.1: bytes=56 Sequence=4 ttl=252 time=104 ms
```

```
  Reply from 10.2.1.1: bytes=56 Sequence=5 ttl=252 time=56 ms
--- 10.2.1.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 56/91/106 ms
```

ASBR 上没有 VPNv4 路由。在 ASBR 上执行 **display bgp routing-table label** 命令，可以看到路由的标签信息。

以 ASBR1 为例：

```
[ASBR1] display bgp routing-table label
Total Number of Routes: 2
 BGP Local router ID is 2.2.2.9
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
      Network           NextHop           In/Out Label
 *>   1.1.1.9           172.1.1.2         15360/NULL
 *>   4.4.4.9           192.1.1.2         15361/15361
```

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
bgp 65001
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
#
return
```

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 172.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.1.1.2 255.255.255.0
#
interface LoopBack1
```

```
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 2.2.2.9 as-number 100
 peer 2.2.2.9 connect-interface LoopBack1
 peer 4.4.4.9 as-number 200
 peer 4.4.4.9 ebgp-max-hop 10
 peer 4.4.4.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 2.2.2.9 enable
  peer 2.2.2.9 label-route-capability
  peer 4.4.4.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 4.4.4.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.1.1.1 as-number 65001
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 1.1.1.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
return
```

- ASBR1 的配置文件

```
#
 sysname ASBR1
#
 mpls lsr-id 2.2.2.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 172.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip address 192.1.1.1 255.255.255.0
 mpls
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
bgp 100
 peer 192.1.1.2 as-number 200
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  network 1.1.1.9 255.255.255.255 route-policy policy3
  peer 192.1.1.2 enable
  peer 192.1.1.2 route-policy policy1 export
  peer 192.1.1.2 label-route-capability
  peer 1.1.1.9 enable
  peer 1.1.1.9 route-policy policy2 export
  peer 1.1.1.9 label-route-capability
#
ospf 1
 area 0.0.0.0
  network 2.2.2.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
```

```
#
route-policy policy1 permit node 1
 apply mpls-label
route-policy policy2 permit node 1
 if-match mpls-label
route-policy policy3 permit node 1
 if-match mpls-token
 apply mpls-label
#
return
```

- ASBR2 的配置文件

```
#
 sysname ASBR2
#
 mpls lsr-id 3.3.3.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 162.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip address 192.1.1.2 255.255.255.0
 mpls
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
#
bgp 200
 peer 192.1.1.1 as-number 100
 peer 4.4.4.9 as-number 200
 peer 4.4.4.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  network 4.4.4.9 255.255.255.255 route-policy policy3
  peer 192.1.1.1 enable
  peer 192.1.1.1 route-policy policy1 export
  peer 192.1.1.1 label-route-capability
  peer 4.4.4.9 enable
  peer 4.4.4.9 route-policy policy2 export
  peer 4.4.4.9 label-route-capability
#
ospf 1
 area 0.0.0.0
  network 3.3.3.9 0.0.0.0
  network 162.1.1.0 0.0.0.255
#
route-policy policy1 permit node 1
 apply mpls-label
route-policy policy2 permit node 1
 if-match mpls-label
 apply mpls-label
route-policy policy3 permit node 1
 if-match mpls-token
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 200:1
  vpn-target 1:1 export-extcommunity
```

```
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 4.4.4.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 162.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.2.1.2 255.255.255.0
#
interface LoopBack1
 ip address 4.4.4.9 255.255.255.255
#
bgp 200
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 ebgp-max-hop 10
 peer 1.1.1.9 connect-interface LoopBack1
 peer 3.3.3.9 as-number 200
 peer 3.3.3.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
  peer 3.3.3.9 enable
  peer 3.3.3.9 label-route-capability
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.9 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.2.1.1 as-number 65002
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 4.4.4.9 0.0.0.0
  network 162.1.1.0 0.0.0.255
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface GigabitEthernet1/0/0
 ip address 10.2.1.1 255.255.255.0
#
bgp 65002
 peer 10.2.1.2 as-number 200
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
#
return
```

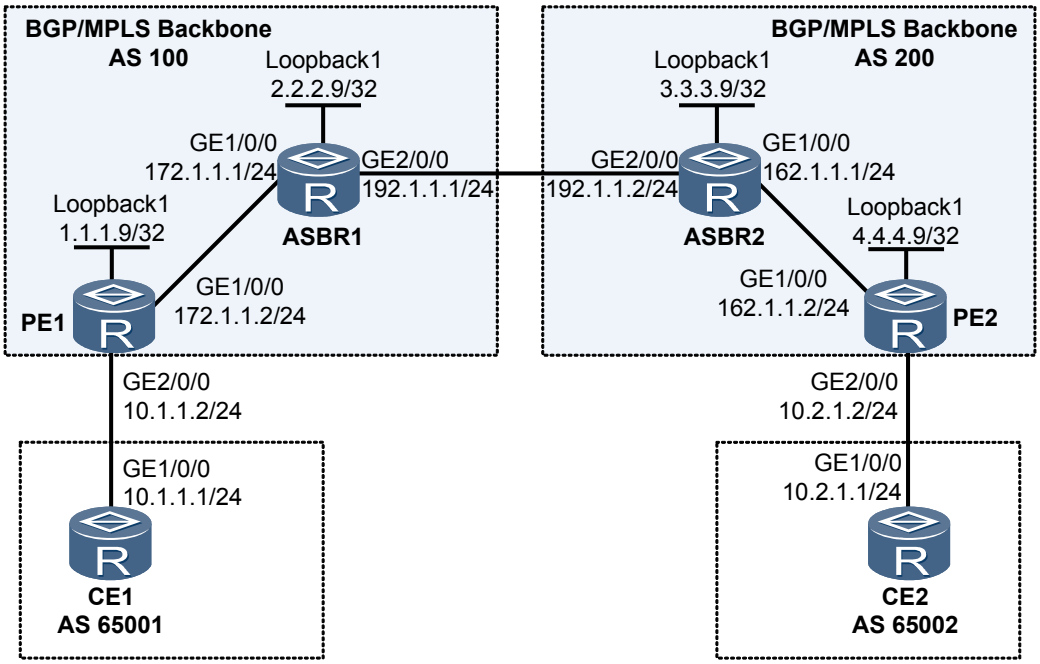
2.18.7 配置 OptionC 方式跨域 VPN 示例（方案二）

PE 和 ASBR 之间没有建立 MP-IBGP 关系时，也可以通过 LDP 为 BGP 分标签，实现跨域的 VPN OptionC 方案。

组网需求

如图 2-8 所示，CE1 和 CE2 属于同一个 VPN。CE1 通过 AS100 的 PE1 接入，CE2 通过 AS200 的 PE2 接入。

图 2-8 跨域 VPN 组网图



在 PE 和 ASBR 之间不用配置 IBGP 邻居关系，当 ASBR 从对端的 ASBR 学到对端 AS 域内的带标签 BGP 公网路由后，通过在 ASBR 上将 BGP 路由引入 IGP 协议之中，LDP 就能够为这些路由分配标签，触发建立跨域的 LDP LSP。这样就能实现 OptionC 方式跨域的 BGP/MPLS IP VPN。

配置思路

本例配置主要思路是：

1. 将域内 PE 的路由发布给对端 PE：先在本端 ASBR 上通过 BGP 将域内 PE 的路由发布给对端 ASBR，在远端 ASBR 上将 BGP 路由引入到 IGP，则远端 PE 就依靠 IGP 学到了本端域内 PE 的路由。
2. 在 ASBR 上配置路由策略：对从本 AS 的 PE 接收的带有 MPLS Token 的 Loopback 路由，在向对端 ASBR 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。
3. ASBR 与对端 ASBR 之间能够交换带标签的 IPv4 路由。
4. 在 ASBR 上配置为带标签的公网 BGP 路由建立 LDP LSP。
5. 在不同 AS 间的 PE 间建立 MP-EBGP 对等体关系；不同 AS 间的 PE 通常不是直连的，为了在它们之间建立 EBGP 连接，需要配置 PE 之间允许的最大跳数。

数据准备

为完成此配置例，需准备如下的数据：

- PE 及 ASBR 上的 MPLS LSR-ID
- PE 上创建的 VPN 实例名、路由标志 RD 及收发路由属性 VPN-Target
- ASBR 上的路由策略

操作步骤

步骤 1 在 AS100 和 AS200 的 MPLS 骨干网上分别配置 IGP 协议，实现各自骨干网内部 PE 和 ASBR 的互通

本例中 IGP 协议采用 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，同一 AS 的 ASBR 与 PE 之间应能建立 OSPF 邻居关系，执行 **display ospf peer** 命令可以看到邻居状态为 Full。

以 PE1 为例：

```
<PE1> display ospf peer

         OSPF Process 1 with Router ID 1.1.1.9
                 Neighbors

 Area 0.0.0.0 interface 172.1.1.2(GigabitEthernet1/0/0)'s neighbors
 Router ID: 2.2.2.9          Address: 172.1.1.1
   State: Full  Mode:Nbr is  Master  Priority: 1
   DR: None   BDR: None   MTU: 0
   Dead timer due in 28  sec
   Neighbor is up for 00:01:04
   Authentication Sequence: [ 0 ]
```

同一 AS 的 ASBR 和 PE 能学习到对方的 Loopback1 的 IP 地址，并能够互相 ping 通。

步骤 2 在 ASBR 间建立 EBGP 对等体

配置 ASBR1：

```
[ASBR1] bgp 100
[ASBR1-bgp] peer 192.1.1.2 as-number 200
[ASBR1-bgp] quit
```

配置 ASBR2：

```
[ASBR2] bgp 200
[ASBR2-bgp] peer 192.1.1.1 as-number 100
[ASBR2-bgp] quit
```

配置完成后，在 ASBR 上执行 **display bgp peer** 命令可以看到邻居状态为“**Established**”。

以 ASBR1 为例：

```
[ASBR1] display bgp peer

 BGP local router ID : 2.2.2.9
 Local AS number : 100
 Total number of peers : 1                 Peers in established state : 1

  Peer        V  AS    MsgRcvd  MsgSent  OutQ  Up/Down       State       PrefRcv

  192.1.1.2   4 200        129      134     0 01:39:21 Established            1
```

步骤 3 将域内 PE 的路由发送给对端 PE。

配置 ASBR1：将 PE1 的带有 MPLS token 的 Loopback 地址发布给 ASBR2。

```
[ASBR1] route-policy policy0 permit node 1
[ASBR1-route-policy] if-match mpls-token
[ASBR1-route-policy] quit
[ASBR1] bgp 100
[ASBR1-bgp] network 1.1.1.9 32 route-policy policy0
[ASBR1-bgp] quit
```

配置 ASBR2：将 PE2 的带有 MPLS token 的 Loopback 地址发布给 ASBR1。

```
[ASBR2] route-policy policy0 permit node 1
[ASBR2-route-policy] if-match mpls-token
[ASBR2-route-policy] quit
[ASBR2] bgp 200
[ASBR2-bgp] network 4.4.4.9 32 route-policy policy0
[ASBR2-bgp] quit
```

配置 ASBR1：将 BGP 路由引入到 OSPF，通过 OSPF 将 PE2 的路由发布给 PE1。

```
[ASBR1] ospf 1
[ASBR1-ospf-1] import-route bgp
```

配置 ASBR2：将 BGP 路由引入到 OSPF，通过 OSPF 将 PE1 的路由发布给 PE2。

```
[ASBR2] ospf 1
[ASBR2-ospf-1] import-route bgp
```

配置完成后，在 PE 上执行 **display ip routing-table** 命令查看路由表，以 PE1 为例：

```
[PE1] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 8        Routes : 8

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        1.1.1.9/32  Direct  0    0           D   127.0.0.1       InLoopBack0
        2.2.2.9/32  OSPF    10   1           D   172.1.1.1       GigabitEthernet1/0/0
        4.4.4.9/32  O_ASE   150  1           D   172.1.1.1       GigabitEthernet1/0/0
      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
      172.1.1.0/24  Direct  0    0           D   172.1.1.2       GigabitEthernet1/0/0
      172.1.1.1/32  Direct  0    0           D   172.1.1.1       GigabitEthernet1/0/0
      172.1.1.2/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

步骤 4 在 AS100 和 AS200 的 MPLS 骨干网上分别配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 PE1 的 MPLS 基本能力，并在与 ASBR1 相连的接口上使能 LDP。

```
[PE1] mpls lsr-id 1.1.1.9
[PE1] mpls
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
[PE1] interface gigabitethernet 1/0/0
[PE1-GigabitEthernet1/0/0] mpls
[PE1-GigabitEthernet1/0/0] mpls ldp
[PE1-GigabitEthernet1/0/0] quit
```

配置 ASBR1 的 MPLS 基本能力，并在与 PE1 相连的接口上使能 LDP。

```
[ASBR1] mpls lsr-id 2.2.2.9
[ASBR1] mpls
[ASBR1-mpls] quit
```

```
[ASBR1] mpls ldp
[ASBR1-mpls-ldp] quit
[ASBR1] interface gigabitethernet 1/0/0
[ASBR1-GigabitEthernet1/0/0] mpls
[ASBR1-GigabitEthernet1/0/0] mpls ldp
[ASBR1-GigabitEthernet1/0/0] quit
```

配置 ASBR2 的 MPLS 基本能力，并在与 PE2 相连的接口上使能 LDP。

```
[ASBR2] mpls lsr-id 3.3.3.9
[ASBR2] mpls
[ASBR2-mpls] quit
[ASBR2] mpls ldp
[ASBR2-mpls-ldp] quit
[ASBR2] interface gigabitethernet 1/0/0
[ASBR2-GigabitEthernet1/0/0] mpls
[ASBR2-GigabitEthernet1/0/0] mpls ldp
[ASBR2-GigabitEthernet1/0/0] quit
```

配置 PE2 的 MPLS 基本能力，并在与 ASBR2 相连的接口上使能 LDP。

```
[PE2] mpls lsr-id 4.4.4.9
[PE2] mpls
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface gigabitethernet 1/0/0
[PE2-GigabitEthernet1/0/0] mpls
[PE2-GigabitEthernet1/0/0] mpls ldp
[PE2-GigabitEthernet1/0/0] quit
```

上述配置完成后，PE1 与 ASBR1、ASBR2 与 PE2 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 项为“Operational”。执行 **display mpls ldp lsp** 命令，可以看到 LDP LSP 的建立情况。

以 PE1 的显示为例：

```
[PE1] display mpls ldp session

 LDP Session(s) in Public Network
 Codes: LAM(Label Advertisement Mode), SsnAge Unit(DDDD:HH:MM)
 A '*' before a session means the session is being deleted.
 ------------------------------------------------------------------------------
 PeerID             Status      LAM  SsnRole  SsnAge      KASent/Rcv
 ------------------------------------------------------------------------------
 2.2.2.9:0          Operational DU   Passive  0000:00:01  5/5
 ------------------------------------------------------------------------------
 TOTAL: 1 session(s) Found.

<PE1> display mpls ldp lsp
 LDP LSP Information
 ------------------------------------------------------------------------------
 DestAddress/Mask   In/OutLabel   UpstreamPeer   NextHop         OutInterface
 ------------------------------------------------------------------------------
 1.1.1.9/32         3/NULL        2.2.2.9        127.0.0.1       InLoop0
*1.1.1.9/32         Liberal
 2.2.2.9/32         NULL/3        -              172.1.1.1       GigabitEthernet1/0/0
 2.2.2.9/32         1024/3        2.2.2.9        172.1.1.1       GigabitEthernet1/0/0
 ------------------------------------------------------------------------------
 TOTAL: 3 Normal LSP(s) Found.
 TOTAL: 1 Liberal LSP(s) Found.
 TOTAL: 0 Frr LSP(s) Found.
 A '*' before an LSP means the LSP is not established
 A '*' before a Label means the USCB or DSCB is stale
 A '*' before a UpstreamPeer means the session is in GR state
 A '*' before a NextHop means the LSP is FRR LSP
```

步骤 5 在 ASBR 上配置标签 IPv4 路由交换能力

配置 ASBR1：在与 ASBR2 相连的接口 GigabitEthernet2/0/0 上使能 MPLS。

```
[ASBR1] interface gigabitethernet 2/0/0
[ASBR1-GigabitEthernet2/0/0] ip address 192.1.1.1 24
[ASBR1-GigabitEthernet2/0/0] mpls
[ASBR1-GigabitEthernet2/0/0] quit
```

配置 ASBR1：创建路由策略。

```
[ASBR1] route-policy policy1 permit node 1
[ASBR1-route-policy] apply mpls-label
[ASBR1-route-policy] quit
```

配置 ASBR1：对向 ASBR2 发布的路由应用路由策略，并使能与 ASBR2 交换标签 IPv4 路由的能力。

```
[ASBR1] bgp 100
[ASBR1-bgp] peer 192.1.1.2 route-policy policy1 export
[ASBR1-bgp] peer 192.1.1.2 label-route-capability
[ASBR1-bgp] quit
```

说明

ASBR2 上的配置分别与 ASBR1 类似，请参见配置文件，此处不再详述。

步骤 6 在 ASBR 上配置为带标签的公网 BGP 路由建立 LDP LSP

配置 ASBR1。

```
[ASBR1] mpls
[ASBR1-mpls] lsp-trigger bgp-label-route
[ASBR1-mpls] quit
```

配置 ASBR2。

```
[ASBR2] mpls
[ASBR2-mpls] lsp-trigger bgp-label-route
[ASBR2-mpls] quit
```

步骤 7 在 PE 上配置 VPN 实例，并接入 CE

配置 PE1。

```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] ipv4-family
[PE1-vpn-instance-vpn1-af-ipv4] route-distinguisher 100:1
[PE1-vpn-instance-vpn1-af-ipv4] vpn-target 1:1 export-extcommunity
[PE1-vpn-instance-vpn1-af-ipv4] vpn-target 1:1 import-extcommunity
[PE1-vpn-instance-vpn1-af-ipv4] quit
[PE1-vpn-instance-vpn1] quit
[PE1] interface gigabitethernet 2/0/0
[PE1-GigabitEthernet2/0/0] ip binding vpn-instance vpn1
[PE1-GigabitEthernet2/0/0] ip address 10.1.1.2 24
[PE1-GigabitEthernet2/0/0] quit
```

配置 PE2。

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] ipv4-family
[PE2-vpn-instance-vpn1-af-ipv4] route-distinguisher 200:1
[PE2-vpn-instance-vpn1-af-ipv4] vpn-target 1:1 export-extcommunity
[PE2-vpn-instance-vpn1-af-ipv4] vpn-target 1:1 import-extcommunity
[PE2-vpn-instance-vpn1-af-ipv4] quit
[PE2-vpn-instance-vpn1] quit
[PE2] interface gigabitethernet 2/0/0
[PE2-GigabitEthernet2/0/0] ip binding vpn-instance vpn1
[PE2-GigabitEthernet2/0/0] ip address 10.2.1.2 24
[PE2-GigabitEthernet2/0/0] quit
```

配置完成后，在 PE 设备上执行 **display ip vpn-instance verbose** 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 PE1 和 CE1 为例：

```
[PE1] display ip vpn-instance verbose
  Total VPN-Instances configured : 1

 VPN-Instance Name and ID : vpn1, 1
  Interfaces : GigabitEthernet2/0/0
 Address family ipv4
  Create date : 2008/02/27 09:53:47
  Up time : 0 days, 00 hours, 35 minutes and 43 seconds
  Route Distinguisher : 100:1
  Export VPN Targets :  1:1
  Import VPN Targets :  1:1
  Label Policy : label per route
  Log Interval : 5
[PE1] ping -vpn-instance vpn1 10.1.1.1
  PING 10.1.1.1: 56  data bytes, press CTRL_C to break
    Request time out
    Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=255 time=50 ms
    Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=255 time=40 ms
    Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=255 time=30 ms
    Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=255 time=10 ms

  --- 10.1.1.1 ping statistics ---
    5 packet(s) transmitted
    4 packet(s) received
    20.00% packet loss
round-trip min/avg/max = 10/32/50 ms
```

步骤 8 在 PE1 与 PE2 之间建立 MP-EBGP 对等体关系

配置 PE1。

```
[PE1] bgp 100
[PE1-bgp] peer 4.4.4.9 as-number 200
[PE1-bgp] peer 4.4.4.9 connect-interface LoopBack 1
[PE1-bgp] peer 4.4.4.9 ebgp-max-hop 10
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 4.4.4.9 enable
[PE1-bgp-af-vpnv4] quit
[PE1-bgp] quit
```

配置 PE2。

```
[PE2] bgp 200
[PE2-bgp] peer 1.1.1.9 as-number 100
[PE2-bgp] peer 1.1.1.9 connect-interface LoopBack 1
[PE2-bgp] peer 1.1.1.9 ebgp-max-hop 10
[PE2-bgp] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 1.1.1.9 enable
[PE2-bgp-af-vpnv4] quit
[PE2-bgp] quit
```

步骤 9 在 PE 与 CE 之间建立 EBGP 对等体关系，引入 VPN 路由

配置 CE1。

```
[CE1] bgp 65001
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] import-route direct
[CE1-bgp] quit
```

配置 CE2。

```
[CE2] bgp 65002
[CE2-bgp] peer 10.2.1.2 as-number 200
```

```
[CE2-bgp] import-route direct
[CE2-bgp] quit
```

配置 PE1。

```
[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
```

配置 PE2。

```
[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
```

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 PE1 与 CE1 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

VPN-Instance vpn1, router ID 1.1.1.9:
 Total number of peers : 1                 Peers in established state : 1

  Peer            V    AS  MsgRcvd  MsgSent  OutQ  Up/Down       State PrefRcv
  10.1.1.1        4 65001        3        3     0 00:00:52 Established       1
```

步骤 10 检查配置结果

上述配置完成后，CE 之间能学习到对方的接口路由，CE1 和 CE2 能够相互 ping 通。

以 CE1 的显示为例：

```
[CE1] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 5        Routes : 5
Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface
   10.1.1.0/24      Direct 0    0           D  10.1.1.1        GigabitEthernet1/0/0
   10.1.1.1/32      Direct 0    0           D  127.0.0.1       InLoopBack0
   10.2.1.0/24      EBGP   255  0           D  10.1.1.2        GigabitEthernet1/0/0
  127.0.0.0/8       Direct 0    0           D  127.0.0.1       InLoopBack0
  127.0.0.1/32      Direct 0    0           D  127.0.0.1       InLoopBack0

[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=252 time=102 ms
    Reply from 10.2.1.1: bytes=56 Sequence=2 ttl=252 time=89 ms
    Reply from 10.2.1.1: bytes=56 Sequence=3 ttl=252 time=106 ms
    Reply from 10.2.1.1: bytes=56 Sequence=4 ttl=252 time=104 ms
    Reply from 10.2.1.1: bytes=56 Sequence=5 ttl=252 time=56 ms

  --- 10.2.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 56/91/106 ms
```

配置完成后，在 ASBR1 上执行 **display ip routing-table** *dest-ip-address* **verbose** 命令，可以看到 ASBR1 到 PE2 的路由为带标签的公网 BGP 路由：Routing Table 为“Public”，协议类型为“BGP”，标签值不为零。

以 ASBR1 的显示为例：

```
[ASBR1] display ip routing-table 4.4.4.9 verbose
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Table : Public
Summary Count : 1

 Destination    : 4.4.4.9/32
    Protocol    : BGP            Process ID    : 0
  Preference    : 255                  Cost    : 1
     NextHop    : 192.1.1.2       Neighbour    : 192.1.1.2
       State    : Active Adv           Age    : 00h12m53s
         Tag    : 0                Priority    : 0
       Label    : 15360             QoSInfo    : 0x0
  IndirectID    : 0x0
RelayNextHop    : 0.0.0.0         Interface    : GigabitEthernet2/0/0
    TunnelID    : 0x6002006           Flags    : D
```

并且在 ASBR1 和 PE2 上分别执行 **display mpls lsp protocol ldp include** *dest-ip-address* **verbose** 命令，可以看到 ASBR1 和 PE2 之间建立了一条 LDP LSP，并且在 PE 上可以看到到达对端 PE 的 LDP Ingress LSP。

```
[ASBR1] display mpls lsp protocol ldp include 4.4.4.9 32 verbose
------------------------------------------------------------------------------
                LSP Information: LDP LSP
------------------------------------------------------------------------------
  No                  :  1
  VrfIndex            :
  Fec                 :  4.4.4.9/32
  Nexthop             :  192.1.1.2
  In-Label            :  1024
  Out-Label           :  NULL
  In-Interface        :  ----------
  Out-Interface       :  ----------
  LspIndex            :  13313
  Token               :  0x0
  FrrToken            :  0x0
  LsrType             :  Egress
  Outgoing token      :  0x6002006
  Label Operation     :  POPGO
  Mpls-Mtu            :  ------
  TimeStamp           :  15829sec
  Bfd-State           :  ---
```

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
bgp 65001
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
```

```
#
return
```

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
mpls lsr-id 1.1.1.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 172.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.1.1.2 255.255.255.0
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 4.4.4.9 as-number 200
 peer 4.4.4.9 ebgp-max-hop 10
 peer 4.4.4.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 4.4.4.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 4.4.4.9 enable
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  peer 10.1.1.1 as-number 65001
#
ospf 1
 area 0.0.0.0
  network 1.1.1.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
return
```

- ASBR1 的配置文件

```
#
 sysname ASBR1
#
mpls lsr-id 2.2.2.9
 mpls
  lsp-trigger bgp-label-route
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 172.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
```

```
 ip address 192.1.1.1 255.255.255.0
 mpls
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
bgp 100
 peer 192.1.1.2 as-number 200
 #
 ipv4-family unicast
  undo synchronization
  network 1.1.1.9 255.255.255.255 route-policy policy0
  peer 192.1.1.2 enable
  peer 192.1.1.2 route-policy policy1 export
  peer 192.1.1.2 label-route-capability
#
ospf 1
 import-route bgp
 area 0.0.0.0
  network 2.2.2.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
route-policy policy0 permit node 1
 if-match mpls-token
route-policy policy1 permit node 1
 apply mpls-label
#
return
```

- ASBR2 的配置文件

```
#
 sysname ASBR2
#
mpls lsr-id 3.3.3.9
 mpls
  lsp-trigger bgp-label-route
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 162.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip address 192.1.1.2 255.255.255.0
 mpls
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
#
bgp 200
 peer 192.1.1.1 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  network 4.4.4.9 255.255.255.255 route-policy policy0
  peer 192.1.1.1 enable
  peer 192.1.1.1 route-policy policy1 export
  peer 192.1.1.1 label-route-capability
#
ospf 1
 import-route bgp
 area 0.0.0.0
  network 3.3.3.9 0.0.0.0
  network 162.1.1.0 0.0.0.255
#
route-policy policy0 permit node 1
 if-match mpls-token
route-policy policy1 permit node 1
```

```
 apply mpls-label
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 200:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 4.4.4.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 162.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.2.1.2 255.255.255.0
#
interface LoopBack1
 ip address 4.4.4.9 255.255.255.255
#
bgp 200
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 ebgp-max-hop 10
 peer 1.1.1.9 connect-interface LoopBack1
#
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
#
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.9 enable
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  peer 10.2.1.1 as-number 65002
#
ospf 1
 area 0.0.0.0
  network 4.4.4.9 0.0.0.0
  network 162.1.1.0 0.0.0.255
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface GigabitEthernet1/0/0
 ip address 10.2.1.1 255.255.255.0
#
bgp 65002
 peer 10.2.1.2 as-number 200
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
```

```
#
return
```

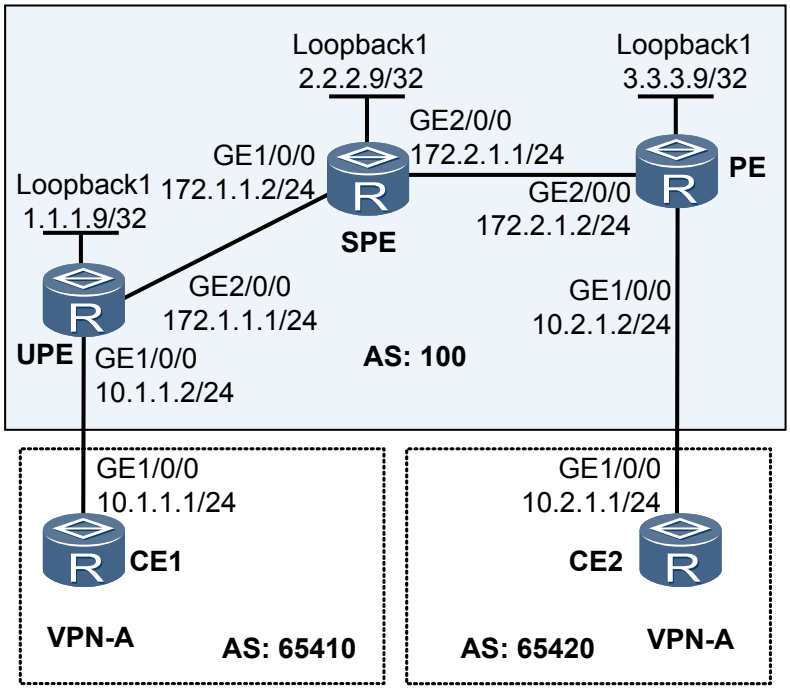
2.18.8 配置 HoVPN 示例

通过配置 HoVPN，使多个 PE 承担不同的角色，并形成层次结构，共同完成一个 PE 功能，以降低对 PE 的性能要求。

组网需求

如图 2-9，CE1、CE2 属于同一个 VPN，使用的 VPN-target 属性也一样；CE1 通过 UPE 接入，CE2 通过普通 PE 接入。UPE、SPE 与 PE 之间配置 OSPF 实现互通。

图 2-9 分层式 BGP/MPLS IP VPN 组网图



配置思路

本例配置主要思路是：

1. 先在骨干网上配置 IGP 协议实现互通，并且 PE 之间能互相学习到对方的 Loopback 地址。
2. PE 之间建立 MPLS 标签分发路径 LSP。
3. UPE 上创建 VPN 实例，并且与 CE1 建立 EBGP 对等体关系；
4. PE 上创建 VPN 实例，并与 CE2 建立 EBGP 对等体关系。
5. UPE 与 SPE、PE 与 SPE 之间建立 MP-IBGP 对等体关系。
6. 在 SPE 上创建 VPN 实例，指定 UPE 为自己的下层 PE（或称为用户层 PE），并向 UPE 发布 VPN 实例的缺省路由。

数据准备

为完成此配置例，需准备如下的数据：

- UPE、SPE 及 PE 上的 MPLS LSR-ID
- UPE、SPE 及 PE 上创建的 VPN 实例名称，RD 和 VPN-Target

操作步骤

步骤 1 在 MPLS 骨干网上配置 OSPF，实现互通

配置完成后，UPE 与 SPE、PE 与 SPE 之间应能建立 OSPF 邻居关系，执行 **display ospf peer** 命令可以看到邻居状态为 Full。执行 **display ip routing-table** 命令可以看到 PE 之间学习到对方的 Loopback 路由。

具体配置过程略。

步骤 2 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置完成后，UPE 与 SPE、SPE 与 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 Session State 项为“Operational”。执行 **display mpls ldp lsp** 命令，可以看到 LDP LSP 的建立情况。

具体配置过程略。

步骤 3 配置 PE 接入 CE，PE 与 CE 之间使用 BGP

配置 UPE。

```
<UPE> system-view
[UPE] ip vpn-instance vpna
[UPE-vpn-instance-vpna] ipv4-family
[UPE-vpn-instance-vpna-af-ipv4] route-distinguisher 100:1
[UPE-vpn-instance-vpna-af-ipv4] vpn-target 1:1
[UPE-vpn-instance-vpna-af-ipv4] quit
[UPE-vpn-instance-vpna] quit
[UPE] interface gigabitethernet 1/0/0
[UPE-GigabitEthernet1/0/0] ip binding vpn-instance vpna
[UPE-GigabitEthernet1/0/0] ip address 10.1.1.2 24
[UPE-GigabitEthernet1/0/0] quit
[UPE] bgp 100
[UPE-bgp] ipv4-family vpn-instance vpna
[UPE-bgp-vpna] peer 10.1.1.1 as-number 65410
[UPE-bgp-vpna] import-route direct
[UPE-bgp-vpna] quit
[UPE-bgp] quit
```

配置 CE1。

```
<Huawei> system-view
[Huawei] sysname CE1
[CE1] interface gigabitethernet 1/0/0
[CE1-GigabitEthernet1/0/0] ip address 10.1.1.1 24
[CE1-GigabitEthernet1/0/0] quit
[CE1] bgp 65410
[CE1-bgp] peer 10.1.1.2 as-number 100
[CE1-bgp] import-route direct
[CE1-bgp] quit
```

配置 PE。

```
<PE> system-view
[PE] ip vpn-instance vpna
[PE-vpn-instance-vpna] ipv4-family
```

```
[PE-vpn-instance-vpna-af-ipv4] route-distinguisher 100:2
[PE-vpn-instance-vpna-af-ipv4] vpn-target 1:1
[PE-vpn-instance-vpna-af-ipv4] quit
[PE-vpn-instance-vpna] quit
[PE] interface gigabitethernet 1/0/0
[PE-GigabitEthernet1/0/0] ip binding vpn-instance vpna
[PE-GigabitEthernet1/0/0] ip address 10.2.1.2 24
[PE-GigabitEthernet1/0/0] quit
[PE] bgp 100
[PE-bgp] ipv4-family vpn-instance vpna
[PE-bgp-vpna] peer 10.2.1.1 as-number 65420
[PE-bgp-vpna] import-route direct
[PE-bgp-vpna] quit
[PE-bgp] quit
```

配置 CE2。

```
<Huawei> system-view
[Huawei] sysname CE2
[CE2] interface gigabitethernet 1/0/0
[CE2-GigabitEthernet1/0/0] ip address 10.2.1.1 24
[CE2-GigabitEthernet1/0/0] quit
[CE2] bgp 65420
[CE2-bgp] peer 10.2.1.2 as-number 100
[CE2-bgp] import-route direct
[CE2-bgp] quit
```

配置完成后，在 UPE 和 PE 上执行 **display ip vpn-instance verbose** 命令可以看到 VPN 实例的配置情况。UPE 和 PE 能用 **ping -vpn-instance** 命令 ping 通自己接入的 CE。

说明

当 PE 上有多个绑定了同一个 VPN 的接口，则使用 **ping -vpn-instance** 命令 ping 对端 PE 接入的 CE 时，要指定源 IP 地址，即要指定 **ping -vpn-instance** *vpn-instance-name* **-a** *source-ip-address dest-ip-address* 命令中的参数**-a** *source-ip-address*，否则可能 ping 不通。

步骤 4 配置 UPE 与 SPE、PE 与 SPE 的 MP-IBGP 对等体关系

配置 UPE。

```
<UPE> system-view
[UPE] bgp 100
[UPE-bgp] peer 2.2.2.9 as-number 100
[UPE-bgp] peer 2.2.2.9 connect-interface loopback 1
[UPE-bgp] ipv4-family vpnv4
[UPE-bgp-af-vpnv4] peer 2.2.2.9 enable
[UPE-bgp-af-vpnv4] quit
[UPE-bgp] quit
```

配置 SPE。

```
<SPE> system-view
[SPE] bgp 100
[SPE-bgp] peer 1.1.1.9 as-number 100
[SPE-bgp] peer 1.1.1.9 connect-interface loopback 1
[SPE-bgp] peer 3.3.3.9 as-number 100
[SPE-bgp] peer 3.3.3.9 connect-interface loopback 1
[SPE-bgp] ipv4-family vpnv4
[SPE-bgp-af-vpnv4] peer 1.1.1.9 enable
[SPE-bgp-af-vpnv4] peer 3.3.3.9 enable
[SPE-bgp-af-vpnv4] quit
[SPE-bgp] quit
```

配置 PE。

```
<PE> system-view
[PE] bgp 100
[PE-bgp] peer 2.2.2.9 as-number 100
[PE-bgp] peer 2.2.2.9 connect-interface loopback 1
```

```
[PE-bgp] ipv4-family vpnv4
[PE-bgp-af-vpnv4] peer 2.2.2.9 enable
[PE-bgp-af-vpnv4] quit
[PE-bgp] quit
```

步骤 5 配置 SPE

配置 VPN 实例。

```
[SPE] ip vpn-instance vpna
[SPE-vpn-instance-vpna] ipv4-family
[SPE-vpn-instance-vpna-af-ipv4] route-distinguisher 200:1
[SPE-vpn-instance-vpna-af-ipv4] vpn-target 1:1
[SPE-vpn-instance-vpna-af-ipv4] quit
[SPE-vpn-instance-vpna] quit
```

指定自己的 UPE。

```
[SPE] bgp 100
[SPE-bgp] ipv4-family vpnv4
[SPE-bgp-af-vpnv4] peer 1.1.1.9 upe
```

向 UPE 发布 VPN 实例的缺省路由。

```
[SPE-bgp-af-vpnv4] peer 1.1.1.9 default-originate vpn-instance vpna
[SPE-bgp-af-vpnv4] quit
```

步骤 6 检查配置结果

配置完成后，CE1 上没有到 CE2 接口网段的路由，但有一条下一跳为 UPE 的缺省路由；CE2 上有到 CE1 接口网段的 BGP 路由。CE1 和 CE2 可以相互 Ping 通。

```
<CE1> display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 5        Routes : 5

  Destination/Mask  Proto  Pre  Cost      Flags  NextHop              Interface

        0.0.0.0/0   EBGP    255  0             D  10.1.1.2             GigabitEthernet1/0/0
       10.1.1.0/24  Direct 0    0             D  10.1.1.1             GigabitEthernet1/0/0
       10.1.1.1/32  Direct 0    0             D  127.0.0.1            InLoopBack0
      127.0.0.0/8   Direct 0    0             D  127.0.0.1            InLoopBack0
      127.0.0.1/32  Direct 0    0             D  127.0.0.1            InLoopBack0
[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=253 time=85 ms
    Reply from 10.2.1.1: bytes=56 Sequence=2 ttl=253 time=70 ms
    Reply from 10.2.1.1: bytes=56 Sequence=3 ttl=253 time=57 ms
    Reply from 10.2.1.1: bytes=56 Sequence=4 ttl=253 time=66 ms
    Reply from 10.2.1.1: bytes=56 Sequence=5 ttl=253 time=55 ms
  --- 10.2.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 55/66/85 ms
[CE2] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 5        Routes : 5

  Destination/Mask  Proto  Pre  Cost        Flags  NextHop              Interface

       10.1.1.0/24  EBGP    255  0               D  10.2.1.2             GigabitEthernet1/0/0
       10.2.1.0/24  Direct 0    0               D  10.2.1.1             GigabitEthernet1/0/0
       10.2.1.1/32  Direct 0    0               D  127.0.0.1            InLoopBack0
      127.0.0.0/8   Direct 0    0               D  127.0.0.1            InLoopBack0
      127.0.0.1/32  Direct 0    0               D  127.0.0.1            InLoopBack0
```

在 UPE 上执行 **display bgp vpnv4 all routing-table** 命令，可以看到有一条 VPN 实例 vpna 的缺省路由，下一跳为 SPE。

```
[UPE] display bgp vpnv4 all routing-table
 BGP Local router ID is 1.1.1.9
 Status codes: * - valid, > - best, d - damped,
               h - history,  i - internal, s - suppressed, S - Stale
               Origin : i - IGP, e - EGP, ? - incomplete
 Total number of routes from all PE: 3
 Route Distinguisher: 100:1

      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>   10.1.1.0/24        0.0.0.0         0                    0      ?
 *                       10.1.1.1        0                    0      65410?

 Route Distinguisher: 200:1

      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  0.0.0.0            2.2.2.9         0          100       0      i

      Network            NextHop        MED        LocPrf    PrefVal Path/Ogn

 *>i  0.0.0.0            2.2.2.9         0          100       0      i
 *>   10.1.1.0/24        0.0.0.0         0                    0      ?
 *                       10.1.1.1        0                    0      65410?
```

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
bgp 65410
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
#
return
```

- UPE 的配置文件

```
#
 sysname UPE
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 100:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip binding vpn-instance vpna
 ip address 10.1.1.2 255.255.255.0
#
```

```
interface GigabitEthernet2/0/0
 ip address 172.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 2.2.2.9 as-number 100
 peer 2.2.2.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 2.2.2.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 2.2.2.9 enable
#
 ipv4-family vpn-instance vpna
  peer 10.1.1.1 as-number 65410
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 1.1.1.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
return
```

- SPE 的配置文件

```
#
 sysname SPE
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 200:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 2.2.2.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 172.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip address 172.2.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
bgp 100
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 peer 3.3.3.9 as-number 100
 peer 3.3.3.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
  peer 3.3.3.9 enable
 #
```

```
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.9 enable
  peer 1.1.1.9 upe
  peer 1.1.1.9 default-originate vpn-instance vpna
  peer 3.3.3.9 enable
#
ospf 1
 area 0.0.0.0
  network 2.2.2.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
  network 172.2.1.0 0.0.0.255
#
return
```

- PE 的配置文件

```
#
 sysname PE
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 100:2
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 3.3.3.9
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip binding vpn-instance vpna
 ip address 10.2.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 172.2.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 3.3.3.9 255.255.255.255
#
bgp 100
 peer 2.2.2.9 as-number 100
 peer 2.2.2.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 2.2.2.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 2.2.2.9 enable
#
 ipv4-family vpn-instance vpna
  peer 10.2.1.1 as-number 65420
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 3.3.3.9 0.0.0.0
  network 172.2.1.0 0.0.0.255
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface GigabitEthernet1/0/0
```

```
 ip address 10.2.1.1 255.255.255.0
#
bgp 65420
 peer 10.2.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
#
return
```

2.18.9 配置 Multi-VPN-Instance CE 示例

通过在 CE 上使用 OSPF 多实例实现局域网不同业务的隔离。

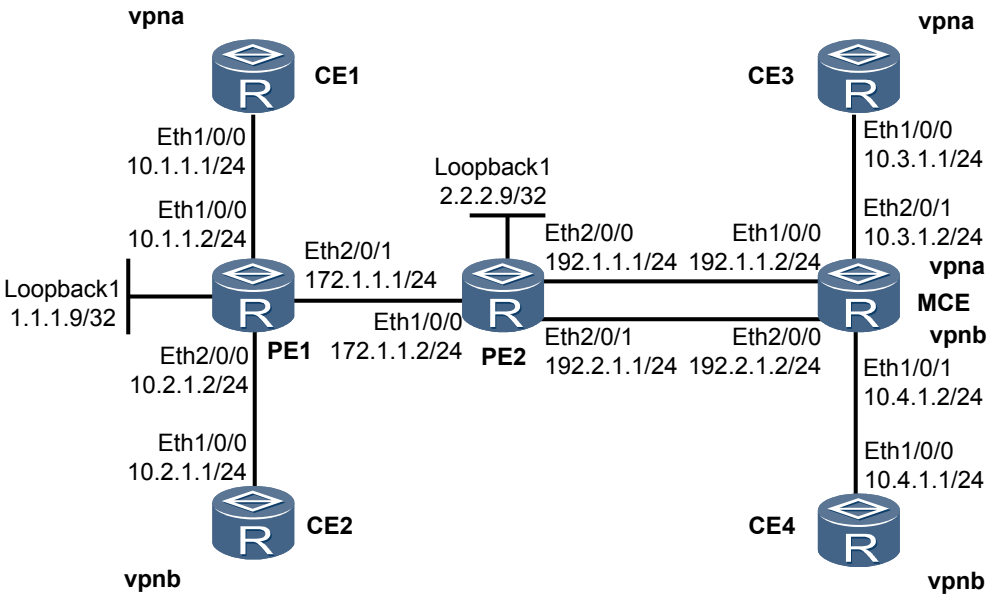
组网需求

如图 2-10 所示，按如下组网：

- CE1 和 CE2 属于同一局域网，MCE，CE3 和 CE4 属于同一局域网。
- MCE 是用户端用作 VPN 多实例交换的设备。
- CE1、CE3 属于 VPN 实例 vpna，CE2、CE4 属于 VPN 实例 vpnb。
- vpna 和 vpnb 使用不同的 VPN-Target 属性。

要求属于相同 VPN 的用户之间能互相访问，但不同 VPN 的用户之间不能互相访问，从而实现局域网内不同 VPN 之间的业务隔离。

图 2-10 Multi-VPN-Instance CE 示例组网图



配置思路

本例配置主要思路是：

1. PE 与 PE 之间配置 OSPF 实现 PE 之间的互通、配置 MP-IBGP 交换 VPN 路由信息。
2. PE 与相连的 CE 之间建立 EBGP 对等体，把 VPN 路由引入 PE 的 VPN 路由表中。
3. MCE 与 PE2 之间配置 OSPF 多实例，交换 VPN 路由信息；MCE 与 CE3、CE4 之间配置 RIP-2 交换 VPN 路由信息。

说明

MCE 与 PE2 之间配置 OSPF 多实例时需进行以下配置。

在 PE2 的 OSPF 视图下（该 OSPF 进程是指在 MCE 与 PE2 之间配置 OSPF 多实例时所使用的 OSPF 进程）引入 BGP 路由，发布 PE1 的私网路由给 MCE;

在 PE2 的 BGP 视图下引入该 OSPF 进程（也是指在 MCE 与 PE2 之间配置 OSPF 多实例时所使用的 OSPF 进程），发布 MCE 的私网路由信息给 PE1 上。

数据准备

为完成此配置例，需准备如下的数据：

- 在 PE1、PE2 及 MCE 上为每个相互隔离的业务创建一个 VPN 实例（不同 VPN 实例的 VPN-Target 互不相同，相同 VPN 实例的 VPN-Target 应相同）
- 配置 OSPF 所需数据（为不同业务配置 OSPF 多实例时所使用的 OSPF 进程号应互不相同）
- 在 MCE 上为引入 CE3、CE4 的 VPN 路由所使用的 RIP 进程（应互不相同）

操作步骤

步骤 1 在骨干网的 PE 上配置 OSPF 协议，实现 PE 之间的互通

具体配置过程略。

完成此步配置后，PE 之间应能互相学习到对方的 Loopback1 的地址。

以 PE2 为例：

```
<PE2> display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        1.1.1.9/32  OSPF    10   2           D   172.1.1.1       Ethernet1/0/0
        2.2.2.9/32  Direct  0    0           D   127.0.0.1       InLoopBack0
       127.0.0.0/8  Direct  0    0           D   127.0.0.1       InLoopBack0
       127.0.0.1/32 Direct  0    0           D   127.0.0.1       InLoopBack0
       172.1.1.0/24 Direct  0    0           D   172.1.1.2       Ethernet1/0/0
       172.1.1.1/32 Direct  0    0           D   172.1.1.1       Ethernet1/0/0
       172.1.1.2/32 Direct  0    0           D   127.0.0.1       InLoopBack0
```

步骤 2 在骨干网的 PE 上配置 MPLS 基本能力和 MPLS LDP，PE 之间建立 LDP LSP

具体配置过程略。

完成此步配置后，在 PE 上执行命令 **display mpls ldp session**，应能看见 PE 之间的 MPLS LDP 会话状态为“Operational”。

以 PE2 为例：

```
<PE2> display mpls ldp session
 LDP Session(s) in Public Network
 Codes: LAM(Label Advertisement Mode), SsnAge Unit(DDDD:HH:MM)
 A '*' before a session means the session is being deleted.
```

```
--------------------------------------------------------------------------------
 PeerID             Status      LAM  SsnRole  SsnAge      KASent/Rcv
--------------------------------------------------------------------------------
1.1.1.9:0          Operational DU   Active   0000:00:04  17/17
--------------------------------------------------------------------------------
 TOTAL: 1 session(s) Found.
```

步骤 3 在 PE 设备上配置 VPN 实例，将 CE1、CE2 接入 PE1，将 MCE 接入 PE2

配置 PE1。

```
<PE1> system-view
[PE1] ip vpn-instance vpna
[PE1-vpn-instance-vpna] ipv4-family
[PE1-vpn-instance-vpna-af-ipv4] route-distinguisher 100:1
[PE1-vpn-instance-vpna-af-ipv4] vpn-target 111:1 both
[PE1-vpn-instance-vpna-af-ipv4] quit
[PE1-vpn-instance-vpna] quit
[PE1] ip vpn-instance vpnb
[PE1-vpn-instance-vpnb] ipv4-family
[PE1-vpn-instance-vpnb-af-ipv4] route-distinguisher 100:2
[PE1-vpn-instance-vpnb-af-ipv4] vpn-target 222:2 both
[PE1-vpn-instance-vpnb-af-ipv4] quit
[PE1-vpn-instance-vpnb] quit
[PE1] interface ethernet1/0/0
[PE1-Ethernet1/0/0] ip binding vpn-instance vpna
[PE1-Ethernet1/0/0] ip address 10.1.1.2 24
[PE1-Ethernet1/0/0] quit
[PE1] interface ethernet2/0/0
[PE1-Ethernet2/0/0] ip binding vpn-instance vpnb
[PE1-Ethernet2/0/0] ip address 10.2.1.2 24
[PE1-Ethernet2/0/0] quit
```

配置 PE2。

```
<PE2> system-view
[PE2] ip vpn-instance vpna
[PE2-vpn-instance-vpna] ipv4-family
[PE2-vpn-instance-vpna-af-ipv4] route-distinguisher 200:1
[PE2-vpn-instance-vpna-af-ipv4] vpn-target 111:1 both
[PE2-vpn-instance-vpna-af-ipv4] quit
[PE2-vpn-instance-vpna] quit
[PE2] ip vpn-instance vpnb
[PE2-vpn-instance-vpnb] ipv4-family
[PE2-vpn-instance-vpnb-af-ipv4] route-distinguisher 200:2
[PE2-vpn-instance-vpnb-af-ipv4] vpn-target 222:2 both
[PE2-vpn-instance-vpnb-af-ipv4] quit
[PE2-vpn-instance-vpnb] quit
[PE2] interface ethernet2/0/0
[PE2-Ethernet2/0/0] ip binding vpn-instance vpna
[PE2-Ethernet2/0/0] ip address 192.1.1.1 24
[PE2-Ethernet2/0/0] quit
[PE2]interface ethernet2/0/1
[PE2-Ethernet2/0/1] ip binding vpn-instance vpnb
[PE2-Ethernet2/0/1] ip address 192.2.1.1 24
[PE2-Ethernet2/0/1] quit
```

步骤 4 在 MCE 设备上配置 VPN 实例，将 CE3、CE4 及 PE2 接入 MCE

```
<Huawei> system-view
[Huawei] sysname MCE
[MCE] ip vpn-instance vpna
[MCE-vpn-instance-vpna] ipv4-family
[MCE-vpn-instance-vpna-af-ipv4] route-distinguisher 300:1
[MCE-vpn-instance-vpna-af-ipv4] vpn-target 111:1 both
[MCE-vpn-instance-vpna-af-ipv4] quit
[MCE-vpn-instance-vpna] quit
[MCE] ip vpn-instance vpnb
[MCE-vpn-instance-vpnb] ipv4-family
[MCE-vpn-instance-vpnb-af-ipv4] route-distinguisher 300:2
```

```
[MCE-vpn-instance-vpnb-af-ipv4] vpn-target 222:2 both
[MCE-vpn-instance-vpnb-af-ipv4] quit
[MCE-vpn-instance-vpnb] quit
[MCE] interface ethernet2/0/1
[MCE-Ethernet2/0/1] ip binding vpn-instance vpna
[MCE-Ethernet2/0/1] ip address 10.3.1.2 24
[MCE-Ethernet2/0/1] quit
[MCE] interface ethernet1/0/1
[MCE-Ethernet1/0/1] ip binding vpn-instance vpnb
[MCE-Ethernet1/0/1] ip address 10.4.1.2 24
[MCE-Ethernet1/0/1] quit
[MCE] interface ethernet1/0/0
[MCE-Ethernet1/0/0] ip binding vpn-instance vpna
[MCE-Ethernet1/0/0] ip address 192.1.1.2 24
[MCE-Ethernet1/0/0] quit
[MCE] interface ethernet2/0/0
[MCE-Ethernet2/0/0] ip binding vpn-instance vpnb
[MCE-Ethernet2/0/0] ip address 192.2.1.2 24
[MCE-Ethernet2/0/0] quit
```

步骤 5 在 PE 之间建立 MP-IBGP 对等体，在 PE1 与 CE1、CE2 之间建立 EBGP 对等体

具体配置过程略。

完成此步配置后，在 PE1 上执行命令 **display bgp vpnv4 all peer** 可以看见 PE1 与 PE2 的 IBGP 对等体关系及 PE1 与 CE1、CE2 之间建立 EBGP 对等体关系均为“Established”。

```
[PE1] display bgp vpnv4 all peer
 BGP local router ID : 1.1.1.9
 Local AS number : 100
 Total number of peers : 3                 Peers in established state : 3
  Peer            V    AS  MsgRcvd  MsgSent    OutQ  Up/Down       State PrefRcv
  2.2.2.9         4   100       13       10       0 00:03:45 Established       6
  Peer of IPv4-family for vpn instance :

 VPN-Instance vpna, router ID 1.1.1.9:
  10.1.1.1        4 65410        9       11       0 00:04:14 Established       2
 VPN-Instance vpnb, router ID 1.1.1.9:
  10.2.1.1        4 65420        9       12       0 00:04:09 Established       2
```

步骤 6 在 PE2 和 MCE 之间配置 OSPF 多实例

配置 PE2。

```
<PE2> system-view
[PE2] ospf 100 vpn-instance vpna
[PE2-ospf-100] area 0
[PE2-ospf-100-area-0.0.0.0] network 192.1.1.0 0.0.0.255
[PE2-ospf-100-area-0.0.0.0] quit
[PE2-ospf-100] import-route bgp
[PE2-ospf-100] quit
[PE2] ospf 200 vpn-instance vpnb
[PE2-ospf-200] area 0
[PE2-ospf-200-area-0.0.0.0] network 192.2.1.0 0.0.0.255
[PE2-ospf-200-area-0.0.0.0] quit
[PE2-ospf-200] import-route bgp
[PE2-ospf-200] quit
[PE2] bgp 100
[PE2-bgp] ipv4-family vpn-instance vpna
[PE2-bgp-vpna] import-route ospf 100
[PE2-bgp-vpna] quit
[PE2-bgp] ipv4-family vpn-instance vpnb
[PE2-bgp-vpnb] import-route ospf 200
[PE2-bgp-vpnb] quit
```

配置 MCE。

```
<MCE> system-view
[MCE] ospf 100 vpn-instance vpna
```

```
[MCE-ospf-100] area 0
[MCE-ospf-100-area-0.0.0.0] network 192.1.1.0 0.0.0.255
[MCE-ospf-100-area-0.0.0.0] quit
[MCE-ospf-100] quit
[MCE] ospf 200 vpn-instance vpnb
[MCE-ospf-200] area 0
[MCE-ospf-200-area-0.0.0.0] network 192.2.1.0 0.0.0.255
[MCE-ospf-200-area-0.0.0.0] quit
[MCE-ospf-200] quit
```

步骤 7 在 MCE 和 CE3、CE4 之间配置 RIP-2

配置 MCE。

```
[MCE] rip 100 vpn-instance vpna
[MCE-rip-100] version 2
[MCE-rip-100] network 10.0.0.0
[MCE-rip-100] import-route ospf 100
[MCE-rip-100] quit
[MCE] rip 200 vpn-instance vpnb
[MCE-rip-200] version 2
[MCE-rip-200] network 10.0.0.0
[MCE-rip-200] import-route ospf 200
```

配置 CE3。

```
<Huawei> system-view
[Huawei] sysname CE3
[CE3] rip 100
[CE3-rip-100] version 2
[CE3-rip-100] network 10.0.0.0
[CE3-rip-100] import-route direct
```

配置 CE4。

```
<Huawei> system-view
[Huawei] sysname CE4
[CE4] rip 200
[CE4-rip-200] version 2
[CE4-rip-200] network 10.0.0.0
[CE4-rip-200] import-route direct
```

步骤 8 在 MCE 上配置不进行环路检查，并引入 RIP 路由

```
<MCE> system-view
[MCE] ospf 100 vpn-instance vpna
[MCE-ospf-100] vpn-instance-capability simple
[MCE-ospf-100] import-route rip 100
[MCE] ospf 200 vpn-instance vpnb
[MCE-ospf-200] vpn-instance-capability simple
[MCE-ospf-200] import-route rip 200
```

步骤 9 检查配置结果

完成上述配置后，在 MCE 设备上执行命令 **display ip routing-table vpn-instance** 命令，可以看到去往对端 CE 的路由。

以 vpna 为例：

```
[MCE] display ip routing-table vpn-instance vpna
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpna
         Destinations : 8        Routes : 8
  Destination/Mask  Proto  Pre  Cost       Flags  NextHop         Interface
       10.1.1.0/24  O_ASE  150  1              D  192.1.1.1       Ethernet1/0/0
       10.1.1.1/32  O_ASE  150  1              D  192.1.1.1       Ethernet1/0/0
       10.3.1.0/24  Direct 0    0              D  10.3.1.2        Ethernet2/0/1
       10.3.1.1/32  Direct 0    0              D  10.3.1.1        Ethernet2/0/1
       10.3.1.2/32  Direct 0    0              D  127.0.0.1       InLoopBack0
```

```
      192.1.1.0/24  Direct 0    0             D  192.1.1.2       Ethernet1/0/0
      192.1.1.1/32  Direct 0    0             D  192.1.1.1       Ethernet1/0/0
      192.1.1.2/32  Direct 0    0             D  127.0.0.1       InLoopBack0
```

在 PE 上执行 **display ip routing-table vpn-instance** 命令，可以看到去往对端 CE 的路由。

以 PE1 上的 vpna 为例：

```
[PE1] display ip routing-table vpn-instance vpna
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpna
         Destinations : 5        Routes : 5
  Destination/Mask  Proto  Pre  Cost       Flags  NextHop         Interface
       10.1.1.0/24  Direct 0    0             D  10.1.1.2        Ethernet1/0/0
       10.1.1.1/32  Direct 0    0             D  10.1.1.1        Ethernet1/0/0
       10.1.1.2/32  Direct 0    0             D  127.0.0.1       InLoopBack0
       10.3.1.0/24  EBGP    255  2             RD  2.2.2.9         Ethernet2/0/1
      192.1.1.0/24  EBGP    255  0             RD  2.2.2.9         Ethernet2/0/1
```

CE1、CE3 之间可以互通，CE2、CE4 之间可以互通。

以 CE1 为例：

```
[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=252 time=125 ms
    Reply from 10.3.1.1: bytes=56 Sequence=2 ttl=252 time=125 ms
    Reply from 10.3.1.1: bytes=56 Sequence=3 ttl=252 time=125 ms
    Reply from 10.3.1.1: bytes=56 Sequence=4 ttl=252 time=125 ms
    Reply from 10.3.1.1: bytes=56 Sequence=5 ttl=252 time=125 ms
  --- 10.3.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 125/125/125 ms
```

CE1 不能与 CE2 和 CE4 互通，CE3 也不能与 CE2 和 CE4 互通。

以 CE1 上 ping CE4 的显示为例。

```
[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
```

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
interface Ethernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
bgp 65410
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
```

```
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface Ethernet1/0/0
 ip address 10.2.1.1 255.255.255.0
#
bgp 65420
 peer 10.2.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
#
return
```

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 100:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
ip vpn-instance vpnb
 ipv4-family
  route-distinguisher 100:2
  vpn-target 222:2 export-extcommunity
  vpn-target 222:2 import-extcommunity
#
 mpls lsr-id 1.1.1.9
 mpls
#
mpls ldp
#
interface Ethernet1/0/0
 ip binding vpn-instance vpna
 ip address 10.1.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip binding vpn-instance vpnb
 ip address 10.2.1.2 255.255.255.0
#
interface Ethernet2/0/1
 ip address 172.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 1.1.1.9 255.255.255.255
#
bgp 100
 peer 2.2.2.9 as-number 100
 peer 2.2.2.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 2.2.2.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
```

```
  peer 2.2.2.9 enable
 #
 ipv4-family vpn-instance vpna
  peer 10.1.1.1 as-number 65410
  import-route direct
#
 ipv4-family vpn-instance vpnb
  peer 10.2.1.1 as-number 65420
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 1.1.1.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 200:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
ip vpn-instance vpnb
 ipv4-family
  route-distinguisher 200:2
  vpn-target 222:2 export-extcommunity
  vpn-target 222:2 import-extcommunity
#
 mpls lsr-id 2.2.2.9
 mpls
#
mpls ldp
#
interface Ethernet1/0/0
 ip address 172.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface Ethernet2/0/0
 ip binding vpn-instance vpna
 ip address 192.1.1.1 255.255.255.0
#
interface Ethernet2/0/1
 ip binding vpn-instance vpnb
 ip address 192.2.1.1 255.255.255.0
#
interface LoopBack1
 ip address 2.2.2.9 255.255.255.255
#
bgp 100
 peer 1.1.1.9 as-number 100
 peer 1.1.1.9 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.9 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.9 enable
 #
 ipv4-family vpn-instance vpna
  import-route ospf 100
 #
 ipv4-family vpn-instance vpnb
```

```
  import-route ospf 200
#
ospf 1
 area 0.0.0.0
  network 2.2.2.9 0.0.0.0
  network 172.1.1.0 0.0.0.255
#
ospf 100 vpn-instance vpna
import-route bgp
 area 0.0.0.0
  network 192.1.1.0 0.0.0.255
#
ospf 200 vpn-instance vpnb
import-route bgp
 area 0.0.0.0
  network 192.2.1.0 0.0.0.255
#
return
```

- MCE 的配置文件

```
#
 sysname MCE
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 300:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
ip vpn-instance vpnb
 ipv4-family
  route-distinguisher 300:2
  vpn-target 222:2 export-extcommunity
  vpn-target 222:2 import-extcommunity
#
interface Ethernet1/0/0
 ip binding vpn-instance vpna
 ip address 192.1.1.2 255.255.255.0
#
interface Ethernet2/0/0
 ip binding vpn-instance vpnb
 ip address 192.2.1.2 255.255.255.0
#
interface Ethernet2/0/1
 ip binding vpn-instance vpna
 ip address 10.3.1.2 255.255.255.0
#
interface Ethernet1/0/1
 ip binding vpn-instance vpnb
 ip address 10.4.1.2 255.255.255.0
#
ospf 100 vpn-instance vpna
 import-route rip 100
 vpn-instance-capability simple
 area 0.0.0.0
  network 192.1.1.0 0.0.0.255
#
ospf 200 vpn-instance vpnb
 import-route rip 200
 vpn-instance-capability simple
 area 0.0.0.0
  network 192.2.1.0 0.0.0.255
#
rip 100 vpn-instance vpna
 version 2
 network 10.0.0.0
 import-route ospf 100
#
rip 200 vpn-instance vpnb
 version 2
```

```
 network 10.0.0.0
 import-route ospf 200
#
return
```

- CE3 的配置文件

```
#
 sysname CE3
#
interface Ethernet1/0/0
 ip address 10.3.1.1 255.255.255.0
#
rip 100
 version 2
 network 10.0.0.0
 import-route direct
#
return
```

- CE4 的配置文件

```
#
 sysname CE4
#
interface Ethernet1/0/0
 ip address 10.4.1.1 255.255.255.0
#
rip 200
 version 2
 network 10.0.0.0
 import-route direct
#
return
```

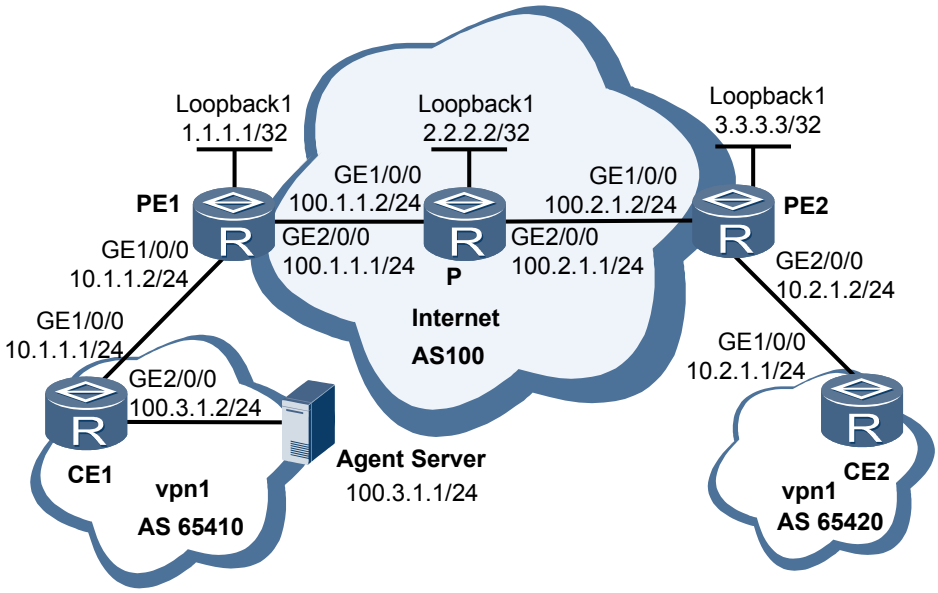
2.18.10 配置 VPN 与 Internet 互联示例

通过在 VPN 中设置代理服务器，使得 VPN 可以与 Internet 互联。

组网需求

如图 2-11，需要配置私网中的 CE1 和 CE2 可互访，同时在 CE1 上连接了一台代理服务器，具有 Internet 中的地址，CE1 的用户可以通过该代理服务器访问 Internet。本示例中，Internet 用路由器 P 来代替。

图 2-11 VPN 与 Internet 互联示例



配置思路

本例配置主要思路是：

1. 配置普通的 L3VPN；
2. 配置 3 条静态路由，具体如下：
   - 在 CE1 上增加一条默认路由，下一跳为 PE1。
   - 在 PE1 上增加一条从 VPN 到 Internet 的默认路由，下一跳为 P。此时，代理服务器上的流量就可以到达 Internet 了。
   - 在 PE1 上增加一条从 Internet 回到代理服务器的静态路由，下一跳为 CE1，并将该路由通过 IGP 发布到 Internet 上。此时，Internet 的流量就可以到达 CE1 所连接的服务器了。

数据准备

为完成此配置例，需准备如下的数据：

- PE 及 P 上的 MPLS LSR ID
- VPN 的路由区分符 RD
- VPN 的 VPN-Target

操作步骤

步骤 1 配置 IGP 协议

配置骨干网各物理接口和 Loopback 接口的 IP 地址；在骨干网 Internet 各设备上运行 IGP 协议，使 PE1、P、PE2 之间能互通，且相互之间能学到对方的 Loopback 地址。具体配置略。

步骤 2 建立 MPLS LDP LSP 和 MP-IBGP 对等体

在 PE 之间建立 MPLS 标签转发路径 LSP，并建立 MP-IBGP 对等体关系。具体配置略。

配置完后，在 P 上执行命令 **display mpls ldp session**，可以看到 PE1 与 P 之间、PE2 与 P 之间的 LDP Session State 为“Operational”。

P 上的显示：

```
<P> display mpls ldp session
 LDP Session(s) in Public Network
 Codes: LAM(Label Advertisement Mode), SsnAge Unit(DDDD:HH:MM)
 A '*' before a session means the session is being deleted.
------------------------------------------------------------------------------
 PeerID            Status      LAM  SsnRole  SsnAge      KASent/Rcv
 ------------------------------------------------------------------------------
 1.1.1.1:0         Operational DU   Active   0000:00:05  23/23
 3.3.3.3:0         Operational DU   Passive  0000:00:04  18/18
------------------------------------------------------------------------------
 TOTAL: 2 session(s) Found.
```

在 PE 上执行命令 **display bgp vpnv4 all peer**，可以看见 PE 之间的 MP-IBGP 对等体关系已建立，“State”为“Established”。以 PE1 为例：

```
<PE1> display bgp vpnv4 all peer
 BGP local router ID : 1.1.1.1
 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        6        8     0 00:03:48 Established
```

步骤 3 创建 VPN 实例和建立 EBGP

在 PE 上创建 VPN 实例 vpn1，并绑定与 CE 相连的接口。在 PE1 和 CE1 之间、PE2 和 CE2 之间建立 EBGP 对等体关系，把 CE 上的路由引入到 PE 中。具体配置略。

配置完后，在 PE 上执行命令 **display ip vpn-instance**，在显示信息中的 VPN-Instance Name 中有 vpn1。

以 PE1 为例：

```
[PE1] display ip vpn-instance
  Total VPN-Instances configured : 1
  VPN-Instance Name               Address-family
  vpn1                            ipv4
```

且在 PE 上执行命令 **display bgp vpnv4 all peer**，可以看见 IBGP 和 EBGP 对等体状态都为“Established”。

以 PE1 为例：

```
<PE1> display bgp vpnv4 all peer
 BGP local router ID : 1.1.1.1
 Local AS number : 100
 Total number of peers : 2                 Peers in established state : 2
  Peer            V    AS  MsgRcvd  MsgSent  OutQ  Up/Down       State     PrefRcv
  3.3.3.3         4   100      127      134     0 01:39:44   Established       2
 Peer of IPv4-family for vpn instance :

 VPN-Instance vpn1, router ID 1.1.1.1:
  10.1.1.1        4 65410      107      110     0 01:26:33   Established       3
```

步骤 4 配置 VPN 访问公网所需的静态路由

在 CE1 上配置一条默认路由，下一跳为 PE1。

```
<CE1> system-view
```

```
[CE1] ip route-static 0.0.0.0 0 10.1.1.2
```

配置 PE1。

配置一条从私网的代理服务器到 Internet 的默认路由，下一跳为 P，并指定下一跳 P 的地址为公网地址，即在命令中下一跳地址后面加上关键字 **public**。

```
<PE1> system-view
[PE1] ip route-static vpn-instance vpn1 0.0.0.0 0 100.1.1.2 public
```

配置一条回到代理服务器的静态路由，下一跳为 CE1。

```
[PE1] ip route-static 100.3.1.0 24 vpn-instance vpn1 10.1.1.1
```

将 PE1 上回到代理服务器的静态路由通过 IGP 发布到 Internet 上。

```
[PE1] ospf 1
[PE1-ospf-1] import-route static
```

配置代理服务器。将代理服务器的 IP 地址配置成：100.3.1.1/24，服务器的缺省网关配置成 CE1，即 100.3.1.2/24。在代理服务器上还需运行代理软件。

步骤 5 检查配置结果

在 PE1 上执行命令 **display ip routing-table vpn-instance**，可以看到 vpn1 私网路由表有条缺省路由，下一跳为 100.1.1.2，出接口为 GigabitEthernet2/0/0。

```
[PE1] display ip routing-table vpn-instance vpn1
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: vpn1
         Destinations : 7        Routes : 7

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        0.0.0.0/0   Static 60   0           RD  100.1.1.2       GigabitEthernet2/0/0
       10.1.1.0/24  Direct 0    0           D   10.1.1.2        GigabitEthernet1/0/0
       10.1.1.1/32  Direct 0    0           D   10.1.1.1        GigabitEthernet1/0/0
       10.1.1.2/32  Direct 0    0           D   127.0.0.1       InLoopBack0
       10.2.1.0/24  IBGP   255  0           RD  3.3.3.3         GigabitEthernet2/0/0
       10.2.1.1/32  IBGP   255  0           RD  3.3.3.3         GigabitEthernet2/0/0
       10.2.1.2/32  IBGP   255  0           RD  3.3.3.3         GigabitEthernet2/0/0
      100.3.1.1/32  EBGP   255  0           D   10.1.1.1        GigabitEthernet1/0/0
```

在 PE1 上执行 **display ip routing-table**，可以看见公网路由表中有到代理服务器的路由，下一跳为 10.1.1.1。

```
[PE1] display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 10       Routes : 10

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        1.1.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
        2.2.2.2/32  OSPF   10   2           D   100.1.1.2       GigabitEthernet2/0/0
        3.3.3.3/32  OSPF   10   3           D   100.1.1.2       GigabitEthernet2/0/0
      100.1.1.0/24  Direct 0    0           D   100.1.1.1       GigabitEthernet2/0/0
      100.1.1.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
      100.1.1.2/32  Direct 0    0           D   100.1.1.2       GigabitEthernet2/0/0
      100.2.1.0/24  OSPF   10   2           D   100.1.1.2       GigabitEthernet2/0/0
      100.3.1.0/24  Static 60   0           D   10.1.1.1        GigabitEthernet1/0/0
      127.0.0.0/8   Direct 0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct 0    0           D   127.0.0.1       InLoopBack0
```

P 上可以 ping 通代理服务器。

```
[P] ping 100.3.1.1
  PING 100.3.1.1: 56  data bytes, press CTRL_C to break
    Reply from 100.3.1.1: bytes=56 Sequence=1 ttl=254 time=62 ms
    Reply from 100.3.1.1: bytes=56 Sequence=2 ttl=254 time=62 ms
```

```
  Reply from 100.3.1.1: bytes=56 Sequence=3 ttl=254 time=62 ms
  Reply from 100.3.1.1: bytes=56 Sequence=4 ttl=254 time=62 ms
  Reply from 100.3.1.1: bytes=56 Sequence=5 ttl=254 time=62 ms
--- 100.3.1.1 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 62/62/62 ms
```

此时，代理服务器可以访问 P 了。

----结束

配置文件

- CE1 的配置文件

```
#
 sysname CE1
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.1 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 100.3.1.1 255.255.255.0
#
bgp 65410
 peer 10.1.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.1.1.2 enable
#
 ip route-static 0.0.0.0 0.0.0.0 10.1.1.2
#
return
```

- PE1 的配置文件

```
#
 sysname PE1
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 1.1.1.1
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip binding vpn-instance vpn1
 ip address 10.1.1.2 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 100.1.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 1.1.1.1 255.255.255.255
#
bgp 100
 peer 3.3.3.3 as-number 100
 peer 3.3.3.3 connect-interface LoopBack1
 #
```

```
 ipv4-family unicast
  undo synchronization
  peer 3.3.3.3 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 3.3.3.3 enable
 #
 ipv4-family vpn-instance vpn1
  peer 10.1.1.1 as-number 65410
  import-route static
  import-route direct
#
ospf 1
 import-route static
 area 0.0.0.0
  network 1.1.1.1 0.0.0.0
  network 100.1.1.0 0.0.0.255
#
 ip route-static 100.3.1.0 24 vpn-instance vpn1 10.1.1.1
 ip route-static vpn-instance vpn1 0.0.0.0 0.0.0.0 100.1.1.2 public
#
return
```

- P 的配置文件

```
#
 sysname P
#
 mpls lsr-id 2.2.2.2
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
 ip address 100.1.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip address 100.2.1.1 255.255.255.0
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 2.2.2.2 0.0.0.0
  network 100.1.1.0 0.0.0.255
  network 100.2.1.0 0.0.0.255
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:2
  vpn-target 1:1 export-extcommunity
  vpn-target 1:1 import-extcommunity
#
 mpls lsr-id 3.3.3.3
 mpls
#
mpls ldp
#
interface GigabitEthernet1/0/0
```

```
 ip address 100.2.1.2 255.255.255.0
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.2.1.2 255.255.255.0
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.255
#
bgp 100
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.1 enable
 #
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.1 enable
 #
 ipv4-family vpn-instance vpn1
  peer 10.2.1.1 as-number 65420
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 100.2.1.0 0.0.0.255
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface GigabitEthernet1/0/0
 ip address 10.2.1.1 255.255.255.0
#
bgp 65420
 peer 10.2.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  import-route direct
  peer 10.2.1.2 enable
#
return
```

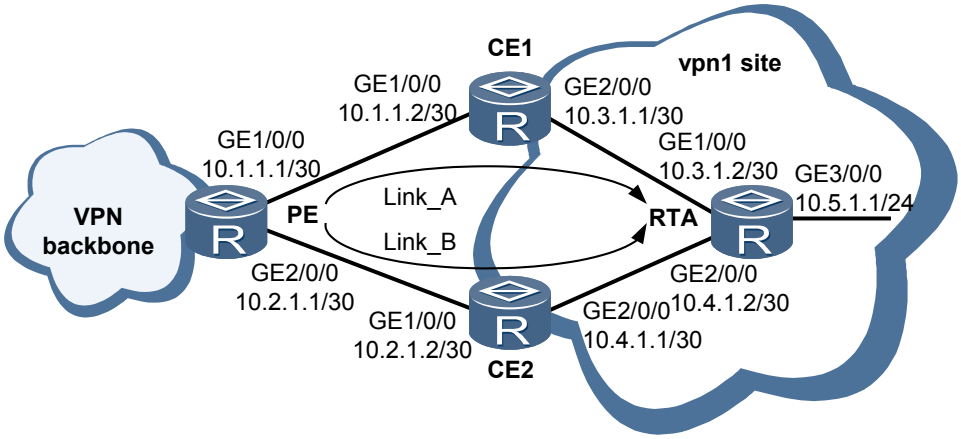
2.18.11 配置私网 IP FRR 示例

VPN site 中的多个 CE 接入到同一台 PE 上时，配置 IP FRR 特性，当 PE 与 CE 之间转发不通时，可以快速将流量切换到另一条 PE 与 CE 相连的链路上。

组网需求

如图 2-12 所示，要求在 PE 上配置私网备份出接口和备份下一跳，使链路 Link_B 为链路 Link_A 的备份，Link_A 出现故障时可以快速切换到 Link_B 上。

图 2-12 配置私网 IP FRR 功能



配置思路

采用如下的思路配置私网 IP FRR 功能。

1. 在各路由器上使能 OSPF 基本功能。
2. 在 PE 上配置 vpn1，将 GE1/0/0、GE2/0/0 绑定到 vpn1 上；并配置 OSPF 多实例。
3. 分别在 PE 和 RTA 的 GE2/0/0 接口上配置较大的 Cost 值，使 OSPF 优选链路 A。
4. 在 PE 上配置私网 IP FRR 功能。
5. 配置 BFD 感知链路状态。

数据准备

为完成此配置例，需准备如下的数据：

- PE 上的 VPN 实例名称（vpn1），RD（100:1），VPN-Target（111:1）
- PE 和 RTA 的 GE2/0/0 的 cost 值 100
- BFD 配置名和会话参数。

操作步骤

步骤 1 配置各接口的 IP 地址（略）

步骤 2 在 CE1、CE2 和 RTA 上配置 OSPF（略）

完成此步骤后，CE1、CE2 和 RTA 之间能互相学到对方各接口的地址。以 CE1 为例：

```
<CE1> display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 9        Routes : 9

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        10.1.1.0/30  Direct  0    0           D   10.1.1.2        GigabitEthernet1/0/0
        10.1.1.2/32  Direct  0    0           D   127.0.0.1       GigabitEthernet1/0/0
        10.3.1.0/30  Direct  0    0           D   10.3.1.1        GigabitEthernet2/0/0
        10.3.1.1/32  Direct  0    0           D   127.0.0.1       GigabitEthernet2/0/0
        10.2.1.0/30  OSPF    10   2           D   10.3.1.2        GigabitEthernet2/0/0
```

```
 10.4.1.0/30  OSPF   10   2             D  10.3.1.2       GigabitEthernet2/0/0
 10.5.1.0/24  OSPF   10   2             D  10.3.1.2       GigabitEthernet2/0/0
127.0.0.0/8   Direct 0    0             D  127.0.0.1       InLoopBack0
127.0.0.1/32  Direct 0    0             D  127.0.0.1       InLoopBack0
```

步骤 3 在 PE 上配置 VPN 实例及 OSPF 多实例

在 PE 上配置 vpn1，并将 GigabitEthernet1/0/0 和 GigabitEthernet2/0/0 都绑定在 vpn1 上。

```
<PE> system-view
[PE] ip vpn-instance vpn1
[PE-vpn-instance-vpn1] ipv4-family
[PE-vpn-instance-vpn1-af-ipv4] route-distinguisher 100:1
[PE-vpn-instance-vpn1-af-ipv4] vpn-target 111:1
[PE-vpn-instance-vpn1-af-ipv4] quit
[PE-vpn-instance-vpn1] quit
[PE] interface gigabitethernet 1/0/0
[PE-GigabitEthernet1/0/0] ip binding vpn-instance vpn1
[PE-GigabitEthernet1/0/0] ip address 10.1.1.1 30
[PE-GigabitEthernet1/0/0] quit
[PE] interface gigabitethernet 2/0/0
[PE-GigabitEthernet2/0/0] ip binding vpn-instance vpn1
[PE-GigabitEthernet2/0/0] ip address 10.2.1.1 30
[PE-GigabitEthernet2/0/0] quit
```

在 PE 上配置 OSPF 多实例。

```
[PE] ospf vpn-instance vpn1
[PE-ospf-1] area 0
[PE-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.3
[PE-ospf-1-area-0.0.0.0] network 10.2.1.0 0.0.0.3
```

步骤 4 配置 OSPF 接口的 Cost 值

在 PE 的 GigabitEthernet2/0/0 上配置 Cost 值，使 OSPF 优选链路 A。

```
[PE] interface gigabitethernet 2/0/0
[PE-GigabitEthernet2/0/0] ospf cost 100
[PE-GigabitEthernet2/0/0] quit
```

在 RTA 的 GigabitEthernet2/0/0 上配置 Cost 值，使 OSPF 优选链路 A。

```
[RTA] interface gigabitethernet 2/0/0
[RTA-GigabitEthernet2/0/0] ospf cost 100
[RTA-GigabitEthernet2/0/0] quit
```

步骤 5 配置路由策略

在 PE 上配置路由策略，配置备份下一跳和备份出接口。同时配置 if-match 项，限制应用范围。

```
[PE] ip ip-prefix frr1 permit 10.5.1.0 24
[PE] route-policy ip_frr_rp permit node 10
[PE-route-policy] if-match ip-prefix frr1
[PE-route-policy] apply backup-nexthop 10.2.1.2
[PE-route-policy] apply backup-interface gigabitethernet2/0/0
[PE-route-policy] quit
```

步骤 6 配置 BFD 与 IP FRR 联动

在 PE 上配置 BFD 与 IP FRR 联动

```
[PE] bfd
[PE-bfd] quit
[PE] bfd for_ip_frr bind peer-ip 10.1.1.2 vpn-instance vpn1 interface gigabitethernet1/0/0
[PE-bfd-session-for_ip_frr] discriminator local 10
[PE-bfd-session-for_ip_frr] discriminator remote 20
[PE-bfd-session-for_ip_frr] commit
```

在 CE1 上配置 BFD 与 IP FRR 联动

```
[CE1] bfd
[CE1-bfd] quit
[CE1] bfd for_ip_frr bind peer-ip 10.1.1.1 interface gigabitethernet1/0/0
[CE1-bfd-session-for_ip_frr] discriminator local 20
[CE1-bfd-session-for_ip_frr] discriminator remote 10
[CE1-bfd-session-for_ip_frr] commit
```

在 PE 和 CE1 上执行 **display bfd session all verbose** 命令，可以看到 BFD 的状态为“UP”。

步骤 7 使能私网 IP FRR 功能

```
[PE] ip vpn-instance vpn1
[PE-vpn-instance-vpn1] ipv4-family
[PE-vpn-instance-vpn1-af-ipv4] ip frr route-policy ip_frr_rp
[PE-vpn-instance-vpn1-af-ipv4] quit
[PE-vpn-instance-vpn1] quit
```

查看备份出接口和备份下一跳信息。

```
<PE> display ip routing-table vpn-instance vpn1 10.5.1.0 verbose
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Table : vpn1
Summary Count : 1
Destination: 10.5.1.0/24
     Protocol: OSPF            Process ID: 1
   Preference: 10                    Cost: 3
      NextHop: 10.1.1.2         Neighbour: 0.0.0.0
        State: Active Adv             Age: 00h00m03s
          Tag: 0                Priority: low
        Label: NULL              QoSInfo: 0x0
   IndirectID: 0x0
 RelayNextHop: 0.0.0.0          Interface: GigabitEthetnet1/0/0
     TunnelID: 0x0                  Flags: D
    BkNextHop: 10.2.1.2       BkInterface: GigabitEthetnet2/0/0
      BkLabel: NULL           SecTunnelID: 0x0
 BkPETunnelID: 0x0        BkPESecTunnelID: 0x0
 BkIndirectID: 0x0
```

----结束

配置文件

- PE 的配置文件

```
#
 sysname PE
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  ip frr route-policy ip_frr_rp
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
 bfd
#
interface GigabitEthernet1/0/0
 ip binding vpn-instance vpn1
 ip address 10.1.1.1 255.255.255.252
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.2.1.1 255.255.255.252
 ospf cost 100
#
```

```
ospf 1 vpn-instance vpn1
area 0.0.0.0
  network 10.1.1.0 0.0.0.3
  network 10.2.1.0 0.0.0.3
#
ip ip-prefix frr1 index 10 permit 10.5.1.0 24
#
route-policy ip_frr_rp permit node 10
 if-match ip-prefix frr1
 apply backup-nexthop 10.2.1.2
 apply backup-interface GigabitEthernet2/0/0
#
bfd for_ip_frr bind peer-ip 10.1.1.2 vpn-instance vpn1 interface GigabitEthernet 1/0/0
 discriminator local 10
 discriminator remote 20
 commit
#
return
```

- CE1 的配置文件

```
#
 sysname CE1
#
bfd
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.2 255.255.255.252
#
interface GigabitEthernet2/0/0
 ip address 10.3.1.1 255.255.255.252
#
ospf 1
 area 0.0.0.0
  network 10.1.1.0 0.0.0.3
  network 10.3.1.0 0.0.0.3
#
bfd for_ip_frr bind peer-ip 10.1.1.1 interface GigabitEthernet 1/0/0
 discriminator local 20
 discriminator remote 10
 commit
#
return
```

- CE2 的配置文件

```
#
 sysname CE2
#
interface GigabitEthernet1/0/0
 ip address 10.2.1.2 255.255.255.252
#
interface GigabitEthernet2/0/0
 ip address 10.4.1.1 255.255.255.252
#
ospf 1
 area 0.0.0.0
  network 10.2.1.0 0.0.0.3
  network 10.4.1.0 0.0.0.3
#
return
```

- RTA 的配置文件

```
#
 sysname RTA
#
interface GigabitEthernet1/0/0
 ip address 10.3.1.2 255.255.255.252
#
interface GigabitEthernet2/0/0
 ip address 10.4.1.2 255.255.255.252
```

```
 ospf cost 100
#
interface GigabitEthernet3/0/0
 ip address 10.5.1.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 10.3.1.0 0.0.0.3
  network 10.4.1.0 0.0.0.3
area 0.0.0.2
  network 10.5.1.0 0.0.0.255
#
return
```

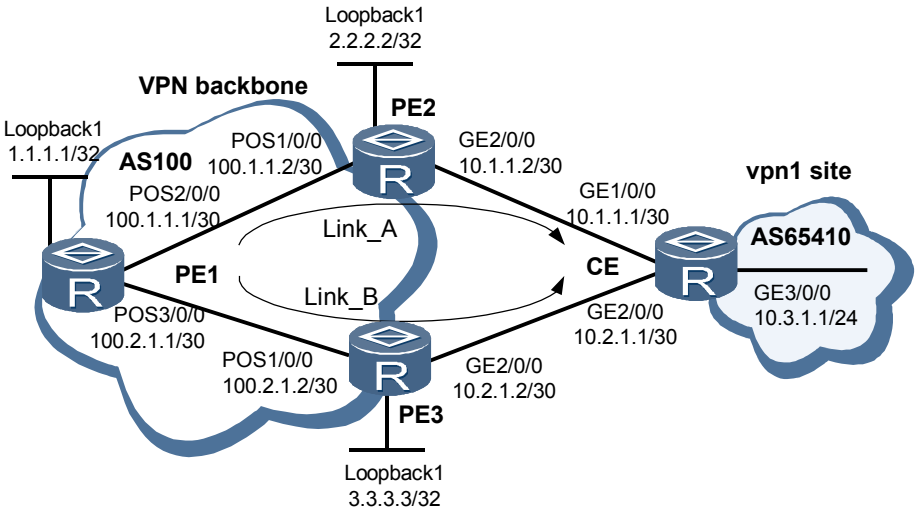
2.18.12 配置 VPN FRR 示例

在 CE 多归属环境中，配置 VPN FRR 可以保证 PE 设备发生故障时实现 VPN 业务端到端的快速切换。

组网需求

如图 2-13 所示，要求在 PE1 上配置备份下一跳，使 PE3 为 PE2 的备份，PE2 出现故障时可以快速切换到 PE3 上。

图 2-13 VPN FRR 示例组网图



配置思路

采用如下的思路配置 VPN FRR 功能。

1. 在 MPLS 骨干网上（PE1、PE2 和 PE3）配置 OSPF，实现骨干网互通。
2. 在 MPLS 骨干网上配置 MPLS 基本能力，使能 MPLS LDP，建立 LDP LSP。
3. 分别在各 PE 设备（PE1、PE2 和 PE3）上配置 VPN 实例，将 CE 接入 PE2 和 PE3。
4. 在各 PE 与 CE 之间建立 EBGP 对等体，引入 VPN 路由；在各 PE 之间建立 MP-IBGP 对等体。
5. 在 PE1 上配置 VPN FRR 路由策略，配置备份下一跳，使能 VPN FRR。

6. PE1 和 PE2 上配置 BFD 多跳检测。

数据准备

为完成此配置例，需准备如下的数据：

- PE 设备所在的 AS 号 100，CE 设备所在的 AS 号 65410。
- PE 设备上配置的 VPN 实例名称。

  说明

  为 VPN 实例配置 RD 时，如果 PE2 和 PE3 的 RD 相同，且与 PE1 的 RD 不同，则在 PE1 上不能形成 VPN FRR。
- 在 PE1 上配置的路由策略名称以及 ip-prefix 名称。
- BFD 配置名和会话参数。

操作步骤

步骤 1 配置 VPN 骨干网和 VPN site 中各接口的 IP 地址（略）

步骤 2 在 MPLS 骨干网上配置 OSPF 协议，实现骨干网 PE 互通（略）

步骤 3 在 MPLS 骨干网上配置 MPLS 基本能力和 MPLS LDP，建立 LDP LSP

配置 PE1。

```
<PE1> system-view
[PE1] mpls lsr-id 1.1.1.1
[PE1] mpls
[PE1-mpls] quit
[PE1] mpls ldp
[PE1-mpls-ldp] quit
[PE1] interface pos2/0/0
[PE1-Pos2/0/0] mpls
[PE1-Pos2/0/0] mpls ldp
[PE1-Pos2/0/0] quit
[PE1] interface pos3/0/0
[PE1-Pos3/0/0] mpls
[PE1-Pos3/0/0] mpls ldp
[PE1-Pos3/0/0] quit
```

配置 PE2。

```
<PE2> system-view
[PE2] mpls lsr-id 2.2.2.2
[PE2] mpls
[PE2-mpls] quit
[PE2] mpls ldp
[PE2-mpls-ldp] quit
[PE2] interface pos1/0/0
[PE2-Pos1/0/0] mpls
[PE2-Pos1/0/0] mpls ldp
[PE2-Pos1/0/0] quit
```

配置 PE3。

```
<PE3> system-view
[PE3] mpls lsr-id 3.3.3.3
[PE3] mpls
[PE3-mpls] quit
[PE3] mpls ldp
[PE3-mpls-ldp] quit
[PE3] interface pos1/0/0
[PE3-Pos1/0/0] mpls
```

```
[PE3-Pos1/0/0] mpls ldp
[PE3-Pos1/0/0] quit
```

此时在 PE 上执行命令 **display mpls lsp**，可看到 PE1 与 PE2、PE1 与 PE3 之间的 LSP 建立成功。以 PE1 的显示为例：

```
[PE1] display mpls lsp
------------------------------------------------------------------------------
                LSP Information: LDP LSP
------------------------------------------------------------------------------
FEC                 In/Out Label  In/Out IF                      Vrf Name
1.1.1.1/32          3/NULL        -/-
2.2.2.2/32          NULL/3        -/P2/0/0
2.2.2.2/32          1025/3        -/P2/0/0
3.3.3.3/32          NULL/3        -/P3/0/0
3.3.3.3/32          1024/3        -/P3/0/0
```

步骤 4 在 PE 设备上配置 VPN 实例，将 CE 接入 PE2 和 PE3

配置 PE1。

```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] ipv4-family
[PE1-vpn-instance-vpn1-af-ipv4] route-distinguisher 100:1
[PE1-vpn-instance-vpn1-af-ipv4] vpn-target 111:1
[PE1-vpn-instance-vpn1-af-ipv4] quit
[PE1-vpn-instance-vpn1] quit
```

配置 PE2。

```
[PE2] ip vpn-instance vpn1
[PE2-vpn-instance-vpn1] ipv4-family
[PE2-vpn-instance-vpn1-af-ipv4] route-distinguisher 100:2
[PE2-vpn-instance-vpn1-af-ipv4] vpn-target 111:1
[PE2-vpn-instance-vpn1-af-ipv4] quit
[PE2-vpn-instance-vpn1] quit
[PE2] interface gigabitethernet2/0/0
[PE2-GigabitEthernet2/0/0] ip binding vpn-instance vpn1
[PE2-GigabitEthernet2/0/0] ip address 10.1.1.2 30
[PE2-GigabitEthernet2/0/0] quit
```

配置 PE3。

```
[PE3] ip vpn-instance vpn1
[PE3-vpn-instance-vpn1] ipv4-family
[PE3-vpn-instance-vpn1-af-ipv4] route-distinguisher 100:3
[PE3-vpn-instance-vpn1-af-ipv4] vpn-target 111:1
[PE3-vpn-instance-vpn1-af-ipv4] quit
[PE3-vpn-instance-vpn1] quit
[PE3] interface gigabitethernet2/0/0
[PE3-GigabitEthernet2/0/0] ip binding vpn-instance vpn1
[PE3-GigabitEthernet2/0/0] ip address 10.2.1.2 30
[PE3-GigabitEthernet2/0/0] quit
```

步骤 5 在 PE1 上引入直连 VPN 路由；在 PE2 与 CE，及 PE3 与 CE 之间建立 EBGP 对等体，引入 VPN 路由

配置 PE1。

```
[PE1] bgp 100
[PE1-bgp] ipv4-family vpn-instance vpn1
[PE1-bgp-vpn1] import-route direct
[PE1-bgp-vpn1] quit
```

配置 PE2。

```
[PE2] bgp 100
[PE2-bgp] ipv4-family vpn-instance vpn1
[PE2-bgp-vpn1] peer 10.1.1.1 as-number 65410
```

```
[PE2-bgp-vpn1] quit
```

配置 PE3。

```
[PE3] bgp 100
[PE3-bgp] ipv4-family vpn-instance vpn1
[PE3-bgp-vpn1] peer 10.2.1.1 as-number 65410
[PE3-bgp-vpn1] quit
```

配置 CE。

```
<CE> system-view
[CE] bgp 65410
[CE-bgp] peer 10.1.1.2 as-number 100
[CE-bgp] peer 10.2.1.2 as-number 100
[CE-bgp] import-route direct
[CE-bgp] network 10.3.1.0 24
[CE-bgp] quit
```

完成此步骤后，在 PE2 和 PE3 上执行 **display bgp vpnv4 all peer** 命令，可看到 PE 和 CE 之间的 EBGP 对等体建立成功，其状态为“Established”。

以 PE2 的显示为例：

```
[PE2] display bgp vpnv4 all 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
  Peer of IPv4-family for vpn instance :

 VPN-Instance vpn1, router ID 2.2.2.2:
  10.1.1.1        4       65410       46       46     0 00:37:41 Established       5
```

步骤 6 在 PE 之间建立 MP-IBGP 对等体

配置 PE1。

```
[PE1] bgp 100
[PE1-bgp] peer 2.2.2.2 as-number 100
[PE1-bgp] peer 2.2.2.2 connect-interface loopback 1
[PE1-bgp] peer 3.3.3.3 as-number 100
[PE1-bgp] peer 3.3.3.3 connect-interface loopback 1
[PE1-bgp] ipv4-family vpnv4
[PE1-bgp-af-vpnv4] peer 2.2.2.2 enable
[PE1-bgp-af-vpnv4] peer 3.3.3.3 enable
[PE1-bgp-af-vpnv4] quit
```

配置 PE2。

```
[PE2] bgp 100
[PE2-bgp] peer 1.1.1.1 as-number 100
[PE2-bgp] peer 1.1.1.1 connect-interface loopback 1
[PE2-bgp] ipv4-family vpnv4
[PE2-bgp-af-vpnv4] peer 1.1.1.1 enable
[PE2-bgp-af-vpnv4] quit
```

配置 PE3。

```
[PE3] bgp 100
[PE3-bgp] peer 1.1.1.1 as-number 100
[PE3-bgp] peer 1.1.1.1 connect-interface loopback 1
[PE3-bgp] ipv4-family vpnv4
[PE3-bgp-af-vpnv4] peer 1.1.1.1 enable
[PE3-bgp-af-vpnv4] quit
```

完成此步骤后，在 PE 上执行 **display bgp vpnv4 all peer** 命令，可看到 MP-IBGP 对等体建立成功，其状态为“Established”。

以 PE1 的显示为例：

```
[PE1] display bgp vpnv4 all peer
 BGP local router ID : 1.1.1.1
 Local AS number : 100
 Total number of peers : 2                 Peers in established state : 2
Peer            V    AS  MsgRcvd  MsgSent    OutQ  Up/Down       State PrefRcv
2.2.2.2         4   100       20       17       0 00:13:26 Established       5
3.3.3.3         4   100       24       19       0 00:17:18 Established       5
```

步骤 7 配置 VPN FRR 路由策略

```
[PE1] ip ip-prefix vpn_frr_list permit 2.2.2.2 32
[PE1] route-policy vpn_frr_rp permit node 10
[PE1-route-policy] if-match ip next-hop ip-prefix vpn_frr_list
[PE1-route-policy] apply backup-nexthop 3.3.3.3
[PE1-route-policy] quit
```

步骤 8 配置 BFD 多跳检测

在 PE1 上配置 BFD 多跳检测

```
[PE1] bfd
[PE1-bfd] quit
[PE1] bfd for_ip_frr bind peer-ip 2.2.2.2
[PE1-bfd-session-for_ip_frr] discriminator local 10
[PE1-bfd-session-for_ip_frr] discriminator remote 20
[PE1-bfd-session-for_ip_frr] commit
```

在 PE2 上配置 BFD 多跳检测

```
[PE2] bfd
[PE2-bfd] quit
[PE2] bfd for_ip_frr bind peer-ip 1.1.1.1
[PE2-bfd-session-for_ip_frr] discriminator local 20
[PE2-bfd-session-for_ip_frr] discriminator remote 10
[PE2-bfd-session-for_ip_frr] commit
```

配置完成后，在 PE1 和 PE2 上执行 **display bfd session all verbose** 命令，可以看到建立了一个多跳（Multi Hop）的 BFD Session，且状态为 Up。

步骤 9 使能 VPN FRR

```
[PE1] ip vpn-instance vpn1
[PE1-vpn-instance-vpn1] ipv4-family
[PE1-vpn-instance-vpn1-af-ipv4] vpn frr route-policy vpn_frr_rp
[PE1-vpn-instance-vpn1-af-ipv4] quit
[PE1-vpn-instance-vpn1] quit
```

查看备份下一跳、备份标签和备份 Tunnel ID 的信息。

```
<PE1> display ip routing-table vpn-instance vpn1 10.3.1.0 verbose
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Table : vpn1
Summary Count : 1
Destination: 10.3.1.0/24
     Protocol: BGP             Process ID: 0
   Preference: 255                   Cost: 0
NextHop: 2.2.2.2            Neighbour: 2.2.2.2
        State: Active Adv Relied      Age: 00h15m06s
          Tag: 0                Priority: low
        Label: 15361             QoSInfo: 0x0
   IndirectID: 0x13
 RelayNextHop: 0.0.0.0          Interface: Pos2/0/0
     TunnelID: 0x6002002            Flags: RD
  BkNextHop: 3.3.3.3        BkInterface:Unknown
    BkLabel: 15362           SecTunnelID: 0x0
 BkPETunnelID: 0x6002000  BkPESecTunnelID: 0x0
 BkIndirectID: 0x15
```

说明

本例中，PE2 和 PE3 都会向 PE1 发布 10.3.1.0/24 这条路由，而且路由的 BGP 属性都一样，只是由于 PE2 的 Router ID 较小，所以 PE1 优选了 PE2 发布的路由，也就是链路 Link_A。如果在实际应用中 PE2 的 Router ID 大于 PE3 的 Router ID，那么 PE1 会选择 Link_B 作为主链路，这时如果依旧按照上面的配置，会导致 VPN FRR 失效。为了防止这种情况出现，可以在 BGP-VPNv4 地址族下修改 VPNv4 路由的属性，保证 PE1 优选链路 Link_A。修改 VPNv4 路由的属性的方法较多，以下是常用的两种。

- 可以在 PE1 的 BGP-VPNv4 地址族下为 PE2 发来的路由设置较高的协议首选值（PrefVal），相关配置如下:

```
route-policy policy1 permit node 10
 apply preferred-value 100
#
bgp 100
 #
 ipv4-family vpnv4
  peer 2.2.2.2 route-policy policy1 import
```

- 也可以在 PE2 的 BGP-VPNv4 地址族下为发布的路由设置较高的本地优先级（Local_Pref），相关配置如下:

```
route-policy policy2 permit node 10
 apply local-preference 200
#
bgp 100
 #
 ipv4-family vpnv4
  peer 1.1.1.1 route-policy policy2 export
```

----结束

配置文件

- PE1 的配置文件

```
#
 sysname PE1
#
 bfd
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:1
  vpn frr route-policy vpn_frr_rp
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
 mpls lsr-id 1.1.1.1
 mpls
#
mpls ldp
#
interface Pos2/0/0
 link-protocol ppp
 ip address 100.1.1.1 255.255.255.252
 mpls
 mpls ldp
#
interface Pos3/0/0
 link-protocol ppp
 ip address 100.2.1.1 255.255.255.252
 mpls
 mpls ldp
#
interface LoopBack1
 ip address 1.1.1.1 255.255.255.255
#
bfd for_ip_frr bind peer-ip 2.2.2.2
 discriminator local 10
```

```
 discriminator remote 20
 commit
#
bgp 100
 peer 2.2.2.2 as-number 100
 peer 2.2.2.2 connect-interface LoopBack1
 peer 3.3.3.3 as-number 100
 peer 3.3.3.3 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
#
 ipv4-family vpnv4
  policy vpn-target
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
#
 ipv4-family vpn-instance vpn1
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 100.1.1.0 0.0.0.3
  network 100.2.1.0 0.0.0.3
  network 1.1.1.1 0.0.0.0
#
ip ip-prefix vpn_frr_list index 10 permit 2.2.2.2 32
#
route-policy vpn_frr_rp permit node 10
 if-match ip next-hop ip-prefix vpn_frr_list
 apply backup-nexthop 3.3.3.3
#
return
```

- PE2 的配置文件

```
#
 sysname PE2
#
 bfd
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:2
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
 mpls lsr-id 2.2.2.2
 mpls
#
mpls ldp
#
interface Pos1/0/0
 link-protocol ppp
 ip address 100.1.1.2 255.255.255.252
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.1.1.2 255.255.255.252
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
#
bfd for_ip_frr bind peer-ip 1.1.1.1
 discriminator local 20
 discriminator remote 10
 commit
```

```
#
bgp 100
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.1 enable
#
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.1 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.1.1.1 as-number 65410
  import-route direct
#
ospf 1
 area 0.0.0.0
  network 100.1.1.0 0.0.0.3
  network 2.2.2.2 0.0.0.0
#
return
```

- PE3 的配置文件

```
#
 sysname PE3
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 100:3
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
 mpls lsr-id 3.3.3.3
 mpls
#
mpls ldp
#
interface Pos1/0/0
 link-protocol ppp
 ip address 100.2.1.2 255.255.255.252
 mpls
 mpls ldp
#
interface GigabitEthernet2/0/0
 ip binding vpn-instance vpn1
 ip address 10.2.1.2 255.255.255.252
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.255
#
bgp 100
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack1
 #
 ipv4-family unicast
  undo synchronization
  peer 1.1.1.1 enable
#
 ipv4-family vpnv4
  policy vpn-target
  peer 1.1.1.1 enable
#
 ipv4-family vpn-instance vpn1
  peer 10.2.1.1 as-number 65410
  import-route direct
#
ospf 1
 area 0.0.0.0
```

```
  network 100.2.1.0 0.0.0.3
  network 3.3.3.3 0.0.0.0
#
Return
```

- CE 的配置文件

```
#
 sysname CE
#
interface GigabitEthernet1/0/0
 ip address 10.1.1.1 255.255.255.252
#
interface GigabitEthernet2/0/0
 ip address 10.2.1.1 255.255.255.252
#
interface GigabitEthernet3/0/0
 ip address 10.3.1.1 255.255.255.0
#
bgp 65410
 peer 10.1.1.2 as-number 100
 peer 10.2.1.2 as-number 100
 #
 ipv4-family unicast
  undo synchronization
  network 10.3.1.0 255.255.255.0
  import-route direct
  peer 10.1.1.2 enable
  peer 10.2.1.2 enable
#
return
```

3 L2TP 协议配置

关于本章

L2TP 是一种 VPN 技术，提供了对 PPP 链路层数据包的隧道传输支持，并允许二层链路端点和 PPP 会话点驻留在不同设备上。

3.1 L2TP 协议简介
L2TP 协议结合了 L2F 协议和 PPTP 协议的优点，成为 IETF 有关二层隧道协议的工业标准。

3.2 配置 L2TP 基本能力
在 L2TP 的各项配置中，只有配置了 L2TP 的基本能力，才可进行其它 L2TP 相关功能特性的配置。

3.3 配置 LAC 侧
通过配置 LAC 侧，设备可以鉴定用户是否为接入用户并向 LNS 发起连接。

3.4 配置 LNS 侧
LNS 接收到 LAC 发来的创建隧道请求，并决定对用户的认证方式和是否允许对端创建 L2TP 隧道。

3.5 配置 L2TP 连接参数
创建 L2TP 连接后，可以对 L2TP 的相关参数进行配置或调整。

3.6 维护 L2TP
如何强制挂断通道、监控 L2TP 协议运行状况。

3.7 配置举例
介绍 L2TP 的配置举例。

3.1 L2TP 协议简介

L2TP 协议结合了 L2F 协议和 PPTP 协议的优点，成为 IETF 有关二层隧道协议的工业标准。

3.1.1 L2TP 协议概述

L2TP 连接的维护以及 PPP 数据的传送都是通过 L2TP 消息的交换来完成的。这些消息通过 UDP 的 1701 端口承载于 TCP/IP 之上，L2TP 消息可以分为两种类型，一种是控制消息，另一种是数据消息。

PPP 协议定义了一种封装技术，可以在二层点到点链路上传输多种协议数据包，这时，用户与 NAS 之间运行 PPP，二层链路端点与 PPP 会话点在相同硬件设备上。

L2TP 协议提供了对 PPP 链路层数据包的隧道（Tunnel）传输支持，允许二层链路端点和 PPP 会话点驻留在不同设备上，并采用包交换技术进行信息交互，从而扩展了 PPP 模型。L2TP 协议结合了 L2F 协议和 PPTP 协议的优点，成为 IETF 有关二层隧道协议的工业标准。

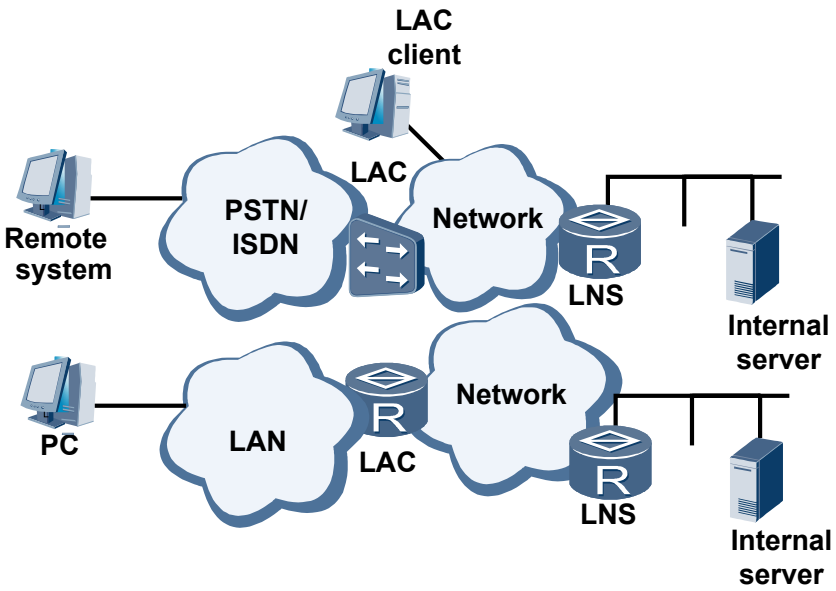
3.1.2 AR1200 支持的 L2TP 协议特性

AR1200 中支持的 L2TP 特性支持三种典型的 L2TP 隧道模式。

三种典型的 L2TP 隧道模式

远端系统或 LAC 客户端（运行 L2TP 协议的主机）与 LNS 之间的隧道模式如图 3-1 所示：

图 3-1 三种典型的 L2TP 隧道模式示意图



有三种方式可以建立连接：

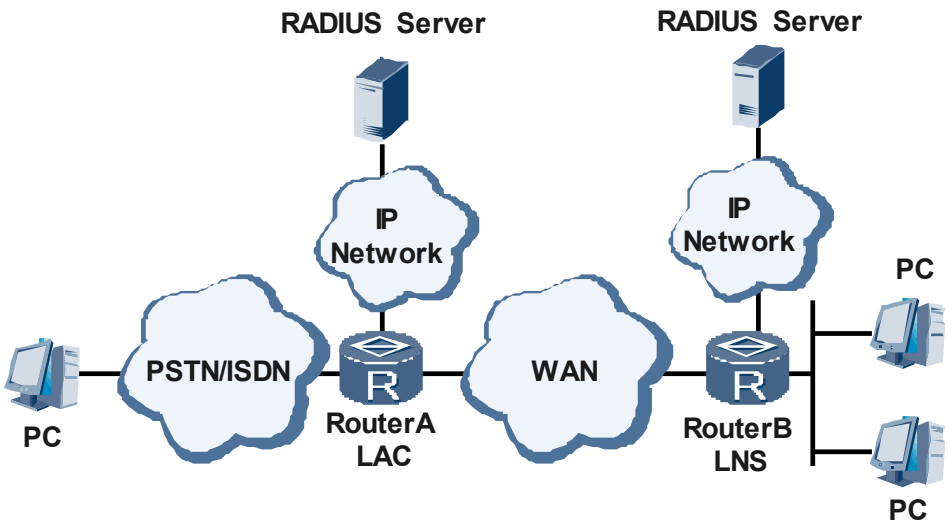
- NAS-Initialized：由远程拨号用户发起，远程系统通过 PSTN/ISDN 拨入 LAC，由 LAC 通过 Internet 向 LNS 发起建立隧道连接请求。拨号用户地址由 LNS 分配；对远程拨号用户的验证与计费既可由 LAC 侧的代理完成，也可在 LNS 完成。
- Client-Initialized：直接由 LAC 客户（指可在本地支持 L2TP 协议的用户）发起。此时 LAC 客户可直接向 LNS 发起隧道连接请求，无需再经过一个单独的 LAC 设备。此时，LAC 客户地址的分配由 LNS 来完成。
- LAC-Auto-Initiated：通常情况下，L2TP 的客户端是拨号连接到 LAC 的用户主机。此时用户与 LAC 的连接总是 PPP 连接。如果使用 LAC 同时作为客户端，用户与 LAC 之间的连接就不受限于 PPP 连接。用户可以直接通过 IP 连接，LAC 也能够将用户的 IP 报文转发到 LNS。使用 LAC 同时作为客户端，需要在 LAC 上建立一个虚拟的 PPP 用户，同时创建一个与之对应的虚拟 PPP Server，该虚拟用户首先和虚拟 Server 进行 PPP 协商，虚拟 Server 再通过建立 L2TP 隧道将 PPP 协商延续至 LNS。

AR1200 支持同时作为 LAC 及 LNS，并支持同时有多路用户呼入；只要内存及线路不受限制，L2TP 可以同时接收和发起多个呼叫。

L2TP 隧道会话的建立过程

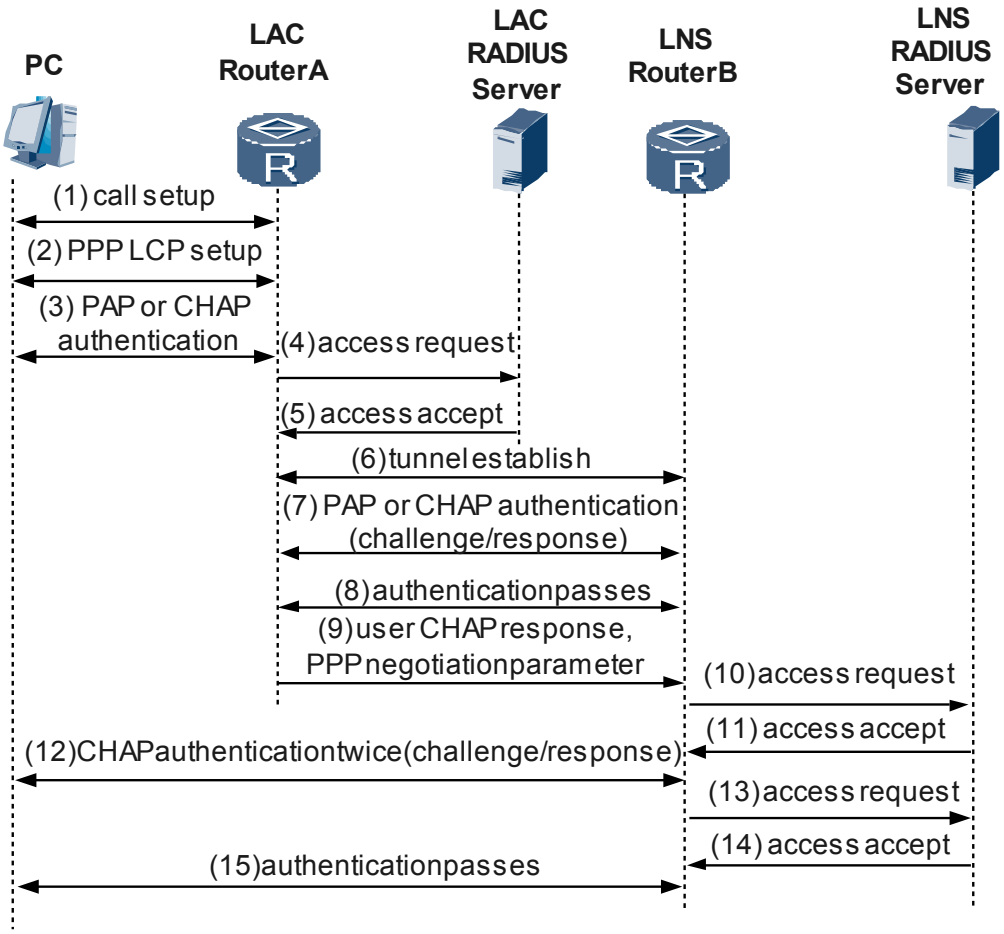
L2TP 的典型组网如图 3-2 所示：

图 3-2 L2TP 隧道的典型组网示意图



L2TP 隧道的呼叫建立流程如图 3-3 所示：

图 3-3 L2TP 隧道的呼叫建立流程



1. 用户端 PC 机发起呼叫连接请求。
2. PC 机和 LAC 端（RouterA）进行 PPP LCP 协商。
3. LAC 对 PC 机提供的用户信息进行 PAP 或 CHAP 认证。
4. LAC 将认证信息（用户名、密码）发送给 RADIUS 服务器进行认证。
5. RADIUS 服务器认证该用户，如果认证通过则返回该用户对应的 LNS 地址等相关信息，并且 LAC 准备发起 Tunnel 连接请求。
6. LAC 端向指定 LNS 发起 Tunnel 连接请求。
7. LAC 端向指定 LNS 发送 CHAP challenge 信息，LNS 回送该 challenge 响应消息 CHAP response，并发送 LNS 侧的 CHAP challenge，LAC 返回该 challenge 的响应消息 CHAP response。
8. 隧道验证通过。
9. LAC 端将用户 CHAP response、response identifier 和 PPP 协商参数传送给 LNS。
10. LNS 将接入请求信息发送给 RADIUS 服务器进行认证。
11. RADIUS 服务器认证该请求信息，如果认证通过则返回响应信息。
12. 若用户在 LNS 侧配置强制本端 CHAP 认证，则 LNS 对用户进行认证，发送 CHAP challenge，用户侧回应 CHAP response。

13. LNS 再次将接入请求信息发送给 RADIUS 服务器进行认证。
14. RADIUS 服务器认证该请求信息，如果认证通过则返回响应信息。
15. 验证通过，用户访问企业内部资源。

3.2 配置 L2TP 基本能力

在 L2TP 的各项配置中，只有配置了 L2TP 的基本能力，才可进行其它 L2TP 相关功能特性的配置。

3.2.1 建立配置任务

在配置 L2TP 基本能力前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

在 L2TP 的各项配置中，必须先启动 L2TP、创建 L2TP 组，然后再进行其它功能特性的配置。根据路由器是作为 LAC 侧还是作为 LNS 侧，在一些具体的配置选择上也有所不同。

L2TP 组是 L2TP 配置中的一个重要概念，创建 L2TP 组后，不仅可以在路由器上灵活地配置 L2TP 各项功能，而且能够方便地实现 LAC 和 LNS 之间一对一、一对多的组网应用。

前置任务

无

数据准备

在配置 L2TP 基本能力之前，需准备以下数据。

序号	数据
1	L2TP 组的组号
2	LAC 侧和 LNS 侧的隧道名

3.2.2 配置 L2TP 基本能力

在 L2TP 的各项配置中，必须先启动 L2TP、创建 L2TP 组，然后再进行其它功能特性的配置。根据设备是作为 LAC 侧还是作为 LNS 侧，在一些具体的配置选择上也有所不同。

背景信息

L2TP 组在 LAC 和 LNS 上独立编号，只需要保证 LAC 和 LNS 之间关联的 L2TP 组的相关配置保持对应关系即可，如接收的通道对端名称、发起 L2TP 连接请求及 LNS 地址等。

创建 L2TP 组后，就可以在 L2TP 组视图下进行和该 L2TP 组相关的其它配置了，如本端名称、发起 L2TP 连接请求及 LNS 地址。

在 LAC 侧和 LNS 侧进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **l2tp enable**，使能 L2TP。

只有使能 L2TP 后，L2TP 功能才能使用；如果禁止 L2TP，则即便配置了 L2TP 的参数，路由器也不会提供相关功能。

缺省情况下，L2TP 功能是被禁止的，也没有任何 L2TP 组。

步骤 3 执行命令 **l2tp-group** *group-number*，创建 L2TP 组，并进入 L2TP 组视图。

组号为 1 表示缺省的 L2TP 组。为了接收这种不知名对端发起的通道请求连接，或者用于测试目的，可以设置一个缺省的 L2TP 组。

步骤 4 执行命令 **tunnel name** *tunnel-name*，设置本端隧道名称。

用户可在 LAC 侧或 LNS 侧配置本端隧道名称。缺省情况下，本端隧道名称为路由器的主机名。

说明

LAC 侧隧道名称要与 LNS 侧配置的接收隧道对端名称保持一致。

----结束

3.3 配置 LAC 侧

通过配置 LAC 侧，设备可以鉴定用户是否为接入用户并向 LNS 发起连接。

3.3.1 建立配置任务

在配 LAC 侧前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

路由器不会主动与其它设备或 LNS 服务器建立 L2TP 隧道，需要满足一定的条件后才会发出建立 L2TP 连接的请求。

通过配置对接入用户信息的判别条件，并指定相应的 LNS 端的 IP 地址，路由器可以鉴定用户是否为接入用户并向 LNS 发起连接。

发起 L2TP 连接请求的触发条件有两种：完整的用户名（full-user-name）、带特定域名的用户（domain）。必须配置一种触发条件，方可发出 L2TP 连接的请求。

LAC 端在发起隧道建立请求时，需要将本端隧道的源地址发给 LNS，用于 LAC 与 LNS 的通信。

前置任务

在配置 LAC 侧之前，需完成以下任务：

- 配置 L2TP 基本能力。

数据准备

在配置 LAC 侧之前，需准备以下数据。

序号	数据
1	L2TP 组的组号
2	LNS 的 IP 地址
3	作为 L2TP 连接触发条件的用户全名或域名
4	发起隧道建立请求时使用的接口类型和编号
5	用于认证的用户名和密码
6	如果使用 RADIUS 认证，需准备认证方案名称、RADIUS 模板名称、RADIUS 服务器 IP 地址和端口、密钥、重传次数

3.3.2 配置 LAC 侧的 L2TP 连接

LAC 在收到 LAC 客户端发起的呼叫后，LAC 按照配置的 LNS 的先后顺序向各 LNS 发送建立 L2TP 隧道的连接请求，当得到 LNS 的接收应答后，该 LNS 就作为隧道的对端；否则 LAC 向下一个 LNS 发起隧道连接请求。

背景信息

在路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **l2tp-group** *group-number*，进入 L2TP 组视图。

步骤 3 指定本端作为 L2TP LAC 端时发起呼叫的触发条件，选择如下方法之一：

- 如果用户使用域名接入，执行命令 **start l2tp ip** *ip-address* **domain** *domain-name*，指定触发条件为域名。
- 如果用户使用全名接入，执行命令 **start l2tp ip** *ip-address* **fullusername** *user-name*，指定触发条件为用户全名。

----结束

3.3.3 （可选）配置 LAC 自拨号功能

路由器发起 VPDN 拨号，路由器既作为 PPP 客户端，同时又实现 LAC 的功能。

背景信息

很多企业分支、总部间希望能够构建 VPN 网络，但向网络运营商申请的 VPN 网络存在较高的费用。通过 VPDN 技术，企业可以自己构建 VPN 网络，运营商只需要提供基础的 Internet 接入功能即可，这样可减少企业的费用。路由器自己拨号建立 VPDN 网络，分支路由器既做 PPP Client，又做 LAC。

在路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface virtual-template** *vt-number*，创建并进入虚拟接口模板。

步骤 3 配置虚拟接口的 IP 地址，可以通过三种方式配置，可任意选择其中一种方式进行配置：

- 执行命令 **ip address** *ip-address* { *mask* | *mask-length* }，直接配置接口的 IP 地址。
- 执行命令 **ip address ppp-negotiate**，通过接受 PPP 协商产生的由对端分配的 IP 地址。
- 执行命令 **ip address unnumbered interface** *interface-type interface-number*，配置接口借用其他接口的 IP 地址。

步骤 4 配置本地设备被对端设备验证的用户名及密码，选择如下方法之一：

- 配置采用 PAP 方式验证时发送的用户名和口令：**ppp pap local-user** *username* **password** { **cipher** | **simple** } *password*
- 配置采用 CHAP 方式验证时发送的用户名和口令：
  1. 执行命令 **ppp chap user** *username*，配置 CHAP 认证的用户名。
  2. 执行命令 **ppp chap password** { **cipher** | **simple** } *password*，配置 CHAP 认证的密码。

在配置用户名和口令时，要和对端的设置的用户名和口令一致。缺省情况下，被对端以 PAP 方式验证时，本地设备发送的用户名和口令均为空。

步骤 5 执行命令 **l2tp-auto-client enable**，使能 LAC 自拨号功能。

步骤 6 执行命令 **quit**，退回系统视图。

步骤 7 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } **virtual-template** *vt-number*,配置静态路由。

需要在 LAC 上配置路由，目的地址段是总部，出接口是虚拟 PPP 用户接口，这样报文会才会发给 PPP 接口，并通过已经建立的 L2TP 隧道发送给 LNS。

----结束

3.3.4 （可选）配置 LAC 侧使用本地认证

LAC 对用户提供的认证信息进行本地认证。LAC 上需要包含该用户名和对应的密码，选用密文还是用明文口令，需要看用户的需求，选用密文的安全度更高。

背景信息

说明

关于认证，授权和计费的详细信息，请参见 Huawei AR1200 系列企业路由器 *配置指南 - 安全*.

在路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **aaa**，进入 AAA 视图。

步骤 3 设置用户名及密码，执行命令 **local-user** *user-name* **password** *password*。

LAC 检查远程拨入用户名与口令是否与本地注册用户名与口令相符，以确认用户是否合法。LAC 侧验证通过后才能发起建立隧道连接的请求。

缺省情况下，LAC 侧未配置本地用户名和口令。缺省的认证方法是本地认证，LAC 侧必须配置本地认证的用户名和密码。

----结束

3.3.5 （可选）配置 LAC 侧使用 RADIUS 认证

LAC 将户提供的认证信息（用户名、密码）发送给 RADIUS 服务器进行认证。RADIUS 协议的主要目的是为了大量的分散用户进行集中管理，对用户进行认证，分配权限以及进行计费。

操作步骤

- 创建 AAA 认证和计费方案

  在路由器上进行如下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **aaa**，进入 AAA 视图。
  3. 执行命令 **authentication-scheme** *authentication-scheme-name*，创建一个认证方案，并进入认证方案视图。
  4. 执行命令 **authentication-mode radius**，指定认证模式为 RADIUS 认证。
  5. 执行命令 **quit**，退回 AAA 视图。
  6. 执行命令 **accounting-scheme** *accounting-scheme-name*，创建一个计费方案，并进入计费方案视图。

     说明

     RADIUS 计费是可选配置。
  7. 执行命令 **accounting-mode radius**，配置使用 RADIUS 服务器计费。
- 配置 RADIUS 模板及其参数。

  在路由器上进行如下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **radius-server template** *template-name*，创建一个 RADIUS 服务器模板。
  3. 执行命令 **radius-server authentication** *ip-address port*，配置 RADIUS 认证服务器 IP 地址和端口。
  4. 执行命令 **radius-server accounting** *ip-address port*，配置 RADIUS 计费服务器 IP 地址和端口。

5. 执行命令 **radius-server shared-key** { **cipher** | **simple** } *key-string*，配置 RADIUS 服务器密钥。
6. 执行命令 **radius-server retransmit** *retry-times*，配置 RADIUS 服务器重传次数。

- 配置域，应用 RADIUS 模板及认证和计费方案

  在路由器上进行如下配置。

  1. 执行命令 **system-view**，进入系统视图。
  2. 执行命令 **aaa**，进入 AAA 视图。
  3. 执行命令 **domain** *domain-name*，创建域，并进入域视图。
  4. 执行命令 **radius-server** *template-name*，应用 RADIUS 模板。
  5. 执行命令 **authentication-scheme** *authentication-scheme-name*，应用认证方案。
  6. 执行命令 **accounting-scheme** *accounting-scheme-name*，应用计费方案。

  说明

  在 RADIUS 服务器上需要设置用户名和口令（与用户端的一致），并设置 NAS 侧设备接入 RADIUS 的 IP 地址、共享密钥、认证（和计费）的端口号。

----结束

3.3.6 检查配置结果

LAC 侧配置成功后，您可以查看到 L2TP 的通道信息和 L2TP 的会话信息。

前提条件

已经完成 LAC 侧功能的所有配置。

操作步骤

- 使用 **display l2tp tunnel** 命令查看当前的 L2TP 通道的信息。
- 使用 **display l2tp session** 命令查看当前的 L2TP 会话的信息。
- 使用 **display l2tp-group** [*group-number*]命令查看指定 L2TP 组的配置信息。

----结束

任务示例

在 LAC 侧和 LNS 侧都配置成功且连接保持时，在 LAC 侧执行命令 **display l2tp tunnel**，可以看到 L2TP 控制连接建立成功。例如：

```
<Huawei> display l2tp tunnel

Total tunnel = 1
 LocalTID RemoteTID RemoteAddress    Port   Sessions RemoteName
 1        1         202.38.160.1     57344  1        LAC
```

执行 **display l2tp session** 命令，可查看 L2TP 会话连接已建立。例如：

```
<Huawei> display l2tp session

 Total session = 1
 LocalSID  RemoteSID  LocalTID
  2036      1469       1
```

执行 **display l2tp-group** [*group-number*]命令，可查看指定 L2TP 组的配置信息。例如：

```
<Huawei> display l2tp-group 1
```

```
-----------------------------------------------
L2tp-index          :    1
GroupType           :    REQUEST_DIALIN_L2TP
TunnelAuth          :    Use tunnel authentication
LocalName           :    lac1
Encrypt             :   0
Hello               :   60
Retransmit          :   5
Timeout             :   2
IfIndex             :   4294967295
SrcIp               :   255.255.255.255
VtNum               :   0
RemoteName          :
ForceChap           :   0
LcpReg              :   0
LcpMismatch         :   0
tunnel each user    :   0
-----------------------------------------------
```

3.4 配置 LNS 侧

LNS 接收到 LAC 发来的创建隧道请求，并决定对用户的认证方式和是否允许对端创建 L2TP 隧道。

3.4.1 建立配置任务

在配置 LNS 侧前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

LNS 可以使用不同的虚拟接口模板接收不同 LAC 创建隧道的请求。接收到 LAC 发来的创建隧道请求后，LNS 检查 LAC 的名称是否与合法隧道对端名称相符，从而决定是否允许对端创建隧道。

LAC 对用户进行验证后，LNS 可再次对用户进行验证。即，对用户进行两次验证：第一次发生在 LAC 侧，第二次发生在 LNS 侧。

LNS 侧验证通过后可以进行接入用户和 LNS 的通信，否则通知 L2TP 清除此 L2TP 连接。因此，只有 LAC 侧和 LNS 侧的验证全部成功后，L2TP 通道才能建立。

LNS 侧对用户的验证方式有三种：代理验证、强制 CHAP 验证和 LCP 重协商。其中，LCP 重协商的优先级最高。

- LCP 重协商
  - LCP 重协商使用相应虚拟接口模板 VT 上配置的验证方式。
  - 对由 NAS 发起的 VPN 服务请求（NAS-Initialized VPN），在 PPP 会话开始时，用户先和 NAS 进行 PPP 协商。若协商通过，则由 NAS 初始化 L2TP 通道连接，并将用户信息传递给 LNS，由 LNS 根据收到的代理验证信息，判断用户是否合法。
  - 如果需要在 LNS 侧进行比 LAC 侧更严格的认证，或者 LNS 侧需要直接从用户获取某些信息（当 LNS 与 LAC 是不同厂商的设备时可能发生这种情况），则可以配置 LNS 与用户间进行 LCP 重协商，此时将忽略 NAS 侧的代理验证信息。
- 强制 CHAP 验证

  如果只配置强制 CHAP 验证，则 LNS 对用户进行 CHAP 验证。

- 代理验证

  如果既不配置 LCP 重协商，也不配置强制 CHAP 验证，则 LNS 对用户进行的是代理验证。在这种情况下，LAC 将它从用户得到的所有验证信息发送给 LNS，LNS 根据 LAC 发来的验证信息和本地配置情况对用户进行验证。

  当 LNS 使用代理验证时，如果虚拟接口模板 VT 配置的验证方式为 CHAP，而 LAC 端配置的验证方式为 PAP，则由于 LNS 要求的 CHAP 验证级别高于 LAC 能够提供的 PAP 验证，验证将无法通过，会话也就不能正确建立。

  说明

  只有一种情况 LNS 侧不对接入用户进行二次验证：启用 LCP 重协商后，不在相应的虚拟接口模板上配置验证。这时，用户只在 LAC 侧接受一次验证。

  其他情况都进行二次验证，验证模式（Authentication-mode）为“none”也算一种验证。

前置任务

在配置 LNS 侧之前，需完成以下任务：

- 配置 L2TP 基本能力。
- 创建用于建立 L2TP 连接的虚拟接口模板。

数据准备

在配置 LNS 侧之前，需准备以下数据。

序号	数据
1	L2TP 组的组号
2	虚拟接口模板编号
3	对端的隧道名称
4	本地用户名和密码

3.4.2 配置 LNS 侧的 L2TP 连接

LNS 接收到 LAC 发来的创建隧道请求后，LNS 检查 LAC 的名称是否与合法隧道对端名称相符，从而决定是否允许对端创建隧道。LNS 可以使用不同的虚拟接口模板接收不同 LAC 创建隧道的请求。

背景信息

在路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **l2tp-group** *group-number*，进入 L2TP 组视图。

步骤 3 设置通道对端的名称

- L2TP 组不为 1 时，执行命令 **allow l2tp virtual-template** *virtual-template-number* **remote** *remote-name*。
- L2TP 组为 1 时，执行命令 **allow l2tp virtual-template** *virtual-template-number* [**remote** *remote-name*]

当 L2TP 组号为 1 时（缺省的 L2TP 组号），可以不指定通道对端名 *remote-name*。如果在 L2TP 组 1 视图下指定对端名称，则 L2TP 组 1 不再作为缺省的 L2TP 组。

说明

只有组号为 1 的 L2TP 组才可以作为缺省组。

对于同一个 L2TP 组，**start** 命令和 **allow l2tp** 命令互斥，配置了一条命令之后另一条命令自动失效。

----结束

3.4.3 （可选）配置 LNS 侧的用户验证

LAC 对用户进行验证后，LNS 可再次对用户进行验证。验证方式有三种：代理验证、强制 CHAP 验证和 LCP 重协商。其中，LCP 重协商的优先级最高。

背景信息

说明

关于 AAA 的详细信息，请参见 Huawei AR1200 系列企业路由器 *配置指南 - 安全*.

在路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **l2tp-group** *group-number*，进入 L2TP 组视图。

步骤 3 配置用户验证方式，选择如下方案之一：

- 强制 LCP 重新协商，执行命令 **mandatory-lcp**。
- 强制本端 CHAP 验证，执行命令 **mandatory-chap**。
- 进行代理验证，不进行本步骤的配置。

缺省情况下，不进行本端 CHAP 验证，也不进行 LCP 重新协商，而是进行代理验证。

步骤 4 执行命令 **quit**，退回系统视图。

步骤 5 执行命令 **aaa**，进入 AAA 视图。

步骤 6 配置用户认证方式：

- 如果进行本地认证：执行命令 **local-user** *user-name* **password** *password*，设置用户名及密码。
- 代理验证、强制本端 CHAP 验证和 LCP 重协商这三种验证方式都需要在 LNS 侧设置用户名和密码。
- 如果进行 RADIUS 认证，请参见**配置 LAC 侧使用 RADIUS 认证**。

在 LNS 侧配置的用户名和口令必须与 LAC 侧配置的用户名和口令一致。

缺省的认证方法是本地认证。

----结束

3.4.4 为接入用户分配地址

每个接入用户都属于某个域，对于没有指定域的用户，将使用缺省域接入。当 LAC 与 LNS 之间的 L2TP 隧道连接建立后，LNS 从用户接入时所属的域的地址池中为接入用户分配 IP 地址。

背景信息

在 AR1200 中，每个接入用户都属于某个域，对于没有指定域的用户，将使用缺省域接入。当 LAC 与 LNS 之间的 L2TP 隧道连接建立后，LNS 从用户接入时所属的域的地址池中为接入用户分配 IP 地址。

操作步骤

步骤 1 关于地址池的配置和地址分配的详细介绍请参见《Huawei AR1200 系列企业路由器 配置指南 安全》以及《Huawei AR1200 系列企业路由器 配置指南 IP 业务》。

----结束

3.4.5 检查配置结果

LNS 侧配置成功后，您可以查看到 L2TP 的通道信息和 L2TP 的会话信息。

前提条件

已经完成 LNS 侧功能的所有配置。

操作步骤

- 使用 **display l2tp tunnel** 命令查看当前的 L2TP 通道的信息。
- 使用 **display l2tp session** 命令查看当前的 L2TP 会话的信息。
- 使用 **display l2tp-group** [*group-number*]命令查看指定 L2TP 组的配置信息。
- 使用 **display access-user** 命令查看当前在线用户的信息。

----结束

任务示例

在 LAC 侧和 LNS 侧都配置成功且连接保持时，在 LNS 侧执行命令 **display l2tp tunnel**，可以看到 L2TP 控制连接建立成功。例如：

```
<Huawei> display l2tp tunnel

 Total tunnel = 1
 LocalTID RemoteTID RemoteAddress    Port   Sessions RemoteName
 1        1         12.1.1.1         1701   1        LNS
```

执行 **display l2tp session** 命令，可查看 L2TP 会话连接已建立。例如：

```
<Huawei> display l2tp session

 Total session = 1
```

```
 LocalSID  RemoteSID  LocalTID
  1         1          1
```

执行 **display l2tp-group** [*group-number*]命令，可查看指定 L2TP 组的配置信息。例如：

```
<Huawei> display l2tp-group 1
 ----------------------------------------------
 L2tp-index          :    1
 GroupType           :    ACCEPT_DIALIN_L2TP
 TunnelAuth          :    Use tunnel authentication
 LocalName           :    lns
  Encrypt            :   0
 Hello               :   60
 Retransmit          :   5
 Timeout             :   2
 IfIndex             :   4294967295
 SrcIp               :   255.255.255.255
 VtNum               :   1
 RemoteName          :   lac1
 ForceChap           :   0
 LcpReg              :   0
 LcpMismatch         :   0
 tunnel each user    :   0
 ----------------------------------------------
```

3.5 配置 L2TP 连接参数

创建 L2TP 连接后，可以对 L2TP 的相关参数进行配置或调整。

3.5.1 建立配置任务

在调整 L2TP 连接前了解此特性的应用环境、配置此特性的前置任务和数据准备，可以帮助您快速、准确地完成配置任务。

应用环境

本节介绍的配置任务都是 L2TP 的公共配置，可以在 LAC 或 LNS 上应用：

- 隧道验证：隧道验证请求可由 LAC 或 LNS 任何一侧发起。只有两端都启用了隧道验证，两端密码完全一致并且不为空的情况下，隧道才能建立；否则本端将自动断开隧道连接。如果隧道两端都禁止隧道验证，隧道验证的密码一致与否将不起作用。
- AVP 隐藏传输：L2TP 协议使用 AVP（Attribute Value Pair，属性值对）来传递和协商 L2TP 的参数属性。为了保证安全，可以将 AVP 隐藏起来传输。
- Hello 报文发送：为了检测 LAC 和 LNS 之间通道的连通性，LAC 和 LNS 定期向对端发送 Hello 报文，接收方接收到 Hello 报文后进行响应。当 LAC 或 LNS 在指定时间间隔内未收到对端的 Hello 响应报文时，重复发送，如果重复发送超过 3 次仍没有收到对端的响应信息，则认为 L2TP 隧道已经断开，需要在 LAC 和 LNS 之间重新建立隧道连接。

说明

本节介绍的都是可选配置，多数应用中保持缺省设置即可。

前置任务

无

数据准备

在调整 L2TP 链接之前，需准备以下数据。

序号	数据
1	L2TP 组号
2	隧道验证密码
3	Hello 报文发送间隔时间

3.5.2 配置 L2TP 连接的安全选项

为了增强 L2TP 连接的安全性，可以配置隧道两端都需要对对方进行验证、配置在创建隧道连接之前启用隧道验证以及配置将 AVP 数据配置成为隐藏传输。

背景信息

在 LNS 侧或 LAC 侧进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **l2tp-group** *group-number*，进入 L2TP 组视图。

步骤 3 执行命令 **tunnel authentication**，启用隧道验证。

缺省情况下，启用隧道验证。

用户可根据实际需要决定是否在创建隧道连接之前启用隧道验证。为保证通道安全，建议不要禁用隧道验证。

说明

如果在 LAC 侧或 LNS 侧启用了隧道验证，则对端也需要进行同样的配置。

步骤 4 设置隧道验证的密码，选择如下方案之一：

- 如果要使密码以不加密的形式显示，执行命令 **tunnel password simple** *password*。
- 如果要使密码以加密的形式显示，执行命令 **tunnel password cipher** *password*。

缺省情况下，隧道验证的密码为空。

步骤 5 执行命令 **tunnel avp-hidden**，将 AVP 数据以隐含的方式传输。

缺省情况下，AVP 采用明文形式传输。隐含 AVP 功能只有在两端都使用隧道验证的情况下才起作用。

----结束

3.5.3 调整 L2TP 连接

创建 L2TP 连接后，可以对 L2TP 的通道 Hello 报文发送时间间隔进行配置或调整。

背景信息

在 LAS 或 LNS 侧进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **l2tp-group** *group-number*，进入 L2TP 组视图。

步骤 3 执行命令 **tunnel timer hello** *interval*，设置通道 Hello 报文发送时间间隔。

通道 Hello 报文的缺省发送时间间隔为 60 秒。

----结束

3.6 维护 L2TP

如何强制挂断通道、监控 L2TP 协议运行状况。

3.6.1 强制挂断通道

当用户数为零、网络发生故障或需要主动要求挂断通道时，可以执行 reset l2tp tunnel 命令来强制断开指定的隧道（Tunnel）连接和隧道内的会话连接。

操作步骤

- 在用户视图下执行命令 **reset l2tp tunnel** { **peer-name** *remote-name* | **local-id** *tunnel-id* }，强制挂断通道。

  指定删除的隧道名称时，将删除所有此名字的隧道；指定 *tunnel-id* 参数时，只删除指定的隧道。

  当用户数为零、网络发生故障或当管理员主动要求挂断通道时，就会产生隧道清除过程。LAC 和 LNS 任何一端也可主动发起隧道清除请求，接收到清除请求的一端发送确认信息（ACK），并等待一定时间再进行隧道清除操作，以确保 ACK 消息丢失后能够正确接收到对端重传过来的清除请求。

  强制挂断通道后，该通道上的所有控制连接与会话连接也将被清除。通道挂断后，当有新用户拨入时，还可重新建立。

----结束

3.6.2 监控 L2TP 协议运行状况

在日常维护工作中，执行与 L2TP 相关的 display 命令，了解 L2TP 协议的运行情况。

背景信息

在日常维护工作中，可以在任意视图下选择执行以下命令，了解 L2TP 协议的运行情况。

操作步骤

- 使用 **display l2tp session** 命令查看当前的 L2TP 会话的信息。

- 使用 **display l2tp tunnel** 命令查看当前的 L2TP 通道的信息。
- 使用 **display access-user** 命令查看当前在线用户的信息。
- 使用 **display l2tp-group** 命令查看当前 L2TP 组的信息。

----结束

3.6.3 调试 L2TP

在出现 L2TP 运行故障时，执行 debugging 命令进行调试，查看调试信息，定位故障并分析故障的原因。

背景信息

注意

打开调试开关将影响系统的性能。调试完毕后，应及时执行 **undo debugging all** 命令关闭调试开关。

在出现 L2TP 运行故障时，请在用户视图下执行下面的 **debugging** 命令进行调试，查看调试信息，定位故障并分析故障的原因。

打开调试信息开关的操作步骤请参考《*配置指南 系统管理*》中的“信息中心配置”。有关 **debugging** 命令的解释请参考《*Debugging 参考*》。

操作步骤

- 在用户视图下执行 **debugging l2tp all** 命令打开所有的 L2TP 调试信息开关。
- 在用户视图下执行 **debugging l2tp control** 命令打开控制报文调试开关。
- 在用户视图下执行 **debugging l2tp dump** 命令打开 PPP 报文内容的调试开关。
- 在用户视图下执行 **debugging l2tp error** 命令打开 L2TP 差错信息的调试开关。
- 在用户视图下执行 **debugging l2tp event** 命令打开 L2TP 的事件调试信息开关。
- 在用户视图下执行用 **debugging l2tp hidden** 命令打开隐藏 AVP 的调试信息开关。
- 在用户视图下执行 **debugging l2tp payload** 命令打开 L2TP 数据报文调试开关。
- 在用户视图下执行 **debugging l2tp timestamp** 命令打开 L2TP 时间戳信息调试开关。

----结束

3.7 配置举例

介绍 L2TP 的配置举例。

3.7.1 配置 NAS-Initialized VPN 示例（用户域名接入）

以典型组网为背景，介绍如何配置 NAS-Initialized VPN 示例，用户使用域名接入，在 LAC 侧和 LNS 侧均使用本地认证方式认证用户名和密码。

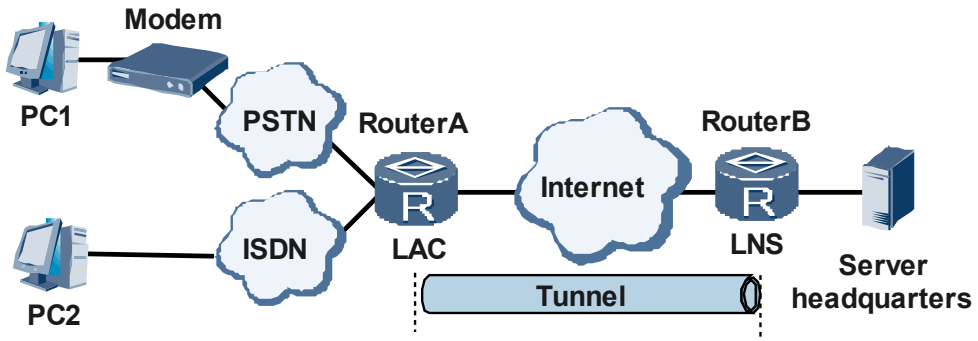
组网需求

如图 3-4，PC1 通过调制解调器 Modem 与 PSTN 网络相连，再连接到接入服务器 LAC（本例的 LAC 使用路由器 RouterA）。PC2 通过隧道 Tunnel 与 RouterA 相连。LAC 与 LNS 之间是广域网，LAC 与 LNS 通讯通过隧道传输。用户使用域名接入，在 LAC 侧和 LNS 侧均使用本地认证方式认证用户名和密码。

说明

配置与友商设备互通时，请根据实际情况确定是否需要配置翻转同步方式下 Serial 接口的时钟信号。

图 3-4 配置 NAS-Initialized VPN 组网图



配置思路

配置 NAS-Initialized VPN 的思路如下：

1. 用户需要与总部进行通讯，而总部网络的地址采用的是私有地址，用户无法通过 Internet 直接访问内部服务器。因此，需要建立 VPN，使用户可以访问内部网络的数据。
2. 用户接入时使用域名 huawei.com，LNS 侧需要在域下配置为用户分配地址的地址池。

数据准备

为完成此配置例，需准备如下的数据：

- 用户侧与 LAC 侧路由器的用户名、域名及口令。用户侧与 LAC 侧的用户名、域名及口令（需设置成相同值）
- LNS 侧采用的协议，选择通道验证方式（这里用 CHAP 验证）、通道的密码；LNS 侧本端名称及远端名称
- 虚拟接口模板编号、IP 地址、掩码
- L2TP 组编号
- 远端地址池编号、范围及掩码

操作步骤

步骤 1 用户侧的配置

创建一个拨号网络，号码为 Huawei1 路由器的接入号码，接收由 LNS 服务器端分配的地址。

在弹出的拨号终端窗口中输入用户名 vpdnuser@huawei.com，口令为 Hello（此用户名与口令已在公司 LNS 中注册）。

步骤 2 路由器 RouterA（LAC 侧）的配置

（本例中 LAC 侧与通道相连接的接口（Serial1/0/0）的 IP 地址为 202.38.160.1，LNS 侧与通道相连接的串口（Serial1/0/0）的 IP 地址为 202.38.160.2）。

在 Serial1/0/0 接口上配置 IP 地址。

```
<Huawei> system-view
[Huawei] sysname RouterA
[RouterA] interface serial 1/0/0
[RouterA-Serial1/0/0] link-protocol ppp
[RouterA-Serial1/0/0] ip address 202.38.160.1 255.255.255.0
[RouterA-Serial1/0/0] quit
```

设置一个 L2TP 组并配置相关属性。

```
[RouterA] l2tp enable
[RouterA] l2tp-group 1
[RouterA-l2tp1] tunnel name LAC
[RouterA-l2tp1] start l2tp ip 202.38.160.2 domain huawei.com
```

启用通道验证并设置通道验证密码。

```
[RouterA-l2tp1] tunnel authentication
[RouterA-l2tp1] tunnel password simple quidway
[RouterA-l2tp1] quit
```

设置用户名及口令（应与用户侧的设置一致）。

```
[RouterA] aaa
[RouterA-aaa] local-user vpdnuser@huawei.com password simple Hello
[RouterA-aaa] local-user vpdnuser@huawei.com service-type ppp
```

配置用户接入时的域

```
[RouterA-aaa] domain huawei.com
```

步骤 3 路由器 RouterB（LNS 侧）的配置

与通道相连接的接口上配置 IP 地址。

```
<Huawei> system-view
[Huawei] sysname RouterB
[RouterB] interface serial 1/0/0
[RouterB-Serial1/0/0] link-protocol ppp
[RouterB-Serial1/0/0] ip address 202.38.160.2 255.255.255.0
[RouterB-Serial1/0/0] quit
```

创建虚模板 Virtual-Template 并配置相关信息。

```
[RouterB] interface virtual-template 1
[RouterB-Virtual-Template1] ip address 192.168.0.1 255.255.255.0
[RouterB-Virtual-Template1] ppp authentication-mode chap
[RouterB-Virtual-Template1] quit
```

使能 L2TP 服务，并设置一个 L2TP 组。

```
[RouterB] l2tp enable
[RouterB] l2tp-group 1
```

配置 LNS 侧本端名称及接收的通道对端名称。

```
[RouterB-l2tp1] tunnel name LNS
[RouterB-l2tp1] allow l2tp virtual-template 1 remote LAC
```

启用通道验证并设置通道验证密码。

```
[RouterB-l2tp1] tunnel authentication
[RouterB-l2tp1] tunnel password simple quidway
```

强制进行本端 CHAP 验证。

```
[RouterB-l2tp1] mandatory-chap
[RouterB-l2tp1] quit
```

设置用户名及口令（应与 LAC 侧的设置一致）。

```
[RouterB] aaa
[RouterB-aaa] local-user vpdnuser@huawei.com password simple Hello
[RouterB-aaa] local-user vpdnuser@huawei.com service-type ppp
```

配置用户接入时的域

```
[RouterB-aaa] domain huawei.com
[RouterB-aaa-domain-huawei.com] quit
[RouterB-aaa] quit
```

配置给用户分配的地址池。

```
[RouterB] ip pool 1
[RouterB-ip-pool-1]network 192.168.0.0 mask 24
```

步骤 4 检查配置结果

配置成功后，当有 VPN 用户上线时，分别在 LAC 和 LNS 上执行 **display l2tp tunnel** 命令可发现隧道建立成功。以 LAC 侧的显示为例：

```
[RouterB] display l2tp tunnel

Total tunnel = 1
 LocalTID RemoteTID RemoteAddress    Port   Sessions RemoteName
 1        1         202.38.160.1     57344  1        LAC
```

执行 **display l2tp session** 命令可看到会话连接建立情况。以 LNS 侧的显示为例：

```
[RouterB] display l2tp session

 Total session = 1
 LocalSID  RemoteSID  LocalTID
  2036      1469       1
```

同时 VPN 用户可以访问公司总部。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 l2tp enable
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 domain huawei.com
```

```
 local-user vpdnuser@huawei.com password simple Hello
 local-user vpdnuser@huawei.com service-type ppp
#
interface Serial1/0/0
 link-protocol ppp
 ip address 202.38.160.1 255.255.255.0
#
l2tp-group 1
 tunnel password simple  quidway
 tunnel name LAC
 start l2tp ip 202.38.160.2 domain huawei.com
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
ip pool 1
 network 192.168.0.0 mask 255.255.255.0
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain huawei.com
 local-user vpdnuser@huawei.com password simple Hello
 local-user vpdnuser@huawei.com service-type ppp
#
interface Virtual-Template1
 ppp authentication-mode chap
 ip address 192.168.0.1 255.255.255.0
#
interface Serial1/0/0
 link-protocol ppp
 ip address 202.38.160.2 255.255.255.0
#
l2tp-group 1
 mandatory-chap
 allow l2tp virtual-template 1 remote LAC
 tunnel password simple  quidway
 tunnel name LNS
#
return
```

3.7.2 配置 NAS-Initialized VPN 示例（用户拨号接入）

以典型组网为背景，介绍如何配置 NAS-Initialized VPN 示例，VPN 用户通过 PSTN 或 ISDN 网络连接到接入服务器（NAS）。

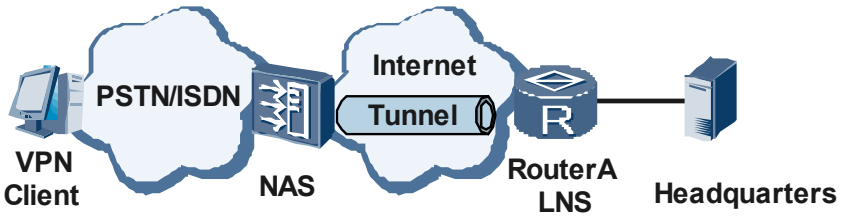
组网需求

如图 3-5，VPN 用户通过 PSTN 或 ISDN 网络连接到接入服务器（NAS），公司总部的 LNS 侧设备 Router A 通过 Internet 与 NAS 相连。

说明

配置与友商设备互通时，请根据实际情况确定是否需要配置翻转同步方式下 Serial 接口的时钟信号。

图 3-5 配置 NAS-Initialized VPN 组网图



配置思路

配置 NAS-Initialized VPN 的思路如下：

1. 用户采用普通的拨号上网方式。
2. 在接入服务器（NAS）处对用户进行验证，发现是 VPN 用户，则由接入服务器向 LNS 发起隧道连接的请求。
3. 在接入服务器与 LNS 隧道建立后，接入服务器把与 VPN 用户已经协商的内容作为报文内容传给 LNS。
4. LNS 根据预协商的内容决定是否接受此连接。
5. 用户与公司总部间的通信通过接入服务器与 LNS 之间的隧道传输。
6. 用户通过缺省域（域名为 default）接入，使用缺省的本地认证，从地址池分配地址。这种方式下，需要在 LNS 侧的 AAA 视图下配置地址池。

数据准备

为完成此配置例，需准备如下的数据：

- VPN 的用户名、口令、拨入号码
- RADIUS 验证的用户名为 VPN 的用户名，口令为 VPN 的用户口令
- LNS 侧设备的虚拟接口模板编号、模板 IP 地址及其掩码、L2TP 组编号
- 远端地址池编号、范围及地址掩码

操作步骤

步骤 1 用户侧的配置

在用户侧，在拨号网络窗口中输入 VPN 用户名 vpdnuser，口令 Hello，拨入号码为 170。在拨号后弹出的拨号终端窗口中输入用于 RADIUS 验证的用户名 username 和口令 userpass。

步骤 2 NAS 侧的配置

以接入服务器作为 LAC 侧设备。在接入服务器上配置拨入号码为 170。

在 RADIUS 服务器上设置一个用户名为 username、口令为 userpass 的 VPN 用户，并设置相应的 LNS 侧设备的 IP 地址（本例中，LNS 侧与通道相连接的接口 IP 地址为 202.38.160.2）。

将本端的设备名称定义为 A8010，需要进行通道验证，通道验证密码为 huawei。

说明

关于 A8010 的相关配置,请参考 A8010 相关配置手册。

步骤 3 路由器（LNS 侧）的配置

配置与通道相连接的串口的 IP 地址。

```
<Huawei> system-view
[Huawei] sysname RouterA
[RouterA] interface serial 1/0/0
[RouterA-Serial1/0/0] link-protocol ppp
[RouterA-Serial1/0/0] ip address 202.38.160.2 255.255.255.0
[RouterA-Serial1/0/0] quit
```

创建并配置虚拟接口模板。

```
[RouterA] interface virtual-template 1
[RouterA-Virtual-Template1] ip address 192.168.0.1 255.255.255.0
[RouterA-Virtual-Template1] ppp authentication-mode chap
[RouterA-Virtual-Template1] remote address pool 1
[RouterA-Virtual-Template1] quit
```

使能 L2TP 服务，创建一个 L2TP 组。

```
[RouterA] l2tp enable
[RouterA] l2tp-group 1
```

配置 LNS 侧本端名称及接收的通道对端名称。

```
[RouterA-l2tp1] tunnel name LNS
[RouterA-l2tp1] allow l2tp virtual-template 1 remote A8010
```

启用通道验证并设置通道验证密码。

```
[RouterA-l2tp1] tunnel authentication
[RouterA-l2tp1] tunnel password simple huawei
[RouterA-l2tp1] quit
```

定义一个地址池，为拨入用户分配地址。

```
[RouterA] ip pool 1
[RouterA-ip-pool-1]network 192.168.0.0 mask 24
```

设置用户名及口令（应与用户侧的设置一致）。

```
[RouterA] aaa
[RouterA-aaa] local-user vpdnuser password Hello
[RouterA-aaa] local-user vpdnuser service-type ppp
[RouterA-aaa] quit
```

步骤 4 检查配置结果

配置成功后，当有 VPN 用户上线时，分别在 LAC 和 LNS 上执行 **display l2tp tunnel** 命令可看到隧道建立成功。以 LNS 的显示为例：

```
<RouterA> display l2tp tunnel

 Total tunnel = 1
 LocalTID RemoteTID RemoteAddress    Port   Sessions RemoteName
 1        1         202.38.160.3     1701   1        A8010
```

执行 **display l2tp session** 命令可看到会话连接建立情况。以 LAC 的显示为例：

```
<RouterA> display l2tp session
 Total session = 1
 LocalSID  RemoteSID  LocalTID
  1469      2036        1
```

同时 VPN 用户可以访问公司总部。

----结束

配置文件

说明

这里只列了 LNS 的配置文件。

LNS 的配置文件

```
#
 sysname RouterA
#
 l2tp enable
#
ip pool 1
 network 192.168.0.0 mask 255.255.255.0
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 local-user vpdnuser password OUM!K%F<+$[Q=^Q`MAF4<1!!
 local-user vpdnuser service-type ppp
#
interface Virtual-Template1
 ppp authentication-mode chap
 remote address pool 1
 ip address 192.168.0.1 255.255.255.0
#
interface Serial 1/0/0
 link-protocol ppp
 ip address 202.38.160.2 255.255.255.0
#
l2tp-group 1
 allow l2tp virtual-template 1 remote A8010
 tunnel password simple  huawei
 tunnel name LNS
#
return
```

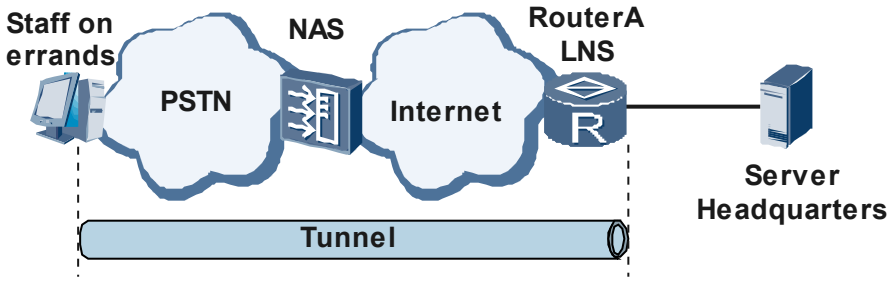
3.7.3 配置 Client-Initialized VPN 示例

以典型组网为背景，介绍如何配置 Client-Initialized VPN 示例，客户通过 PSTN 网络连接到接入服务器（NAS）。

组网需求

如图 3-6，出差员工通过 PSTN 网络连接到接入服务器（NAS），公司总部的 LNS 侧路由器 RouterA 通过 Internet 与 NAS 相连。要求由出差员工直接向 LNS 发起连接请求，与 LNS 的通讯数据通过隧道 Tunnel 传输。

图 3-6 配置 Client-Initialized VPN 组网图



配置思路

配置 Client-Initialized VPN 的思路如下：

1. 用户连接 Internet 后，直接由用户向 LNS 发起 Tunnel 连接的请求。
2. LNS 接受此连接请求之后，VPN 用户与 LNS 之间建立一条虚拟 Tunnel。
3. 用户与公司总部间的通信通过 VPN 用户与 LNS 之间的 Tunnel 进行。
4. 用户通过缺省域（域名为 default）接入，使用缺省的本地认证，从地址池分配地址。这种方式下，需要在 LNS 侧的 AAA 视图下配置地址池。

数据准备

为完成此配置例，需准备如下的数据：

- VPN 用户名、口令
- LNS 侧与通道相连接的串口的 IP 地址
- 虚拟接口模板编号、模板 IP 地址、掩码及 L2TP 组编号
- 远端地址池编号、地址池范围、掩码

操作步骤

步骤 1 用户侧的配置

用户侧主机上必须装有 L2TP 客户端软件，并通过拨号方式连接到 Internet。然后再进行如下配置（设置过程与相应的客户端软件有关）：

在用户侧设置 VPN 用户名为 vpdnuser，口令为 Hello。

将 LNS 的 IP 地址设为路由器的 Internet 接口地址（本例中 LNS 侧与通道相连接的串口的 IP 地址为 202.38.160.2）。

修改连接属性，将采用的协议设置为 L2TP。

如果用户侧主机提供 IPSec 功能，需要禁用 IPSec。

步骤 2 路由器（LNS 侧）的配置

创建并配置虚拟接口模板。

```
<Huawei> system-view
[Huawei] sysname RouterA
[RouterA] interface virtual-template 1
```

```
[RouterA-Virtual-Template1] ip address 192.168.0.1 255.255.255.0
[RouterA-Virtual-Template1] ppp authentication-mode chap
[RouterA-Virtual-Template1] remote address pool 1
[RouterA-Virtual-Template1] quit
```

使能 L2TP 服务，并创建一个 L2TP 组。

```
[RouterA] l2tp enable
[RouterA] l2tp-group 1
```

配置 LNS 侧本端名称及接收的通道对端名称。

```
[RouterA-l2tp1] tunnel name LNS
[RouterA-l2tp1] allow l2tp virtual-template 1 remote vpdnuser
```

禁止通道验证。

```
[RouterA-l2tp1] undo tunnel authentication
[RouterA-l2tp1] quit
```

定义一个地址池，为拨入用户分配地址。

```
[RouterA] ip pool 1
[RouterA-ip-pool-1]network 192.168.0.0 mask 24
```

设置用户名及口令（应与用户侧的设置一致）。

```
[RouterA] aaa
[RouterA-aaa] local-user vpdnuser password Hello
[RouterA-aaa] local-user vpdnuser service-type ppp
[RouterA-aaa] quit
```

步骤 3 检查配置结果

配置成功后，当该 VPN 用户上线时，在 LNS 上执行 **display l2tp tunnel** 命令可看到隧道建立成功。

```
[RouterA] display l2tp tunnel
 Total tunnel = 1
 LocalTID RemoteTID RemoteAddress    Port   Sessions RemoteName
 1        1         192.168.0.2      2134   1        vpdnuser
```

在 LNS 上执行 **display l2tp session** 命令可看到会话连接建立情况。

```
[RouterA] display l2tp session
 Total session = 1
 LocalSID  RemoteSID  LocalTID
 1576      1036       1
```

同时 VPN 用户可以访问公司总部。

----结束

配置文件

说明

这里只列出与 L2TP 相关的配置文件。

LNS 的配置文件。

```
#
 sysname RouterA
#
 l2tp enable
#
ip pool 1
 network 192.168.0.0 mask 255.255.255.0
```

```
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 local-user vpdnuser password OUM!K%F<+$[Q=^Q`MAF4<1!!
 local-user vpdnuser service-type ppp
#
interface Virtual-Template1
 ppp authentication-mode chap
 remote address pool 1
 ip address 192.168.0.1 255.255.255.0
#
l2tp-group 1
 undo tunnel authentication
 allow l2tp virtual-template 1 remote vpdnuser
 tunnel name LNS
#
return
```

3.7.4 配置 LAC-Auto-Initiated VPN 示例

以典型组网为背景，介绍如何配置 LAC-Auto-Initiated VPN 示例。

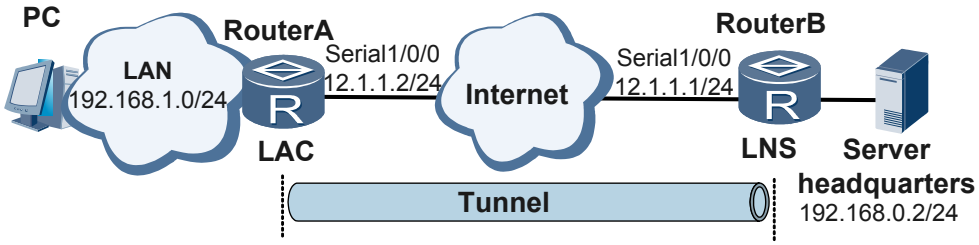
组网需求

企业总部有多个部门，不同部门使用独立的网段。企业分支机构有多个部门员工，不同部门的员工希望接入自己部门的网络，此时可以通过分支路由器和企业总部建立 L2TP 隧道实现通信。如图 3-7 所示，分支部门的 PC 通过 LAN 侧口连接到分支服务器 LAC（本例的 LAC 使用路由器 RouterA），LAC 与 LNS 之间通过 Internet 连接。通过 LAC 与 LNS 之间的隧道，实现 PC 与总部之间的数据通信。

说明

配置与友商设备互通时，请根据实际情况确定是否需要配置翻转同步方式下 Serial 接口的时钟信号。

图 3-7 配置 LAC-Auto-Initiated VPN 组网图



配置思路

配置 LAC-Auto-Initiated VPN 的思路如下：

1. 在 LAC 上配置 L2TP 功能并创建虚拟 PPP 用户，PPP 用户通过 L2TP 隧道向总部发出接入请求，总部认证成功后分配公司内部私网地址。
2. 在 LAC 上配置路由，目的地址段是总部，出接口是虚拟 PPP 用户接口，并使能 LAC 自拨号功能。
3. LNS 侧需要在域下配置为用户分配地址的地址池。

数据准备

为完成此配置例，需准备如下的数据：

- LAC 虚拟接口模板编号、IP 地址、掩码
- L2TP 组编号
- LNS 侧采用的协议，选择通道验证方式（这里用 CHAP 验证）、通道的密码；LNS 侧本端名称及远端名称
- 远端地址池编号、范围及掩码

操作步骤

步骤 1 RouterA（LAC 侧）的配置

（本例中 LAC 侧与通道相连接的接口（Serial1/0/0）的 IP 地址为 12.1.1.2，LNS 侧与通道相连接的串口（Serial1/0/0）的 IP 地址为 12.1.1.1）。

在 Serial1/0/0 接口上配置 IP 地址。

```
<Huawei> system-view
[Huawei] sysname RouterA
[RouterA] interface serial 1/0/0
[RouterA-Serial1/0/0] link-protocol ppp
[RouterA-Serial1/0/0] ip address 12.1.1.2 255.255.255.0
[RouterA-Serial1/0/0] quit
```

设置用户名及口令（应与用户侧的设置一致）。

```
[RouterA] aaa
[RouterA-aaa] local-user huawei password 123
[RouterA-aaa] local-user huawei service-type ppp
[RouterA-aaa] quit
```

设置一个 L2TP 组并配置相关属性。

```
[RouterA] l2tp enable
[RouterA] l2tp-group 1
[RouterA-l2tp1] tunnel name LAC
[RouterA-l2tp1] start l2tp ip 12.1.1.1 fullusername huawei
```

启用通道验证并设置通道验证密码。

```
[RouterA-l2tp1] tunnel authentication
[RouterA-l2tp1] tunnel password simple 123
[RouterA-l2tp1] quit
```

配置虚拟 PPP 用户的用户名和密码，及 PPP 验证方式以及 IP 地址。

```
[RouterA] interface virtual-template 1
[RouterA-Virtual-Template1] ppp pap local-user huawei password simple 123
[RouterA-Virtual-Template1] ip address 13.1.1.2 255.255.255.0
[RouterA-Virtual-Template1] quit
```

配置私网路由，访问公司总部的报文将通过 L2TP 隧道转发。

```
[RouterA] ip route-static 192.168.0.0 255.255.255.0 Virtual-Template1
```

触发 LAC 建立 L2TP 隧道。

```
[RouterA] interface virtual-template 1
[RouterA-virtual-template1] l2tp-auto-client enable
```

步骤 2 RouterB（LNS 侧）的配置

与通道相连接的接口上配置 IP 地址。

```
<Huawei> system-view
[Huawei]  sysname RouterB
[RouterB] interface serial 1/0/0
[RouterB-Serial1/0/0] link-protocol ppp
[RouterB-Serial1/0/0] ip address 12.1.1.1 255.255.255.0
[RouterB-Serial1/0/0] quit
```

创建虚模板 Virtual-Template 并配置相关信息。

```
[RouterB] interface virtual-template 1
[RouterB-Virtual-Template1] ppp authentication-mode pap
[RouterB-Virtual-Template1] remote address pool 1
[RouterB-Virtual-Template1] ip address 13.1.1.1 255.255.255.0
[RouterB-Virtual-Template1] quit
```

使能 L2TP 服务，并设置一个 L2TP 组。

```
[RouterB] l2tp enable
[RouterB]l2tp-group 1
```

配置 LNS 侧本端名称及接收的通道对端名称。

```
[RouterB-l2tp1] tunnel name LNS
[RouterB-l2tp1]allow l2tp virtual-template 1 remote LAC
```

启用通道验证并设置通道验证密码。

```
[RouterB-l2tp1] tunnel authentication
[RouterB-l2tp1] tunnel password simple 123
[RouterB-l2tp1] quit
```

设置用户名及口令（应与 LAC 侧的设置一致）。

```
[RouterB] aaa
[RouterB-aaa] local-user huawei password 123
[RouterB-aaa] local-user huawei service-type ppp
[RouterB-aaa] quit
```

配置给用户分配的地址池。

```
[RouterB] ip pool 1
[RouterB-ip-pool-1] gateway-list 13.1.1.1
[RouterB-ip-pool-1] network 13.1.1.0 mask 255.255.255.0
[RouterB-ip-pool-1] quit
```

步骤 3 检查配置结果

配置成功后，当有 VPN 用户上线时，分别在 LAC 和 LNS 上执行 **display l2tp tunnel** 命令可发现隧道建立成功。以 LNS 侧的显示为例：

```
[RouterB] display l2tp tunnel

 Total tunnel = 1
 LocalTID RemoteTID RemoteAddress    Port   Sessions RemoteName
 1        1         12.1.1.1         1701   1        LNS
```

执行 **display l2tp session** 命令可看到会话连接建立情况。以 LNS 侧的显示为例：

```
[RouterB] display l2tp session

 Total session = 1
 LocalSID  RemoteSID  LocalTID
  1         1          1
```

同时 VPN 用户可以访问公司总部。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 l2tp enable
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 local-user huawei password OUM!K%F<+$[Q=^Q`MAF4<1!!
 local-user huawei service-type ppp
#
interface Virtual-Template1
 ppp pap local-user huawei password simple 123
 ip address 13.1.1.2 255.255.255.0
 l2tp-auto-client enable
#
interface Serial1/0/0
 link-protocol ppp
 ip address 12.1.1.2 255.255.255.0
#
l2tp-group 1
 tunnel password simple  123
 tunnel name LAC
 start l2tp ip 12.1.1.1 fullusername huawei
#
ip route-static 192.168.0.0 255.255.255.0 Virtual-Template1
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 l2tp enable
#
ip pool 1
 gateway-list 13.1.1.1
 network 13.1.1.0 mask 255.255.255.0
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 local-user huawei password OUM!#(%<+$[Q=^Q`MAF4<1!!
 local-user huawei service-type ppp
#
interface Virtual-Template1
 ppp authentication-mode pap
 remote address pool 1
 ip address 13.1.1.1 255.255.255.0
#
interface Serial1/0/1
 link-protocol ppp
 ip address 12.1.1.1 255.255.255.0
#
l2tp-group 1
 allow l2tp virtual-template 1 remote LAC
 tunnel password simple  123
 tunnel name LNS
#
return
```

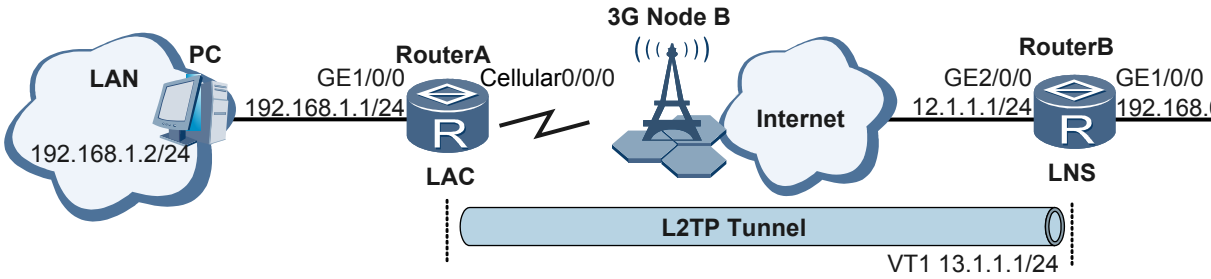
3.7.5 配置 LAC-Auto-Initiated VPN 示例（使用 3G 接口）

以 3G 网络为背景，介绍如何配置 LAC-Auto-Initiated VPN 示例。

组网需求

企业总部有多个部门，不同部门使用独立的网段。企业分支机构有多个部门员工，不同部门的员工希望接入自己部门的网络，此时可以通过分支路由器和企业总部建立 L2TP 隧道实现通信，其中分支路由器使用 3G 数据卡通过无线方式和总部通信。如图 3-8 所示，分支部门的 PC 通过 LAN 侧口连接到分支服务器 LAC（本例的 LAC 使用路由器 RouterA），LAC 与 LNS 之间通过 3G 网络连接。通过 LAC 与 LNS 之间的隧道，实现 PC 与总部之间的数据通信。

图 3-8 配置 LAC-Auto-Initiated VPN 组网图



配置思路

配置 LAC-Auto-Initiated VPN 的思路如下：

1. 配置 LAC 上的 3G 接口的拨号串，访问公网地址的路由。
2. 在 LAC 上配置 L2TP 功能，PPP 用户通过 L2TP 隧道向总部发出接入请求，总部认证成功后建立隧道。
3. 在 LAC 上配置路由，目的地址段是总部私网地址，出接口是虚拟 PPP 用户接口，并使能 LAC 自拨号功能。
4. 在 LNS 上配置 L2TP 功能及 PPP 用户，并配置访问公网的路由。

数据准备

为完成此配置示例，需准备如下的数据：

- 3G 接口的拨号串*99#，具体拨号串请咨询 3G 网络服务商
- LAC 虚拟接口模板编号、IP 地址、掩码
- L2TP 组编号
- LNS 侧采用的协议，选择通道验证方式（这里用 PAP 验证）、通道的密码，LNS 侧本端名称及远端名称

操作步骤

步骤 1 RouterA（LAC 侧）的配置

（本例中 LAC 侧与通道相连接的 3G 接口（Cellular0/0/0）的 IP 地址由运营商自动分配，LNS 侧与通道相连接的接口（GE2/0/0）的 IP 地址为 12.1.1.1）。

在 Cellular0/0/0 接口上配置拨号。

```
<Huawei> system-view
[Huawei] sysname RouterA
[RouterA] interface cellular 0/0/0
[RouterA-Cellular0/0/0] link-protocol ppp
[RouterA-Cellular0/0/0] ip address ppp-negotiate
[RouterA-Cellular0/0/0] dialer enable-circular
[RouterA-Cellular0/0/0] dialer-group 1
[RouterA-Cellular0/0/0] dialer timer autodial 60
[RouterA-Cellular0/0/0] dialer number *99# autodial
[RouterA-Cellular0/0/0] quit
```

配置私网 IP 地址。

```
[RouterA] interface gigabitEthernet 1/0/0
[RouterA-GigabitEthernet1/0/0] ip address 192.168.1.1 255.255.255.0
[RouterA-GigabitEthernet1/0/0] quit
```

设置一个 L2TP 组并配置相关属性。

```
[RouterA] l2tp enable
[RouterA] l2tp-group 1
[RouterA-l2tp1] tunnel name LAC
[RouterA-l2tp1] start l2tp ip 12.1.1.1 fullusername huawei
```

启用通道验证并设置通道验证密码。

```
[RouterA-l2tp1] tunnel authentication
[RouterA-l2tp1] tunnel password simple 123
[RouterA-l2tp1] quit
```

配置虚拟 PPP 用户的用户名和密码，及 PPP 验证方式以及 IP 地址。

```
[RouterA] interface virtual-template 1
[RouterA-Virtual-Template1] ppp pap local-user huawei password simple 123
[RouterA-Virtual-Template1] ip address 13.1.1.2 255.255.255.0
[RouterA-Virtual-Template1] quit
```

配置私网路由，访问公司总部的报文将通过 L2TP 隧道转发。

```
[RouterA] ip route-static 192.168.0.0 255.255.255.0 Virtual-Template1
```

配置公网路由，访问公司总部的报文将通过 3G 接口转发。

```
[RouterA] ip route-static 12.1.1.1 255.255.255.255 Cellular0/0/0
```

触发 LAC 建立 L2TP 隧道。

```
[RouterA] interface virtual-template 1
[RouterA-virtual-template1] l2tp-auto-client enable
```

步骤 2 RouterB（LNS 侧）的配置

与通道相连接的接口上配置 IP 地址。

```
<Huawei> system-view
[Huawei] sysname RouterB
[RouterB] interface gigabitEthernet 2/0/0
[RouterB-GigabitEthernet2/0/0] ip address 12.1.1.1 255.255.255.0
[RouterB-GigabitEthernet2/0/0] quit
```

配置私网 IP 地址。

```
[RouterB] interface GigabitEthernet 1/0/0
```

```
[RouterB-GigabitEthernet1/0/0] ip address 192.168.0.1 255.255.255.0
[RouterB-GigabitEthernet1/0/0] quit
```

创建虚模板 Virtual-Template 并配置相关信息。

```
[RouterB] interface virtual-template 1
[RouterB-Virtual-Template1] ppp authentication-mode pap
[RouterB-Virtual-Template1] ip address 13.1.1.1 255.255.255.0
[RouterB-Virtual-Template1] quit
```

使能 L2TP 服务，并设置一个 L2TP 组。

```
[RouterB] l2tp enable
[RouterB] l2tp-group 1
```

配置 LNS 侧本端名称及接收的通道对端名称。

```
[RouterB-l2tp1] tunnel name LNS
[RouterB-l2tp1] allow l2tp virtual-template 1 remote LAC
```

启用通道验证并设置通道验证密码。

```
[RouterB-l2tp1] tunnel authentication
[RouterB-l2tp1] tunnel password simple 123
[RouterB-l2tp1] quit
```

设置用户名及口令（应与 LAC 侧的设置一致）。

```
[RouterB] aaa
[RouterB-aaa] local-user huawei password 123
[RouterB-aaa] local-user huawei service-type ppp
[RouterB-aaa] quit
```

配置私网路由，访问分支机构的报文将通过 L2TP 隧道转发。

```
[RouterB] ip route-static 192.168.1.0 255.255.255.0 Virtual-Template1
```

步骤 3 检查配置结果

配置成功后，分别在 LAC 和 LNS 上执行 **display l2tp tunnel** 命令可发现隧道建立成功，以 LAC 侧的显示为例：

```
[RouterB] display l2tp tunnel

 Total tunnel = 1
 LocalTID RemoteTID RemoteAddress    Port   Sessions RemoteName
 1        1         12.1.1.1         1701   1        LNS
```

执行 **display l2tp session** 命令可看到会话连接建立情况，以 LNS 侧的显示为例：

```
[RouterB] display l2tp session

 Total session = 1
 LocalSID  RemoteSID  LocalTID
  1         1          1
```

分支机构的 PC 可以访问公司总部服务器。

----结束

配置文件

- RouterA 的配置文件

```
#
 sysname RouterA
#
 l2tp enable
#
```

```
interface Virtual-Template1
 ppp pap local-user huawei password simple 123
 ip address 13.1.1.2 255.255.255.0
 l2tp-auto-client enable
#
interface Cellular0/0/0
 link-protocol ppp
 ip address ppp-negotiate
 dialer enable-circular
 dialer-group 1
 dialer timer autodial 60
 dialer number *99# autodial
#
interface GigabitEthernet1/0/0
 ip address 192.168.1.1 255.255.255.0
#
l2tp-group 1
 tunnel password simple  123
 tunnel name LAC
 start l2tp ip 12.1.1.1 fullusername huawei
#
ip route-static 192.168.0.0 255.255.255.0 Virtual-Template1
ip route-static 0.0.0.0 0.0.0.0 Cellular0/0/0
#
return
```

- RouterB 的配置文件

```
#
 sysname RouterB
#
 l2tp enable
#
aaa
 local-user huawei password !9$"OZ$+1"-',917]_2Y71!!
 local-user huawei service-type ppp
#
interface Virtual-Template1
 ppp authentication-mode pap
 ip address 13.1.1.1 255.255.255.0
#
interface GigabitEthernet1/0/0
 ip address 192.168.0.1 255.255.255.0
#
interface GigabitEthernet2/0/0
 ip address 12.1.1.1 255.255.255.0
#
l2tp-group 1
 allow l2tp virtual-template 1 remote LAC
 tunnel password simple  123
 tunnel name LNS
#
ip route-static 192.168.1.0 255.255.255.0 Virtual-Template1
#
return
```

4 IPSec 配置

关于本章

介绍在 IP 层通过加密与数据源认证等方式，来保证数据报在网络上传输时的私有性、完整性、真实性和防重放。IKE 为 IPSec 提供了自动协商交换密钥、建立安全联盟的服务，以简化 IPSec 的使用和管理。

4.1 IPSec 概述
IPSec 协议族是 IETF（Internet Engineering Task Force）制定的一系列协议，它为 IP 数据包提供了高质量的、可互操作的、基于密码学的安全性。特定的通信双方在 IP 层通过加密与数据源认证等方式，来保证数据报在网络上传输时的私有性、完整性、真实性和防重放。

4.2 AR1200 支持的 IPSec 特性
AR1200 支持的 IPSec 包括：手工方式建立 IPSec 隧道、IKE 协商方式建立 IPSec 隧道、采用 IPSec 虚拟隧道接口建立 IPSec 隧道、采用 Efficient VPN 策略建立 IPSec 隧道。

4.3 配置采用手工方式建立的 IPSec 隧道
采用手工方式建立的 IPSec 隧道主要应用于网络拓扑比较简单的情况下。

4.4 配置采用 IKE 方式协商的 IPSec 隧道
IKE 具有一套自保护机制，可以在不安全的网络上安全地分发密钥、认证身份、建立 IPSec 安全联盟。

4.5 配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道
配置 IPSec 虚拟隧道接口，并在 IPSec 虚拟隧道接口视图下应用安全框架使得 IPSec 虚拟隧道的配置生效。

4.6 配置采用 Efficient VPN 策略建立 IPSec 隧道
部署 Efficient VPN 可以简化远程设备的配置，将管理员从繁琐的 IPSec 配置中解脱出来。

4.7 维护 IPSec
显示 IPSec 的配置信息，清除 IPSec 的统计信息。

4.8 配置举例
介绍使用 IPSec 提高数据传输安全性的各种示例。

4.1 IPSec 概述

IPSec 协议族是 IETF（Internet Engineering Task Force）制定的一系列协议，它为 IP 数据包提供了高质量的、可互操作的、基于密码学的安全性。特定的通信双方在 IP 层通过加密与数据源认证等方式，来保证数据报在网络上传输时的私有性、完整性、真实性和防重放。

IPSec 通过认证头 AH（Authentication Header）和封装安全载荷 ESP（Encapsulating Security Payload）这两个安全协议来实现上述目标。因特网密钥交换协议 IKE（Internet Key Exchange）为 IPSec 提供自动协商交换密钥、建立和维护安全联盟的服务，以简化 IPSec 的使用和管理。

IPSec 的基本概念包括：

- 安全联盟
  - 安全联盟 SA（Security Association）是通信对等体间对某些要素的约定。例如，使用哪种协议（AH、ESP、AH 和 ESP 两种协议结合使用）、协议的封装模式（传输模式和隧道模式）、加密算法（DES、3DES 和 AES）、特定数据流中保护数据的共享密钥以及密钥的生存周期等。安全联盟是 IPSec 的基础，也是 IPSec 的本质。
  - 安全联盟是单向的，在两个对等体之间的双向通信，最少需要两个 SA 来分别对两个方向的数据流进行安全保护。同时，如果两个对等体希望同时使用 AH 和 ESP 来进行安全通信，则每个对等体都会针对每一种协议来构建一个独立的 SA。
  - SA 由一个三元组来唯一标识，这个三元组包括安全参数索引 SPI（Security Parameter Index）、目的 IP 地址、安全协议号（AH 或 ESP）。
- IPSec 协议的封装模式
  - 传输模式。在传输模式下，AH 或 ESP 被插入到 IP 头之后但在所有传输层协议之前。如图 4-1 所示。
  - 隧道模式。在隧道模式下，AH 或 ESP 插在原始 IP 头之前，另外生成一个新 IP 头放到 AH 或 ESP 之前。如图 4-2 所示。

图 4-1 传输模式下的报文格式

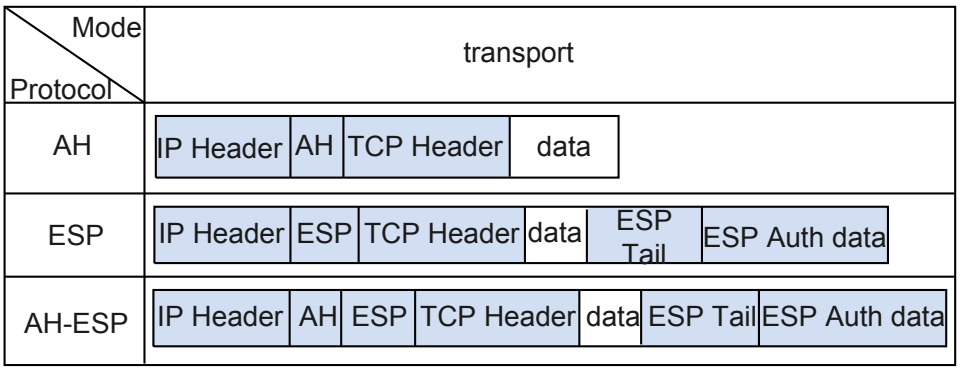
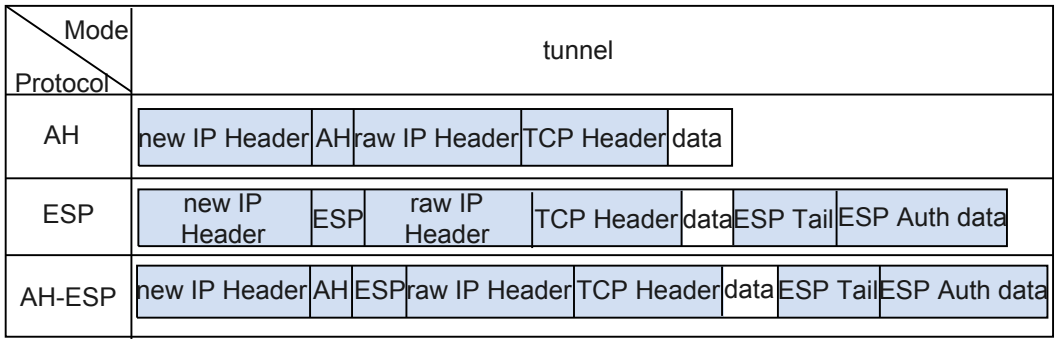


图 4-2 隧道模式下的报文格式



- 认证算法与加密算法
  - IPSec 可以使用 MD5（Message Digest 5）、SHA-1（Secure Hash Algorithm）、SHA-2 三种认证算法。MD5 算法的计算速度比 SHA-1 算法快，而 SHA-1 算法的安全强度比 MD5 算法高，SHA-2 算法相对于 SHA-1 来说，加密数据位数的上升增加了破解的难度，使得安全性能要远远高于 SHA-1。
  - IPSec 可以使用 DES、3DES（Triple Data Encryption Standard）和 AES（Advanced Encryption Standard）三种加密算法。其中，AES 使用 128bit、192bit 或 256bit 密钥长度的加密算法对明文进行加密。
- 协商方式

  IPSec 有两种协商方式建立安全联盟，一种是手工方式（**manual**），一种是 IKE 自动协商方式（**isakmp**）。

4.2 AR1200 支持的 IPSec 特性

AR1200 支持的 IPSec 包括：手工方式建立 IPSec 隧道、IKE 协商方式建立 IPSec 隧道、采用 IPSec 虚拟隧道接口建立 IPSec 隧道、采用 Efficient VPN 策略建立 IPSec 隧道。

AR1200 实现了上述介绍的 IPSec 全部功能。

- 手工方式和 IKE 协商方式建立 IPSec 安全隧道都是基于 ACL。通过 IPSec，对等体之间（指 IPSec 对等体）能够对不同的数据流实施不同的安全保护（认证、加密或两者同时使用）。

  简要的配置思路如下：

  1. 定义被保护的数据流。通过配置 ACL 来区分数据流。
  2. 定义安全提议。通过配置安全提议来确定安全保护所用到的安全协议、认证算法、加密算法和封装模式。
  3. 定义安全策略或安全策略组。通过配置安全策略或安全策略组来确定数据流和安全提议的关联（即定义对何种数据流实施何种保护）、SA 的协商方式、对等体 IP 地址的设置（即保护路径的起/终点）、所需要的密钥和 SA 的生存周期等。
  4. 在路由器接口上实施安全策略。

     IPSec 还支持 MPLS VPN 接入，包括：
     - VPN 实例与安全联盟进行关联。
     - 路由器作为 PE 设备，VPN 实例与 PE 上连接 CE 的接口进行关联。

- 采用 IPSec 虚拟隧道接口建立 IPSec 隧道是基于路由方式。这种方式下，由路由来选择需要保护的数据流，通过配置安全框架并在 IPSec 虚拟隧道接口上应用安全框架来完成 IPSec 的配置。

  简要的配置思路如下：

  1. 定义安全提议。通过配置安全提议来确定安全保护所用到的安全协议、认证算法、加密算法和封装模式。
  2. 定义 IKE Peer。
  3. 定义安全框架。通过配置安全框架来选择保护数据流时使用的安全提议，IKE 对等体的参数和 SA 的生存周期等。
  4. 在接口上应用安全框架。

- Efficient VPN 方案中，Remote 设备仅需配置接入总部的 IP 地址、预共享密钥等 IPSec 隧道必须参数，而 IKE 协商认证算法、IKE 协商加密算法、IPSec 策略等大部分 IPSec VPN 参数都可以在 Server 端进行预配置。Remote 设备发起 IPSec 隧道协商建立时，Remote 设备将所支持的 IKE 协商认证能力、IKE 协商认证的加解密能力、IPSec 策略等参数全部发往 Server，Server 端根据管理员预配置的 IPSec 隧道参数与 Remote 设备上报的 IPSec 能力数据协商建立 IPSec 隧道。

说明

Efficient VPN 功能使用 License 授权，缺省情况下，设备的 Efficient VPN 功能受限无法使用。如果需要使用 Efficient VPN 功能，请联系华为办事处申请并购买如下 License，

- AR1200 安全业务增值包

4.3 配置采用手工方式建立的 IPSec 隧道

采用手工方式建立的 IPSec 隧道主要应用于网络拓扑比较简单的情况下。

4.3.1 建立配置任务

在配置采用手工方式建立 IPSec 隧道之前，了解其应用环境，以及配置采用手工方式建立的 IPSec 隧道需要提前完成的任务和准备的数据。

应用环境

网络拓扑结构较简单时，采用手工方式建立 IPSec 隧道。

前置任务

在配置采用 Manual 方式建立的 IPSec 隧道之前，需要完成以下任务：

- 配置接口的链路层协议参数，使接口的链路协议状态为 Up。
- 配置路由，保证 IPSec 服务发起端设备的接口 IP 地址和 IPSec 服务终结端设备的接口 IP 地址之间路由可达。

数据准备

在配置采用 Manual 方式建立的 IPSec 隧道之前，需要准备以下数据。

序号	数据
1	高级 ACL 相关参数
2	安全提议名称、使用的安全协议、AH 协议的认证算法、ESP 协议的认证算法、ESP 协议的加密算法、报文封装形式
3	安全策略名称和序列号、隧道本端和对端 IP 地址、AH 协议入方向和出方向的 SPI、ESP 协议入方向和出方向的 SPI、AH 协议入方向和出方向的认证密钥（以字符串方式输入）、ESP 协议入方向和出方向的认证密钥（以字符串形式输入）、AH 协议入方向和出方向的认证密钥（以 16 进制方式输入）、ESP 协议入方向和出方向的认证密钥（以 16 进制方式输入）、ESP 协议入方向和出方向的加密密钥（以 16 进制方式输入）、（可选）VPN 实例名称
4	应用安全策略组的接口类型和编号

说明

可根据实际应用，选择 AH 或 ESP 协议的一种进行相应的数据规划。

4.3.2 创建需要保护的数据流

IPSec 能够对不同的数据流进行安全保护。实际应用中，需要首先通过 ACL 定义数据流，再在安全策略中引用该 ACL，从而起到保护该数据流的作用。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **acl** [**number**] *acl-number* [**match-order** { **config** | **auto** }]，创建一个高级访问控制列表并进入其视图。

步骤 3 执行命令 **rule**，在 ACL 视图下，配置访问控制规则。

说明

- 需要精确配置 ACL。建议只对确实需要 IPSec 保护的数据流进行保护，即配置对应的 ACL 规则的动作关键字为 permit。
- 对于有不同安全要求的数据流，需要创建不同的 ACL 和相应的安全策略。

----结束

4.3.3 配置 IPSec 安全提议

隧道两端的设备，安全协议、认证算法、加密算法、报文封装格式需要配置相同。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec proposal** *proposal-name*，创建安全提议并进入安全提议视图。

步骤 3 （可选）执行命令 **transform** { **ah** | **esp** | **ah-esp** }，配置安全协议。

缺省情况下采用 **esp**，即 RFC2406 规定的 ESP 协议。

步骤 4 （可选）执行命令 **ah authentication-algorithm** { **md5** | **sha1** | **sha2-256** | **sha2-384** | **sha2-512** }，设置 AH 协议采用的认证算法。

缺省情况下，在 IPSec 提议中 AH 协议采用 MD5 认证。

步骤 5 （可选）执行命令 **esp authentication-algorithm** [**md5** | **sha1** | **sha2-256** | **sha2-384** | **sha2-512**]，设置 ESP 协议采用的认证算法。

缺省情况下，ESP 协议和 AH 协议采用的认证算法都是 **md5**。

需要首先通过 **transform** 命令选择了相应的安全协议后，才能配置该安全协议所需的安全算法。例如，如果使用 **transform** 命令选择了 **esp**，则只能配置 ESP 所需的安全算法，而 AH 所需的安全算法则不能配置。

步骤 6 （可选）执行命令 **esp encryption-algorithm** [**3des** | **des** | **aes-128** | **aes-192** | **aes-256**]，设置 ESP 协议采用的加密算法。

缺省情况下，ESP 协议采用的加密算法是 **des**。

步骤 7 （可选）执行命令 **encapsulation-mode** { **transport** | **tunnel** }，选择 IPSec 对 IP 报文的封装形式。

缺省情况下采用 Tunnel，即隧道模式。

----结束

4.3.4 配置 IPSec 安全策略

配置手工方式建立的 IPSec 隧道的安全策略。

背景信息

注意

隧道两端的设备，安全联盟参数 SPI、string-key、authentication-hex 和 encryption-hex 需要镜像配置，即本端的入方向安全联盟参数必须和对端的出方向安全联盟参数一样；本端的出方向安全联盟参数必须和对端的入方向安全联盟参数一样。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec policy** *policy-name seq-number* **manual**，创建安全策略。

一个安全策略组最多支持配置 10000 条安全策略。缺省情况下，没有安全策略存在。

步骤 3 执行命令 **security acl** *acl-number*，在安全策略中引用访问控制列表。

一条安全策略只能引用一条 ACL。如果设置安全策略引用了多于一条 ACL，最后配置的有效。

步骤 4 执行命令 **proposal** *proposal-name*，在安全策略中引用安全提议。

如果安全策略为手工方式,一条安全策略只能引用一个安全提议，并且如果已经设置了安全提议，必须先取消原先的安全提议才能设置新的安全提议。在安全隧道的两端设置的安全策略所引用的安全提议必须设置成采用同样的安全协议、算法和报文封装形式。

步骤 5 执行命令 **tunnel local** *ip-address*，配置隧道的本端地址。

步骤 6 执行命令 **tunnel remote** *ip-address*，配置隧道的对端地址。

步骤 7 执行命令 **sa spi inbound** { **ah** | **esp** } *spi-number*，配置入方向安全联盟的 SPI。

说明

执行该命令配置的安全协议必须与 **4.3.3 配置 IPSec 安全提议**中 **transform** 命令配置的安全协议一致。如果在 **transform** 命令中配置的安全协议为 **ah-esp**，则 **sa spi** 命令必须同时配置 **ah** 和 **esp** 两种安全协议。

步骤 8 执行命令 **sa spi outbound** { **ah** | **esp** } *spi-number*，配置出方向安全联盟的 SPI。

说明

- 在配置安全联盟时，需要分别设置 **inbound** 和 **outbound** 两个方向的安全联盟的参数。
- 在安全隧道的两端设置的安全联盟参数必须是完全匹配的。本端的入方向安全联盟的 SPI 必须和对端的出方向安全联盟的 SPI 一样；本端的出方向安全联盟的 SPI 必须和对端的入方向安全联盟的 SPI 一样。

步骤 9 执行命令 **sa authentication-hex** { **inbound** | **outbound** } { **ah** | **esp** } *hex-key*，配置协议的认证密钥（以 16 进制方式输入）。

步骤 10 执行命令 **sa string-key** { **inbound** | **outbound** } { **ah** | **esp** } *string-key*，配置协议的认证密钥（以字符串方式输入）。

注意

在安全隧道的两端，应当以相同的方式输入密钥。如果一端以字符串方式输入密钥，另一端以 16 进制方式输入密钥，则不能正确地建立安全隧道。

如果分别以两种方式输入了密钥，则最后设定的密钥有效。

步骤 11 执行命令 **sa encryption-hex** { **inbound** | **outbound** } **esp** *hex-key*，配置 ESP 协议的加密密钥（以 16 进制方式输入）。

说明

- 如果设置的是 AH 安全协议，选择 **sa authentication-hex** 和 **sa string-key** 其中的一条命令执行即可。
- 如果设置的是 ESP 安全协议，选择 **sa authentication-hex**、**sa string-key** 和 **sa encryption-hex** 其中的一条命令执行即可。
- **sa spi** 命令需要与 **sa authentication-hex** 或 **sa string-key** 或 **sa encryption-hex** 共同使用，才能成功创建手工 IPSec 隧道。

步骤 12 （可选）执行命令 **sa binding vpn-instance** *vpn-instance-name*，配置 VPN 实例与安全联盟关联。

----结束

4.3.5 应用 IPSec 安全策略

手工方式配置的安全策略组只能应用到一个接口上。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ipsec policy** *policy-name*，在接口上应用安全策略组。

一个接口只能应用一个安全策略组，采用 IKE 自动协商配置的一个安全策略组可以应用到多个接口上。但手工方式配置的安全策略组只能应用到一个接口。如果所应用的安全策略是手工方式建立安全联盟，会立即生成安全联盟。

----结束

4.3.6 检查配置结果

配置采用 Manual 方式协商的 IPSec 隧道完成之后，可以查看安全联盟、安全提议和安全策略的信息。

前提条件

已完成采用 Manual 方式协商的 IPSec 隧道的所有配置。

操作步骤

- 执行 **display ipsec sa** 命令查看安全联盟的相关信息。
- 执行 **display ipsec proposal** [**name** *proposal-name*]命令查看安全提议的信息。
- 执行 **display ipsec policy** [**brief** | **name** *policy-name* [*seq-number*]]命令查看安全策略的信息。

----结束

4.4 配置采用 IKE 方式协商的 IPSec 隧道

IKE 具有一套自保护机制，可以在不安全的网络上安全地分发密钥、认证身份、建立 IPSec 安全联盟。

4.4.1 建立配置任务

在配置采用 IKE 协商方式建立的 IPSec 隧道之前，了解其应用环境，以及需要提前完成的任务和准备的数据。

应用环境

网络拓扑结构复杂时，采用 IKE 协商方式建立 IPSec 隧道。

前置任务

在配置采用 IKE 方式协商的 IPSec 之前，需要完成以下任务：

- 配置接口的链路层协议参数（和 IP 地址），使接口的链路协议状态为 Up。

- 配置路由，保证 IPSec 服务发起端设备的接口 IP 地址和 IPSec 服务终结端设备的接口 IP 地址之间路由可达。

数据准备

在配置采用 IKE 方式协商的 IPSec 之前，需要准备以下数据。

序号	数据
1	高级 ACL 相关参数
2	IKE 安全提议的优先级、IKE 协商使用的加密算法、IKE 协商使用的认证算法、IKE 协商使用的认证方法、Diffie-Hellman 组标识、安全联盟生存周期
3	IKE peer 名称、协商模式、IKE 安全提议名称、IKE Peer 的 ID 类型、与对端共享的 pre-shared key、IKE Peer 对端的地址、（可选）IPSec 隧道绑定的 VPN 实例、对端名称
4	安全提议名称、使用的安全协议、AH 协议的认证算法、ESP 协议的认证算法、ESP 协议的加密算法、报文封装形式
5	安全策略名称和序列号、（可选）协商时使用的 PFS 特性
6	（可选）安全策略模板名称
7	可选：安全策略组的本端地址、以时间为基准的全局生存周期的时间、或以流量为基准的全局生存周期的流量、发送 Keepalive 报文的时间间隔、等待 Keepalive 报文的超时时间、发送 NAT 更新报文的时间间隔
8	应用安全策略的接口类型和编号

说明

可根据实际应用，选择 AH 或 ESP 协议的一种进行相应的数据规划。

4.4.2 创建需要保护的数据流

IPSec 能够对不同的数据流进行安全保护。实际应用中，需要首先通过 ACL 定义数据流，再在安全策略中引用该 ACL，从而起到保护该数据流的作用。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **acl** [**number**] *acl-number* [**match-order** { **config** | **auto** }]，创建一个高级访问控制列表并进入其视图。

步骤 3 执行命令 **rule**，在 ACL 视图下，配置访问控制规则。

说明

- 需要精确配置 ACL。建议只对确实需要 IPSec 保护的数据流进行保护，即配置对应的 ACL 规则的动作关键字为 permit。
- 对于有不同安全要求的数据流，需要创建不同的 ACL 和相应的安全策略。

----结束

4.4.3 （可选）配置 IKE 安全提议

用户可以按照优先级创建多条 IKE 提议，但是协商双方必须至少有一条匹配的 IKE 提议才能协商成功。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ike proposal** *proposal-number*，创建 IKE 安全提议并进入 IKE 安全提议视图。

协商双方所引用的 IKE 安全提议必须匹配，才能协商成功。

步骤 3 （可选）执行命令 **encryption-algorithm** { **des-cbc** | **3des-cbc** | **aes-cbc-128** | **aes-cbc-192** | **aes-cbc-256** }，选择加密算法。

缺省情况下，使用 des-cbc 加密算法。

步骤 4 （可选）执行命令 **authentication-method** { **pre-share** | **rsa-signature** }，设置一个 IKE 提议使用的认证方法。

缺省情况下，使用 pre-shared key 的认证方法。

步骤 5 （可选）执行命令 **authentication-algorithm** { **md5** | **sha1** | **aes_xcbc_mac_96** }，选择认证算法。

缺省情况下，使用 SHA1 认证算法。

步骤 6 （可选）执行命令 **dh** { **group1** | **group2** | **group5** | **group14** }，选择 Diffie-Hellman 组标识。

步骤 7 （可选）执行命令 **prf** { **hmac-md5** | **hmac-sha1** | **aes_xcbc_128** } ，配置伪随机数产生函数的算法。

步骤 8 （可选）执行命令 **sa duration** *interval*，设置 SA 的存活时间。

如果 **duration** 时间超时，IKE SA 将自动更新。

生存周期只对通过 **isakmp** 方式建立的该安全联盟有效，对通过 **manual** 方式建立的安全联盟没有生存周期的限制，即手工建立的安全联盟永远不会失效。

----结束

4.4.4 配置 IKE Peer

配置 IKE 对等体的一系列属性，包括：IKE 协商所使用的版本、IKE 协商使用的 ID 类型、对端的 IP 地址或对端的名称、预共享密钥值、是否需要进行 NAT 穿越，针对 IKEv1，还需配置选用主模式还是野蛮模式进行 IKE 协商。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ike peer** *peer-name* [**v1** | **v2**]，创建 IKE Peer 并进入 IKE Peer 视图。

步骤 3 （可选）执行命令 **exchange-mode** { **main** | **aggressive** }，配置协商模式。

在野蛮模式下使用步骤 5**local-id-type** 命令配置本地 ID 类型为 **ip** 或 **name**，主模式下只能配置本端 IP 地址。

如果定义的 IKE 对等体，使用的是 IKE v2 版本，则不需要配置协商模式，即不用执行该步骤。

步骤 4 （可选）执行命令 **ike-proposal** *proposal-number*，配置 IKE 安全提议。

步骤 5 （可选）执行命令 **local-id-type** { **ip** | **name** }，配置 IKE Peer 的 ID 类型。
缺省情况下，本地 ID 类型为 IP 地址形式。

步骤 6 （可选）执行命令 **local-address** *address*，配置 IKE 本端 IP 地址。

缺省情况下，本端 IP 地址是绑定此 IPSec 策略的接口地址。

步骤 7 （可选）执行命令 **peer-id-type** { **ip** | **name** }，配置远端 IKE Peer 的 ID 类型。

缺省情况下，ID 类型为 IP 地址形式。本命令只在 IKE V2 版本时配置才有效。

步骤 8 （可选）执行命令 **nat traversal**，配置是否需要进行 NAT 穿越。

配置 NAT 穿越时，需要配置 **local-id-type name**。

步骤 9 （可选）执行命令 **pre-shared-key** *key-string*，配置与对端共享的 pre-shared key。

如果选择了 pre-shared key 验证方法，需要为每个对端配置预共享密钥。建立安全连接的两个对端的预共享密钥必须一致。

使用 pre-shared key 的验证方法时必须配置验证字。

步骤 10 执行命令 **remote-address** { *ip-address* | *host-name* }，配置对端的 IP 地址或域名。

说明

IPSec 安全策略模板方式下，不需要执行该命令。

步骤 11 （可选）执行命令 **sa binding vpn-instance** *vpn-instance-name*，配置 VPN 实例与安全联盟关联。

通过配置 IPSec 隧道对端所属的 VPN 的方式可以实现 IPSec 的多实例连接。这种配置方式只对隧道的发起方有意义。当隧道的发起方发送报文时，需要知道报文的发送接口。通过配置该命令指定隧道对端所属的 VPN，从而知道报文的发送接口，并将报文发送出去。而对于接收方来说，接收的报文已经具有 VPN 属性，即使不配置该命令，也能够成功接收报文。

步骤 12 （可选）执行命令 **remote-name** *name*，配置对端名称（只在野蛮模式下且使用名字认证时使用）。

如果是 IKEv2 版本，**local-id-type** 为 *ip*，**peer-id-type** 为 *name*，则也要指定 **remote-name**。

步骤 13 （可选）执行命令 **inband ocsp**，指定通过 IKE 协议承载 OCSP 请求和 OCSP 响应进行在线证书状态认证。

步骤 14 （可选）执行命令 **pki realm** *realm-name*，在 IKE 对等体上指定 PKI 域。

在 IKE 对等体上指定 PKI 域，可以根据 PKI 域下的配置信息获取本端的 CA 证书和设备证书。

步骤 15 执行命令 **quit**，返回系统视图。

步骤 16 （可选）执行命令 **ike local-name** *local-name*，设置 IKE 协商时的本机名称。

执行命令 **local-id-type** 配置本地 ID 类型为 **name** 时，需要配置本机 ID。

----结束

4.4.5 配置 IPSec 安全提议

隧道两端的设备，安全协议、认证算法、加密算法、报文封装格式需要配置相同。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec proposal** *proposal-name*，创建安全提议并进入安全提议视图。

步骤 3 （可选）执行命令 **transform** { **ah** | **esp** | **ah-esp** }，配置安全协议。

缺省情况下，IPSec 提议采用的安全协议为 RFC2406 规定的 ESP 协议。

步骤 4 （可选）执行命令 **ah authentication-algorithm** { **md5** | **sha1** | **sha2-256** | **sha2-384** | **sha2-512** }，设置 AH 协议采用的认证算法。

缺省情况下，在 IPSec 提议中 AH 协议采用 MD5 认证。

步骤 5 （可选）执行命令 **esp authentication-algorithm** [**md5** | **sha1** | **sha2-256** | **sha2-384** | **sha2-512**]，设置 ESP 协议采用的认证算法。

缺省情况下，采用 MD5 认证算法。

步骤 6 （可选）执行命令 **esp encryption-algorithm** { **3des** | **des** | **aes-128** | **aes-192** | **aes-256** }，设置 ESP 协议采用的加密算法。

缺省情况下，采用 DES 加密算法。

步骤 7 （可选）执行命令 **encapsulation-mode** { **transport** | **tunnel** }，选择报文封装形式。

缺省情况下，安全协议对 IP 报文的封装形式采用隧道模式。

----结束

4.4.6 配置 IPSec 安全策略

配置完成的 IKE Peer 需要在 IPSec 的安全策略视图下引用，才能自动协商 IPSec。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec policy** *policy-name seq-number* **isakmp** [**template** *template-name*]，创建安全策略。

步骤 3 执行命令 **proposal** *proposal-name*，引用 IPSec 安全提议。

IKE 协商方式的安全策略最多可以引用六个安全提议。隧道两端进行 IKE 协商时将在安全策略中最先引用能够完全匹配的安全提议。

步骤 4 执行命令 **security acl** *acl-number*，引用 ACL。

步骤 5 （可选）执行命令 **sa trigger-mode** { **auto** | **traffic-based** }，配置 IPSec SA 的触发方式。

在 IKE 第一阶段协商成功后，会根据触发方式去协商 IPSec SA。如果配置成自动触发方式，则 IKE 第一阶段协商成功后就会开始协商 IPSec SA；如果配置成基于流量的触发方式，则需要有报文的时候才会触发协商 IPSec SA。

缺省情况下为自动触发方式。

步骤 6 （可选）执行命令 **sa duration** { **traffic-based** *kilobytes* | **time-based** *interval* }，配置 IPSec SA 的生存周期。

- 当使用 IKE V1 版本时，对等体两端协商使用本地设置的生存周期和对端提议的生存周期中较小的一个，作为安全联盟的生存周期。
- 当使用 IKE V2 版本时，对等体两端不协商安全联盟的生存周期。安全联盟的生存周期由本地设置的参数决定。
- IPSec SA 的生存周期时间的缺省值为 3600s，流量的缺省值为 1843200 kilobytes。

步骤 7 执行命令 **ike-peer** *peer-name*，引用 IKE Peer。

说明

配置 IKE 对等体的步骤请参见 **4.4.4 配置 IKE Peer**。

步骤 8 （可选）执行命令 **pfs** { **dh-group1** | **dh-group2** | **dh-group5** | **dh-group14** }，设置协商时使用的 PFS 特性。

如果本端指定了 PFS，对端在发起协商时必须是 PFS 交换。本端和对端指定的 DH 组必须一致，否则协商会失败。如果对端是模板方式的时候，DH 组可以不一致。

----结束

4.4.7 配置 IPSec 安全策略模板

配置安全策略模板可以简化多个 IPSec 隧道建立时的配置工作量。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec policy-template** *policy-template-name seq-number*，创建安全策略模板。

步骤 3 （可选）执行命令 **security acl** *acl-number*，引用 ACL。

步骤 4 执行命令 **proposal** *proposal-name*，引用安全提议。

IKE 协商方式的安全策略最多可以引用六个安全提议。隧道两端进行 IKE 协商时将在安全策略中最先引用能够完全匹配的安全提议。

步骤 5 （可选）执行命令 **sa duration** { **traffic-based** *kilobytes* | **time-based** *interval* }，配置 IPSec SA 的生存周期。

步骤 6 执行命令 **ike-peer** *peer-name*，引用 IKE Peer。

步骤 7 （可选）执行命令 **pfs** { **dh-group1** | **dh-group2** | **dh-group5** | **dh-group14** }，设置协商时使用的 PFS 特性。

缺省情况下，安全策略发起协商时未采用完善的前向安全 PFS（Perfect Forward Secrecy）特性。

----结束

4.4.8 （可选）配置其它可选参数

生存周期只对通过 isakmp 方式建立的安全联盟有效，对通过 manual 方式建立的安全联盟没有生存周期的限制。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec sa global-duration** { **time-based** *interval* | **traffic-based** *kilobytes* }，设置全局的安全联盟生存周期。

生存周期只对通过 **isakmp** 方式建立的安全联盟有效，对通过 **manual** 方式建立的安全联盟没有生存周期的限制，即手工建立的安全联盟永远不会失效。

如果没有单独为某安全策略设置安全联盟生存周期，则采用设定的全局生存周期。

改变全局生存周期，不会影响单独配置了自己的生存周期的安全策略，也不会影响已经建立的安全联盟，但没有单独配置生存周期的安全联盟在下次 IKE 协商中会使用新的全局生存周期。

步骤 3 执行命令 **ike heartbeat-timer interval** *interval*，配置发送 heartbeat 报文的时间间隔。

步骤 4 执行命令 **ike heartbeat-timer timeout** *interval*，配置等待 heartbeat 报文的超时时间。

配置 **ike heartbeat-timer timeout** 和 **ike heartbeat-timer interval** 时，interval 和 timeout 要成对出现，即在一个设备上配置了 timeout 参数，在对端就要配置 interval 参数。

在网络上一般不会出现超过连续三次的报文丢失，所以可以配置 timeout 为对端配置的 interval 的三倍，而不应该与本端进行比较。

步骤 5 执行命令 **ike nat-keepalive-timer interval** *interval*，配置发送 NAT 保活报文的时间间隔。

步骤 6 执行命令 **ipsec anti-replay** { **enable** | **disable** }，配置抗重放功能。

步骤 7 执行命令 **ipsec df-bit** { **clear** | **set** | **copy** }，配置 IPSec 隧道的 DF 标志位。

步骤 8 执行命令 **ipsec fragmentation before-encryption**，配置 IPSec 隧道报文的分片方式。

步骤 9 执行命令 **ike peer**,进入 IKE 对等体视图。

步骤 10 执行命令 **local-address** *address*，配置 IKE 本端 IP 地址。

步骤 11 执行如下命令，进行对等体存活检测 DPD（Dead peer detection）相关配置。

- 执行命令 **dpd** { **idle-time** *seconds* | **retransmit-interval** *seconds* | **retry-limit** *times* }，配置对等体存活检测空闲时间、DPD 报文重传间隔和重传次数。
- 执行命令 **dpd msg** { **seq-hash-notify** | **seq-notify-hash** }，配置 DPD 报文中的载荷顺序。
- 执行命令 **dpd type** { **on-demand** | **periodic** }，配置 DPD 检测模式。

----结束

4.4.9 （可选）配置路由注入

如果 IPSec 隧道 Up，IPSec 对等体的路由可以添加到本地并向外发布；如果 IPSec 隧道 Down，IPSec 对等体的路由可以从本地删除并撤销发布，实现网络流量的选路与 IPSec 隧道状态相关联。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec policy** *policy-name seq-number* **isakmp**，进入非手动配置的 IPSec 策略视图。

步骤 3 执行命令 **route inject** { **static** | **dynamic** } [**preference** *preference*]，使能路由注入功能。

缺省情况下，系统未使能路由注入功能。

----结束

4.4.10 应用安全策略

一个接口只能应用一个安全策略组，采用 IKE 自动协商配置的一个安全策略组可以应用到多个接口上。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 3 执行命令 **ipsec policy** *policy-name*，在接口上应用 IPSec 安全策略组。

一个接口只能应用一个安全策略组，一个安全策略组可以应用到多个接口上。

配置完上述步骤后，IPSec 隧道两端设备之间发送的报文，将触发 IKE 进行协商建立安全联盟。如果配置为自动协商方式，则 IKE 协商成功后就会生成 SA；如果配置为流量触发方式，则只有当符合某 IPSec 安全策略的数据流从该接口外出时，才会触发 IKE 去协商 IPSec 安全联盟。IKE 协商成功并创建了安全联盟后，两端设备间的数据流将被加密传输。

----结束

4.4.11 检查配置结果

配置采用 IKE 协商方式建立的 IPSec 隧道完成之后，可以查看安全通道信息、IKE Peer 的配置情况、IKE 提议配置的参数。

前提条件

已完成采用 IKE 方式协商的 IPSec 隧道的所有配置。

操作步骤

- 执行 **display ike sa** [**v2**] [**conn-id** *connid* | **peer-name** *peername* | **phase** *phase-number* | **verbose**]命令,查看当前由 IKE 建立的安全隧道。

- 执行 **display ike peer** [**name** *peer-name*] [**verbose**]命令,查看 IKE 对等体的配置情况。
- 执行 **display ike proposal** 命令,查看每个 IKE 提议配置的参数。
- 执行 **display ipsec sa** [**brief** | **duration** | **policy** *policy-name* [*seq-number*] | **peerip** *peer-ip-address*]命令,查看当前安全联盟的相关信息。
- 执行 **display ipsec policy** [**brief** | **name** *policy-name* [*seq-number*]]命令,查看 IPSec 策略的信息。
- 执行 **display ipsec proposal** [**name** *proposal-name*]命令,查看安全提议的信息。

----结束

4.5 配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道

配置 IPSec 虚拟隧道接口，并在 IPSec 虚拟隧道接口视图下应用安全框架使得 IPSec 虚拟隧道的配置生效。

4.5.1 建立配置任务

在配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道之前，了解其应用环境，以及配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道需要提前完成的任务和准备的数据。

应用环境

为简化 IPSec 策略管理的复杂度，提供了 IPSec 安全框架功能。在 IPSec 虚拟隧道接口下应用安全框架后只会生成一条 IPSec 隧道，并对所有路由到该隧道接口的数据流进行 IPSec 保护。

前置任务

在配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道之前，需要完成以下任务：

- 配置接口的链路层协议参数（和 IP 地址），使接口的链路协议状态为 Up。
- 配置路由，保证 IPSec 服务发起端设备的接口 IP 地址和 IPSec 服务终结端设备的接口 IP 地址之间路由可达。

数据准备

在配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道之前，需要准备以下数据。

序号	数据
1	安全提议名称、使用的安全协议、AH 协议的认证算法、ESP 协议的认证算法、ESP 协议的加密算法、报文封装形式、协商时使用的 PFS 特性
2	IKE peer 名称、协商模式、IKE 安全提议名称、IKE Peer 的 ID 类型、与对端共享的 pre-shared key
3	SA 的生存周期、全局 SA 的生存周期
4	Tunnel 接口的编号、Tunnel 接口的 IP 地址、Tunnel 接口的源地址和目的地址

序号	数据
5	应用安全框架的接口编号

4.5.2 配置安全框架

为简化 IPSec 策略管理的复杂度，系统提供 IPSec 安全框架功能。

背景信息

安全框架定义了 IKE 对等体、IPSec 安全提议、安全联盟的生存周期以及 PFS。为保证 IKE 协商成功，安全框架中所有配置的参数必须在本端和对端相匹配。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec profile** *profile-name*，创建一个安全框架，并进入安全框架视图。

安全框架只可应用于 Tunnel 接口。

步骤 3 执行命令 **proposal** *proposal-name*，配置安全框架引用的安全提议。

一个安全框架下最多可以引用十二个安全提议。缺省情况下，安全框架没有引用任何安全提议。

说明

配置 IPSec 安全提议的步骤请参见 **4.4.5 配置 IPSec 安全提议**。

步骤 4 执行命令 **ike-peer** *peer-name*，在安全框架中引用 IKE 对等体。

缺省情况下，安全框架没有引用 IKE 对等体。

说明

配置 IKE 对等体的步骤请参见 **4.4.4 配置 IKE Peer**。

步骤 5 （可选）执行命令 **pfs** { **dh-group1** | **dh-group2** | **dh-group5** | **dh-group14** }，配置使用此安全框架发起协商时使用的 PFS 特性。

缺省情况下，安全框架发起协商时没有使用 PFS 特性。

步骤 6 （可选）执行命令 **sa duration** { **traffic-based** *kilobytes* | **time-based** *seconds* }，配置 SA 的生存周期。

步骤 7 执行命令 **quit**，返回系统视图。

步骤 8 （可选）执行命令 **ipsec sa global-duration** { **time-based** *seconds* | **traffic-based** *kilobytes* }，配置全局 SA 的生存周期。

缺省情况下，SA 基于时间的全局生存周期为 3600 秒，基于流量的全局生存周期为 1843200 千字节。

----结束

4.5.3 配置 IPSec 虚拟隧道接口

IPSec 虚拟隧道接口下应用安全框架。

背景信息

IPSec 虚拟隧道接口就是采用 IPSec 协议对报文进行封装的隧道接口，通过配置 IPSec 虚拟隧道接口，并在 IPSec 虚拟隧道接口视图下应用安全框架使得 IPSec 虚拟隧道的配置生效。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，进入 Tunnel 接口视图。

步骤 3 执行命令 **ip address**，配置 Tunnel 接口的 IPv4 私网地址。

步骤 4 执行命令 **tunnel-protocol** { **gre** [**p2mp**] | **ipsec** | **ipv4-ipv6** | **none** }，配置隧道封装模式。

说明

隧道接口的封装模式根据实际需要设置为 IPSec、GRE 或者 MGRE 方式，才能在 Tunnel 口下绑定 IPSec 安全框架。

步骤 5 执行命令 **source** { [**vpn-instance** *vpn-instance-name*] *source-ip-address* | *interface-type interface-number* }，配置 Tunnel 接口的源地址。

说明

建议配置 **source** 地址时指定为接口类型。如果直接指定 IP 地址，如果该 IP 地址是某接口动态获取到的，则在 IPSec 配置恢复时，会因为该地址不存在而导致 IPSec 配置恢复失败。

步骤 6 （可选）执行命令 **destination** *dest-ip-address*，配置 Tunnel 接口的目的地址。

如果隧道接口的封装模式设置为 IPSec 方式，则只需要 IKE 对等体的一端配置目的地址即可；如果隧道接口的封装模式设置为 GRE 方式，则设备两端都需要设置目的地址。

步骤 7 执行命令 **ipsec profile** *profile-name*，在 Tunnel 接口上应用安全框架。

缺省情况下，接口上没有应用的安全框架。

----结束

4.5.4 检查配置结果

配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道完成之后，可以查看 IPSec 安全框架、安全提议、安全联盟以及 Tunnel 接口下的配置信息。

前提条件

已完成采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道的所有配置。

操作步骤

- 执行 **display ipsec profile** [**brief** | **name** *profile-name*]命令,查看 IPSec 安全框架的信息。
- 执行 **display ipsec proposal** [**name** *proposal-name*] 命令,查看安全提议的信息。

- 执行 **display ipsec sa** [**brief** | **duration** | **policy** *policy-name* [*seq-number*] | **profile** *profile-name* | **peerip** *peer-ip-address*] 命令,查看安全联盟的相关信息。
- 在接口视图下执行 **display this** 命令,查看 Tunnel 接口下的配置信息。

----结束

4.6 配置采用 Efficient VPN 策略建立 IPSec 隧道

部署 Efficient VPN 可以简化远程设备的配置，将管理员从繁琐的 IPSec 配置中解脱出来。

4.6.1 建立配置任务

在配置 Efficient VPN 策略之前，了解其应用环境，以及配置 Efficient VPN 需要提前完成的任务和准备的数据。

应用环境

两个对等体之间建立 IPSec 隧道，必须在两个对等体上做大量的 IPSec 配置。包括配置 IKE 协商认证算法、IKE 协商加密算法、Diffie-Hellman、IPSec Proposal 等。在包含数百个站点的大型网络场景中，Remote 设备上的 IPSec 配置将非常复杂。而 Efficient VPN 方案，方便远程分支接入企业总部，将管理员从繁琐的 IPSec 配置中解脱出来。

前置任务

在配置 Efficient VPN 之前，需要完成以下任务：

- 配置接口的链路层协议参数，使接口的链路协议状态为 Up。
- 配置路由，保证 IPSec 服务发起端设备的接口 IP 地址和服务终结端设备接口的 IP 地址之间路由可达。

数据准备

在配置 Efficient VPN 策略之前，需要准备以下数据。

序号	数据
1	高级 ACL 相关参数
2	IKE 安全提议名称、IKE 安全提议的优先级、IKE 协商使用的加密算法、IKE 协商使用的认证算法、IKE 协商使用的认证方法、Diffie-Hellman 组标识、安全联盟生存周期、IPSec Proposal 的名称
3	安全策略名称和序列号、安全策略模板名称和序列号
4	DNS 服务器地址、WINS 服务器地址、全局地址池下可分配的网段地址
5	IKE 本端地址、对端 IKE Peer 的地址、对端名称

4.6.2 配置 client 模式

Efficient VPN Client 模式对做 NAT 后的数据流进行安全保护。

背景信息

Remote 设备上仅需配置接入 Server 端的 IP 地址、预共享密钥等 IPSec 隧道必须参数。IKE 协商认证算法、IKE 协商加密算法、IPSec Proposal 等大部分 IPSec 参数在 Sever 端上进行配置。

操作步骤

步骤 1 在 Remote 端的路由器上进行如下配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **ipsec efficient-vpn** *efficient-vpn-name* **mode client**，创建一个 client 模式的 IPSec Efficient VPN 策略，并进入 Efficient VPN 策略视图。
3. 执行命令 **remote-address** { *ip-address* | *host-name* } { **v1** | **v2** }，配置对端 IKE 对等体的 IP 地址或域名。
4. （可选）执行命令 **remote-name** *name*，配置对端 IKE 对等体名称。
5. （可选）执行命令 **authentication-method** { **pre-share** | **rsa-signature** }，配置一个 IKE 提议使用的认证方法。
6. （可选）执行命令 **pre-shared-key** *key*，配置 pre-shared key 认证方法的认证字。
7. 执行命令 **quit**，返回系统视图。
8. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
9. 执行命令 **ipsec efficient-vpn(接口视图)** *efficient-vpn-name*，接口上应用 Efficient VPN 策略。

步骤 2 在 Server 的路由器上进行如下配置。

1. 执行命令 **system-view**，进入系统视图。
2. 执行命令 **ip pool** *ip-pool-name*，创建一个全局地址池。
3. 执行命令 **network** *ip-address* [**mask** { *mask* | *mask-length* }]，配置全局地址池下可分配的网段地址。
4. 执行命令 **quit**，返回系统视图。
5. 执行命令 **aaa**，进入 AAA 视图。
6. 执行命令 **service-scheme** *service-scheme-name*，创建一个业务方案，并进入业务方案视图。
7. （可选）执行命令 **dns** *ip-address*，配置业务方案使用的主 DNS 服务器。
8. （可选）执行命令 **dns** *ip-address* **secondary**，配置业务方案使用的备 DNS 服务器。
9. 执行命令 **ip-pool** *pool-name* [**move-to** *new-position*]，设置 AAA 业务方案下的 IP 地址池下已配置的地址池的位置。。
10. （可选）执行命令 **wins** *ip-address*，配置业务方案使用的主 WINS 服务器。
11. （可选）执行命令 **wins** *ip-address* **secondary**，配置业务方案使用的备 WINS 服务器。
12. 执行命令 **quit**，返回 AAA 视图。
13. 执行命令 **quit**，返回系统视图。

14. 执行命令 **ike proposal** *proposal-number*，创建 IKE 安全提议。

    配置 IKE 安全提议的步骤请参见 **4.4.3 （可选）配置 IKE 安全提议**。

    说明

    Efficient VPN 策略中，IKE 阶段密钥协商时所使用的 DH 密钥交换参数必须为 **dh group2**。
15. 执行命令 **quit**，返回系统视图。
16. 执行命令 **ike peer** *peer-name* { **v1** | **v2** }，创建一个 IKE 对等体。

    配置 IKE Peer 的步骤请参见 **4.4.4 配置 IKE Peer**。

    说明

    - 使用 IKE v1 版本时，exchange-mode 必须设置为 **aggressive** 方式。
    - IKE 对等体下需要执行命令 service-scheme，绑定已创建的 AAA 业务方案。
17. 执行命令 **quit**，返回系统视图。
18. 执行命令 **ipsec proposal** *proposal-name*，创建安全提议。

    配置 IPSec 安全提议的步骤请参见 **4.4.5 配置 IPSec 安全提议**。

    说明

    - Efficient VPN 策略 **encapsulation-mode** 必须设置为 **tunnel** 模式。
    - Efficient VPN 策略仅支持 ESP 安全协议。
19. 执行命令 **quit**，返回系统视图。
20. 执行命令 **ipsec policy-template** *template-name seq-number*，创建一个安全策略模板。

    配置 IPSec 安全策略模板的步骤请参见 **4.4.7 配置 IPSec 安全策略模板**。
21. 执行命令 **quit**，返回系统视图。
22. 执行命令 **ipsec policy** *policy-name seq-number* **isakmp template** *template-name*，指定采用策略模板创建安全联盟。

    配置 IPSec 安全策略的步骤请参见 **4.4.6 配置 IPSec 安全策略**。
23. 执行命令 **quit**，返回系统视图。
24. 执行命令 **interface** *interface-type interface-number*，进入接口视图。
25. 执行命令 **ipsec policy** *policy-name*，在接口上应用指定的安全策略组。

----结束

4.6.3 配置 network 模式

Efficient VPN Network 模式下对符合 ACL 特征的数据流进行安全保护。

背景信息

在 Remote 端和 Server 端的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **acl** [**number**] *acl-number* [**match-order** { **config** | **auto** }]，创建一个高级访问控制列表并进入其视图。

步骤 3 执行命令 **rule**，在 ACL 视图下，配置访问控制规则。

说明

引用 ACL，rule 规则只可以匹配 IP 报文，即 **permit ip**。

步骤 4 执行命令 **quit**，返回系统视图。

步骤 5 执行命令 **ipsec efficient-vpn** *efficient-vpn-name* **mode network**，创建一个 network 模式的 IPSec Efficient VPN 策略，并进入 Efficient VPN 策略视图。

步骤 6 执行命令 **security acl** *acl-number*，引用 ACL。

步骤 7 执行命令 **remote-address** { *ip-address* | *host-name* } { **v1** | **v2** }，配置对端 IKE 对等体的 IP 地址或域名。

步骤 8 （可选）执行命令 **remote-name** *name*，配置对端 IKE Peer 的名称。

步骤 9 （可选）执行命令 **authentication-method** { **pre-share** | **rsa-signature** }，配置一个 IKE 提议使用的认证方法。

缺省情况下，使用的认证方法为 pre-shared key。

步骤 10 （可选）执行命令 **pre-shared-key** *key*，配置 pre-shared key 认证方法的认证字。

缺省情况下，pre-shared key 认证方法没有认证字。

步骤 11 （可选）执行命令 **pfs** { **dh-group1** | **dh-group2** | **dh-group5** | **dh-group14** }，设置在协商时使用 PFS 特性。

步骤 12 （可选）执行命令 **pki realm** *realm-name*，指定 PKI 域。

步骤 13 （可选）执行命令 **sa binding vpn-instance** *vpn-instance-name*，指定 IPSec 隧道绑定的 VPN 实例。

说明

执行该命令前，需要先配置 VPN 实例。

步骤 14 （可选）执行命令 **local-id-type** { **ip** | **name** }，指定 IKE 的 ID 类型。

缺省情况下，ID 类型为 IP 地址形式。

步骤 15 （可选）执行命令 **local-address** *address*，配置 IKE 本端 IP 地址。

步骤 16 执行命令 **quit**，返回系统视图。

步骤 17 （可选）执行命令 **aaa**，进入 AAA 视图。

步骤 18 （可选）执行命令 **service-scheme** *service-scheme-name*，创建一个业务方案，并进入业务方案视图。

步骤 19 （可选）执行命令 **dns** *ip-address*，配置业务方案使用的主 DNS 服务器。

步骤 20 （可选）执行命令 **dns** *ip-address* **secondary**，配置业务方案使用的备 DNS 服务器。

步骤 21 （可选）执行命令 **wins** *ip-address*，配置业务方案使用的主 WINS 服务器。

步骤 22 （可选）执行命令 **wins** *ip-address* **secondary**，配置业务方案使用的备 WINS 服务器。

步骤 23 （可选）执行命令 **quit**，返回 AAA 视图。

步骤 24 （可选）执行命令 **quit**，返回系统视图。

步骤 25 执行命令 **interface** *interface-type interface-number*，进入接口视图。

步骤 26 执行命令 **ipsec efficient-vpn** *efficient-vpn-name*，接口上应用 Efficient VPN 策略。

----结束

4.6.4 检查配置结果

配置 Efficient VPN 完成之后，可以查看安全通道信息、安全提议、安全联盟以及 IPSec Efficient VPN 策略的配置信息。

前提条件

已完成 Efficient VPN 的所有配置。

操作步骤

- 执行 **display ike sa** [**v2**] [**conn-id** *connid* | **peer-name** *peername* | **phase** *phase-number* | **verbose**]命令,查看当前由 IKE 建立的安全隧道。
- 执行 **display ipsec sa** [**brief** | **duration** | **policy** *policy-name* [*seq-number*] | **profile** *profile-name* | **efficient-vpn** *efficient-vpn-name* | **peerip** *peer-ip-address*] 命令,查看安全联盟的相关信息。
- 执行 **display ipsec proposal** [**name** *proposal-name*]命令,查看安全提议的信息。
- 执行 **display ipsec efficient-vpn** [**brief** | **capality** | **name** *efficient-vpn-name*]命令,查看 Efficient VPN 信息。

----结束

4.7 维护 IPSec

显示 IPSec 的配置信息，清除 IPSec 的统计信息。

4.7.1 显示 IPSec 配置

需要查看 IPSec 的基本信息时，可以通过显示命令查看安全联盟的相关信息、已建立的安全通道、IPSec 处理报文的统计信息 IPSec 框架信息以及 Efficient VPN 策略信息。

前提条件

已经完成 IPSec 的所有配置。

操作步骤

- 执行 **display ipsec sa** [**brief** | **duration** | **policy** *policy-name* [*seq-number*] | **profile** *profile-name* | **peerip** *peer-ip-address*] 命令显示 IPSec 安全联盟的相关信息。
- 执行 **display ike sa** [**v2**] [**conn-id** *connid* | **peer-name** *peername* | **phase** *phase-number* | **verbose**]命令显示当前已建立的安全隧道。
- 执行 **display ipsec statistics** { **ah** | **esp** }命令显示 IPSec 处理报文的统计信息。
- 执行 **display ike statistics** { **all** | **msg** | **v1** | **v2** }命令显示 IKE 处理报文的统计信息。
- 执行 **display ipsec profile** [**brief** | **name** *profile-name*]命令显示 IPSec 框架的信息。

- 执行 **display ipsec efficient-vpn** [**brief** | **capality** | **name** *efficient-vpn-name*]命令,显示 Efficient VPN 的信息。

----结束

4.7.2 清除 IPSec 信息

清除 IPSec 和 IKE 的统计信息、安全联盟和 IKE 安全通道。

背景信息

注意

清除信息后，以前的信息将无法恢复，务必仔细确认。

操作步骤

- 在确认需要清除的信息后，请在用户视图下执行 **reset ipsec statistics** { **ah** | **esp** }命令清除 IPSec 的报文统计信息。
- 在确认需要清除的信息后，请在用户视图下执行 **reset ike statistics** { **all** | **msg** }命令清除 IKE 的报文统计信息。
- 在确认需要清除的信息后，请在用户视图下执行 **reset ipsec sa** [**remote** *ip-address* | **policy** *policy-name* [*seq-number*] | **parameters** *dest-address* {**ah** | **esp** } *spi*] 命令清除安全联盟。
- 在确认需要清除的信息后，请在用户视图下执行 **reset ipsec sa profile** *profile-name* 命令清除安全框架生成的 SA。
- 在确认需要清除的信息后，请在用户视图下执行 **reset ipsec sa efficient-vpn** *efficient-vpn-name* 命令清除 Efficient VPN 策略生成的 SA。
- 在确认需要清除的信息后，请在用户视图下执行 **reset ike sa** { **all** | **conn-id** *connection-id* }命令删除当前已建立的安全通道。

----结束

4.8 配置举例

介绍使用 IPSec 提高数据传输安全性的各种示例。

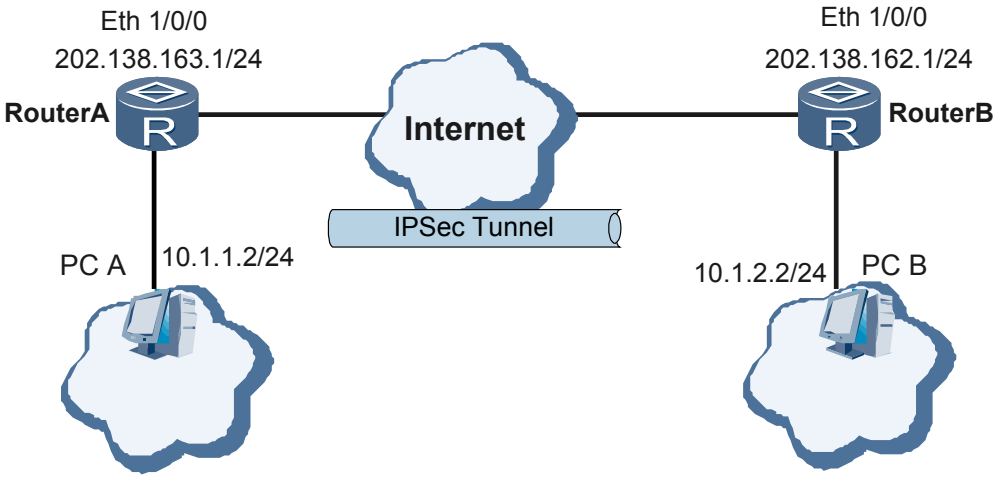
4.8.1 配置采用手工方式建立安全联盟示例

采用手工方式建立安全联盟，一般在实际组网环境比较简单时应用。当网络中节点增多时，手工配置将非常困难，而且难以保证安全性。

组网需求

如图 4-3 所示，在 RouterA 和 RouterB 之间建立一个安全隧道，对 PCA 代表的子网（10.1.1.0/24）与 PCB 代表的子网（10.1.2.0/24）之间的数据流进行安全保护。安全协议采用 ESP 协议，加密算法采用 DES，认证算法采用 SHA1。

图 4-3 配置 IPSec 组网图



配置思路

采用如下思路配置采用手工方式建立安全联盟：

1. 配置接口的 IP 地址。
2. 配置 ACL，以定义要保护的数据流。
3. 配置到对端的静态路由。
4. 配置安全提议。
5. 配置安全策略，并引用 ACL 和安全提议。
6. 在接口上应用安全策略。

操作步骤

步骤 1 分别在 RouterA 和 RouterB 上配置各接口的 IP 地址。

在 RouterA 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 202.138.163.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

在 RouterB 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 202.138.162.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

步骤 2 分别在 RouterA 和 RouterB 上配置访问控制列表，定义各自要保护的数据流。

在 RouterA 上配置访问控制列表。

```
[Huawei] acl number 3101
[Huawei-acl-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
[Huawei-acl-adv-3101] quit
```

在 RouterB 上配置访问控制列表。

```
[Huawei] acl number 3101
[Huawei-acl-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
[Huawei-acl-adv-3101] quit
```

步骤 3 分别在 RouterA 和 RouterB 上配置到对端的静态路由。

在 RouterA 上配置到目的端的静态路由，此处假设到达 PCB 的下一跳地址为 202.138.163.2。

```
[Huawei] ip route-static 10.1.2.0 255.255.255.0 202.138.163.2
```

在 RouterB 上配置到目的端的静态路由，此处假设到达 PCA 的下一跳地址为 202.138.162.2。

```
[Huawei] ip route-static 10.1.1.0 255.255.255.0 202.138.162.2
```

步骤 4 分别在 RouterA 和 RouterB 上创建安全提议。

在 RouterA 上配置安全提议。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] esp authentication-algorithm sha1
[Huawei-ipsec-proposal-tran1] quit
```

在 RouterB 上配置安全提议。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] esp authentication-algorithm sha1
[Huawei-ipsec-proposal-tran1] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec proposal** 会显示所配置的信息，以 RouterA 为例：

```
[Huawei] display ipsec proposal
Number of Proposals: 1

IPSec proposal name: tran1
 Encapsulation mode: Tunnel
 Transform         : esp-new
 ESP protocol      : Authentication SHA1-HMAC-96
                     Encryption     DES
```

步骤 5 分别在 RouterA 和 RouterB 上创建安全策略。

在 RouterA 上配置安全策略。

```
[Huawei] ipsec policy map1 10 manual
[Huawei-ipsec-policy-manual-map1-10] security acl 3101
[Huawei-ipsec-policy-manual-map1-10] proposal tran1
[Huawei-ipsec-policy-manual-map1-10] tunnel remote 202.138.162.1
[Huawei-ipsec-policy-manual-map1-10] tunnel local 202.138.163.1
[Huawei-ipsec-policy-manual-map1-10] sa spi outbound esp 12345
[Huawei-ipsec-policy-manual-map1-10] sa spi inbound esp 54321
[Huawei-ipsec-policy-manual-map1-10] sa string-key outbound esp abcdefg
[Huawei-ipsec-policy-manual-map1-10] sa string-key inbound esp gfedcba
[Huawei-ipsec-policy-manual-map1-10] quit
```

在 RouterB 上配置安全策略。

```
[Huawei] ipsec policy use1 10 manual
[Huawei-ipsec-policyl-manual-use1-10] security acl 3101
[Huawei-ipsec-policyl-manual-use1-10] proposal tran1
[Huawei-ipsec-policyl-manual-use1-10] tunnel remote 202.138.163.1
[Huawei-ipsec-policyl-manual-use1-10] tunnel local 202.138.162.1
[Huawei-ipsec-policyl-manual-use1-10] sa spi outbound esp 54321
[Huawei-ipsec-policyl-manual-use1-10] sa spi inbound esp 12345
[Huawei-ipsec-policyl-manual-use1-10] sa string-key outbound esp gfedcba
[Huawei-ipsec-policyl-manual-use1-10] sa string-key inbound esp abcdefg
```

```
[Huawei-ipsec-policyl-manual-use1-10] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec policy** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec policy
==========================================
IPSec Policy Group: "map1"
Using interface: {}
==========================================

    Sequence number: 10
    Security data flow: 3101
    Tunnel local  address: 202.138.163.1
    Tunnel remote address: 202.138.162.1
    Proposal name:tran1
    Inbound AH setting:
      AH SPI:
      AH string-key:
      AH authentication hex key:
    Inbound ESP setting:
      ESP SPI: 54321 (0xd431)
      ESP string-key: gfedcba
      ESP encryption hex key:
      ESP authentication hex key:
    Outbound AH setting:
      AH SPI:
      AH string-key:
      AH authentication hex key:
    Outbound ESP setting:
      ESP SPI: 12345 (0x3039)
      ESP string-key: abcdefg
      ESP encryption hex key:
      ESP authentication hex key:
```

步骤 6 分别在 RouterA 和 RouterB 的接口上引用各自的安全策略。

在 RouterA 的接口上引用安全策略。

```
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ipsec policy map1
[Huawei-Ethernet1/0/0] quit
```

在 RouterB 的接口上引用安全策略。

```
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ipsec policy use1
[Huawei-Ethernet1/0/0] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec sa** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec sa
==============================
Interface: Ethernet1/0/0
    Path MTU: 1500
==============================

  ----------------------------
  IPSec policy name: "map1"
  Sequence number: 10
  Acl Group: 3101
  Acl rule: 0
  Mode: Manual
  ----------------------------
    Encapsulation mode: Tunnel
    Tunnel local : 202.138.163.1
    Tunnel remote: 202.138.162.1
```

```
[Outbound ESP SAs]
  SPI: 12345 (0x3039)
  Proposal: ESP-ENCRYPT-DES-64 ESP-AUTH-SHA1
  No duration limit for this SA

[Inbound ESP SAs]
  SPI: 54321 (0xd431)
  Proposal: ESP-ENCRYPT-DES-64 ESP-AUTH-SHA1
  No duration limit for this SA
```

步骤 7 检查配置结果

配置成功后，在主机 PCA 上执行 **ping** 操作仍然可以 ping 通主机 PCB，执行命令 **display ipsec statistics esp** 可以查看数据包的统计信息。

----结束

配置文件

- RouterA 的配置文件

```
#
acl number 3101
 rule 5 permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
#
ipsec proposal tran1
 esp authentication-algorithm sha1
#
ipsec policy map1 10 manual
 security acl 3101
 proposal tran1
 tunnel local 202.138.163.1
 tunnel remote 202.138.162.1
 sa spi inbound esp 54321
 sa string-key inbound esp gfedcba
 sa spi outbound esp 12345
 sa string-key outbound esp abcdefg
#
 ip route-static 10.1.2.0 255.255.255.0 202.138.163.2
#
interface Ethernet1/0/0
 ip address 202.138.163.1 255.255.255.0
 ipsec policy map1
#
return
```

- RouterB 的配置文件

```
#
acl number 3101
 rule 5 permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
#
ipsec proposal tran1
 esp authentication-algorithm sha1
#
ipsec policy use1 10 manual
 security acl 3101
 proposal tran1
 tunnel local 202.138.162.1
 tunnel remote
202.138.163.1
 sa spi inbound esp 12345
 sa string-key inbound esp abcdefg
 sa spi outbound esp 54321
 sa string-key outbound esp gfedcba
#
 ip route-static 10.1.1.0 255.255.255.0 202.138.162.2
#
interface Ethernet1/0/0
 ip address 202.138.162.1 255.255.255.0
```

```
 ipsec policy use1
#
return
```

4.8.2 采用默认配置通过 IKE 协商方式建立安全联盟示例

该示例介绍了采用 IKE 协商方式建立安全联盟基本的配置步骤。

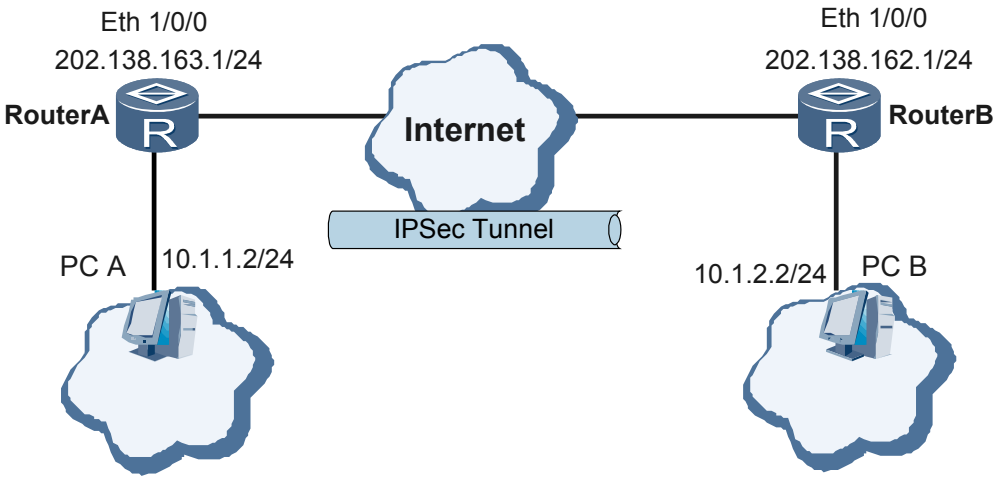
组网需求

如图 4-4 所示，在 RouterA 和 RouterB 之间建立一个安全隧道，对 PC A 代表的子网（10.1.1.0/24）与 PC B 代表的子网（10.1.2.0/24）之间的数据流进行安全保护。安全协议采用 ESP 协议，加密算法采用 DES，认证算法采用 MD5。

说明

- 该示例中没有配置 ike proposal，采用的是系统提供的一条缺省的 IKE 提议。
- 在使用 **ipsec proposal** 命令创建一个新的 IPSec 提议后，其各项参数的缺省值分别为 ESP 协议、DES 加密算法、MD5 认证算法和隧道模式。

图 4-4 配置采用 IKE 协商方式建立安全联盟组网图



配置思路

采用如下思路配置采用 IKE 协商方式建立安全联盟：

1. 配置接口的 IP 地址。
2. 配置 IKE 协商时需要的本机 ID 和 IKE Peer。
3. 配置 ACL，以定义要保护的数据流。
4. 配置到对端的静态路由。
5. 配置安全提议。
6. 配置安全策略，并引用 ACL 和安全提议。
7. 在接口上应用安全策略。

操作步骤

步骤 1 分别在 RouterA 和 RouterB 上配置各接口的 IP 地址。

在 RouterA 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 202.138.163.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

在 RouterB 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 202.138.162.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

步骤 2 分别在 RouterA 和 RouterB 上配置本机 ID 和 IKE Peer。

在 RouterA 上配置进行 IKE 协商时需要的本机 ID 和 IKE Peer。

```
[Huawei] ike peer spub v1
[Huawei-ike-peer-spub] pre-shared-key huawei
[Huawei-ike-peer-spub] remote-address 202.138.162.1
[Huawei-ike-peer-spub] quit
```

说明

野蛮模式中，如果 **local-id-type** 取值为 **name** 的时候，对于发起协商端需要增加 **remote-adress** x.x.x.x 的配置。

在 RouterB 上配置进行 IKE 协商时需要的本机 ID 和 IKE Peer。

```
[Huawei] ike peer spua v1
[Huawei-ike-peer-spua] pre-shared-key huawei
[Huawei-ike-peer-spua] remote-address 202.138.163.1
[Huawei-ike-peer-spua] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ike peer** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ike peer name spub verbose
---------------------------------------
  Peer name                : spub
  Exchange mode            : main on phase 1
  Pre-shared-key           : huawei
  Local ID type            : IP
  DPD                      : Disable
  DPD mode                 : Periodic
  DPD idle time            : 30
  DPD retransmit interval  : 15
  DPD retry limit          : 3
  Host name                :
  Peer Ip address          : 202.138.162.1
  VPN name                 :
  Local IP address         :
  Remote name              :
  Nat-traversal            : Disable
  Configured IKE version   : Version one
  PKI realm                : NULL
  Inband OCSP              : Disable
---------------------------------------
```

步骤 3 分别在 RouterA 和 RouterB 上配置访问控制列表，定义各自要保护的数据流。

在 RouterA 上配置访问控制列表。

```
[Huawei] acl number 3101
```

```
[Huawei-acl-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
[Huawei-acl-adv-3101] quit
```

在 RouterB 上配置访问控制列表。

```
[Huawei] acl number 3101
[Huawei-acl-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
[Huawei-acl-adv-3101] quit
```

步骤 4 分别在 RouterA 和 RouterB 上配置到对端的静态路由。

在 RouterA 上配置到目的端的静态路由，此处假设到达 PCB 的下一跳地址为 202.138.163.2。

```
[Huawei] ip route-static 10.1.2.0 255.255.255.0 202.138.163.2
```

在 RouterB 上配置到目的端的静态路由，此处假设到达 PCA 的下一跳地址为 202.138.162.2。

```
[Huawei] ip route-static 10.1.1.0 255.255.255.0 202.138.162.2
```

步骤 5 分别在 RouterA 和 RouterB 上创建安全提议。

在 RouterA 上配置安全提议。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] quit
```

在 RouterB 上配置安全提议。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec proposal** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec proposal
Number of Proposals: 1

IPSec proposal name: tran1
 Encapsulation mode: Tunnel
 Transform         : esp-new
 ESP protocol      : Authentication MD5-HMAC-96
                     Encryption     DES
```

步骤 6 分别在 RouterA 和 RouterB 上创建安全策略。

在 RouterA 上配置安全策略。

```
[Huawei] ipsec policy map1 10 isakmp
[Huawei-ipsec-policy-isakmp-map1-10] ike-peer spub
[Huawei-ipsec-policy-isakmp-map1-10] proposal tran1
[Huawei-ipsec-policy-isakmp-map1-10] security acl 3101
[Huawei-ipsec-policy-isakmp-map1-10] quit
```

在 RouterB 上配置安全策略。

```
[Huawei] ipsec policy use1 10 isakmp
[Huawei-ipsec-policy-isakmp-use1-10] ike-peer spua
[Huawei-ipsec-policy-isakmp-use1-10] proposal tran1
[Huawei-ipsec-policy-isakmp-use1-10] security acl 3101
[Huawei-ipsec-policy-isakmp-use1-10] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec policy** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec policy
==========================================
```

```
IPSec policy group: "map1"
Using interface: {}
============================================

    Sequence number: 10
    Security data flow: 3101
    Peer name:  spub
    Perfect forward secrecy: None
    Proposal name:  tran1
    IPSec SA local duration(time based): 3600 seconds
    IPSec SA local duration(traffic based): 1843200 kilobytes
    SA trigger mode: Automatic
    Route inject: None
```

步骤 7 分别在 RouterA 和 RouterB 的接口上应用各自的安全策略。

在 RouterA 的接口上引用安全策略。

```
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ipsec policy map1
[Huawei-Ethernet1/0/0] quit
```

在 RouterB 的接口上引用安全策略。

```
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ipsec policy use1
[Huawei-Ethernet1/0/0] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec sa** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec sa
==============================
Interface: Ethernet 1/0/0
    path MTU: 1500
==============================
  ------------------------------
  IPSec policy name: "map1"
  sequence number: 10
  mode: isakmp
  ------------------------------
    Connection id: 3
    encapsulation mode: tunnel
    tunnel local : 202.138.163.1    tunnel remote: 202.138.162.1
    [inbound ESP SAs]
      spi: 1406123142 (0x53cfbc86)
      proposal: ESP-ENCRYPT-DES ESP-AUTH-MD5
      sa remaining key duration (bytes/sec): 1887436528/3575
      max received sequence-number: 4
      udp encapsulation used for nat traversal: N
    [outbound ESP SAs]
      spi: 3835455224 (0xe49c66f8)
      proposal: ESP-ENCRYPT-DES ESP-AUTH-MD5
      sa remaining key duration (bytes/sec): 1887436464/3575
      max sent sequence-number: 5
      udp encapsulation used for nat traversal: N
```

步骤 8 检查配置结果

配置成功后，在主机 PCA 执行 **ping** 操作仍然可以 ping 通主机 PCB，它们之间的数据传输将被加密。

在 RouterA 上执行 **display ike sa** 操作，结果如下。

```
[Huawei] display ike sa
     Conn-ID      Peer           VPN    Flag(s)     Phase
  -------------------------------------------------------------
       14          202.138.162.1    0      RD|ST         1
       16          202.138.162.1    0      RD|ST         2
```

```
Flag Description:
RD--READY    ST--STAYALIVE    RL--REPLACED    FD--FADING    TO--TIMEOUT
HRT--HEARTBEAT    LKG--LAST KNOWN GOOD SEQ NO.    BCK--BACKED UP
```

----结束

配置文件

- RouterA 的配置文件

```
#
acl number 3101
 rule 5 permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
#
ipsec proposal tran1
#
ike peer spub v1
 pre-shared-key huawei
 remote-address 202.138.162.1
#
ipsec policy map1 10 isakmp
 security acl 3101
 ike-peer spub
 proposal tran1
#
 ip route-static 10.1.2.0 255.255.255.0 202.138.163.2
#
interface Ethernet1/0/0
 ip address 202.138.163.1 255.255.255.0
 ipsec policy map1
#
return
```

- RouterB 的配置文件

```
#
acl number 3101
  rule 5 permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
#
ipsec proposal tran1
 esp authentication-algorithm sha1
#
ike peer spua v1
 pre-shared-key huawei
 remote-address 202.138.163.1
#
ipsec policy use1 10 isakmp
 security acl 3101
 ike-peer spua
 proposal tran1
#
 ip route-static 10.1.1.0 255.255.255.0 202.138.162.2
#
interface Ethernet1/0/0
 ip address 202.138.162.1 255.255.255.0
 ipsec policy use1
#
return
```

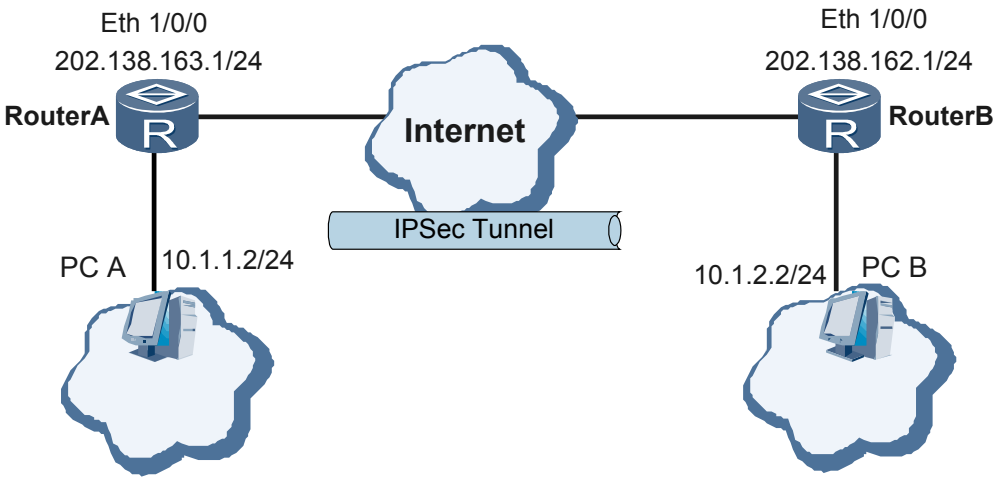
4.8.3 配置采用 IKE 协商方式建立安全联盟示例

使用 IKE 自动地进行安全联盟的建立与密钥的交换，不仅可以大幅度的提升效率，而且可以保证安全性。

组网需求

如图 4-5 所示，在 RouterA 和 RouterB 之间建立一个安全隧道，对 PC A 代表的子网（10.1.1.0/24）与 PC B 代表的子网（10.1.2.0/24）之间的数据流进行安全保护。安全协议采用 ESP 协议，加密算法采用 DES，认证算法采用 SHA1。

图 4-5 配置采用 IKE 协商方式建立安全联盟组网图



配置思路

采用如下思路配置采用 IKE 协商方式建立安全联盟：

1. 配置接口的 IP 地址。
2. 配置 IKE 提议。
3. 配置 IKE 协商时需要的本机 ID 和 IKE Peer。
4. 配置 ACL，以定义要保护的数据流。
5. 配置到对端的静态路由。
6. 配置安全提议。
7. 配置安全策略，并引用 ACL 和安全提议。
8. 在接口上应用安全策略。

操作步骤

步骤 1 分别在 RouterA 和 RouterB 上配置各接口的 IP 地址。

在 RouterA 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 202.138.163.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

在 RouterB 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
```

```
[Huawei-Ethernet1/0/0] ip address 202.138.162.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

步骤 2 分别在 RouterA 和 RouterB 上配置 IKE 提议。

在 RouterA 上配置 IKE 提议。

```
[Huawei] ike proposal 1
[Huawei-ike-proposal-1] encryption-algorithm aes-cbc-128
[Huawei-ike-proposal-1] authentication-algorithm md5
[Huawei-ike-proposal-1] quit
```

在 RouterB 上配置 IKE 提议。

```
[Huawei] ike proposal 1
[Huawei-ike-proposal-1] encryption-algorithm aes-cbc-128
[Huawei-ike-proposal-1] authentication-algorithm md5
[Huawei-ike-proposal-1] quit
```

步骤 3 分别在 RouterA 和 RouterB 上配置本机 ID 和 IKE Peer。

在 RouterA 上配置进行 IKE 协商时需要的本机 ID 和 IKE Peer。

```
[Huawei] ike local-name huawei01
[Huawei] ike peer spub v1
[Huawei-ike-peer-spub] exchange-mode aggressive
[Huawei-ike-peer-spub] ike-proposal 1
[Huawei-ike-peer-spub] local-id-type name
[Huawei-ike-peer-spub] pre-shared-key huawei
[Huawei-ike-peer-spub] remote-name huawei02
[Huawei-ike-peer-spub] remote-address 202.138.162.1
[Huawei-ike-peer-spub] local-address 202.138.163.1
[Huawei-ike-peer-spub] quit
```

说明

野蛮模式中，如果 **local-id-type** 取值为 **name** 的时候，对于发起协商端需要增加 **remote-adress** x.x.x.x 的配置。

在 RouterB 上配置进行 IKE 协商时需要的本机 ID 和 IKE Peer。

```
[Huawei] ike local-name huawei02
[Huawei] ike peer spua v1
[Huawei-ike-peer-spua] exchange-mode aggressive
[Huawei-ike-peer-spua] ike-proposal 1
[Huawei-ike-peer-spua] local-id-type name
[Huawei-ike-peer-spua] pre-shared-key huawei
[Huawei-ike-peer-spua] remote-name huawei01
[Huawei-ike-peer-spua] remote-address 202.138.163.1
[Huawei-ike-peer-spua] local-address 202.138.162.1
[Huawei-ike-peer-spua] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ike peer** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ike peer name spub verbose
----------------------------------------
  Peer name                : spub
  Exchange mode            : aggressive on phase 1
  Pre-shared-key           : huawei
  Proposal                 : 1
  Local ID type            : Name
  DPD                      : Disable
  DPD mode                 : Periodic
  DPD idle time            : 30
  DPD retransmit interval  : 15
  DPD retry limit          : 3
  Host name                :
  Peer Ip address          : 202.138.162.1
  VPN name                 :
```

```
  Local IP address         : 202.138.163.1
  Remote name              : huawei02
  Nat-traversal            : Disable
  Configured IKE version   : Version one
  Auto-configure           : Disable
  PKI realm                : NULL
  Inband OCSP              : Disable
-----------------------------------
```

步骤 4 分别在 RouterA 和 RouterB 上配置访问控制列表，定义各自要保护的数据流。

在 RouterA 上配置访问控制列表。

```
[Huawei] acl number 3101
[Huawei-acl-adv-3101] rule permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
[Huawei-acl-adv-3101] quit
```

在 RouterB 上配置访问控制列表。

```
[Huawei] acl number 3101
[Huawei-acl-adv-3101] rule permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
[Huawei-acl-adv-3101] quit
```

步骤 5 分别在 RouterA 和 RouterB 上配置到对端的静态路由。

在 RouterA 上配置到目的端的静态路由，此处假设到达 PCB 的下一跳地址为 202.138.163.2。

```
[Huawei] ip route-static 10.1.2.0 255.255.255.0 202.138.163.2
```

在 RouterB 上配置到目的端的静态路由，此处假设到达 PCA 的下一跳地址为 202.138.162.2。

```
[Huawei] ip route-static 10.1.1.0 255.255.255.0 202.138.162.2
```

步骤 6 分别在 RouterA 和 RouterB 上创建安全提议。

在 RouterA 上配置安全提议。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] encapsulation-mode tunnel
[Huawei-ipsec-proposal-tran1] transform esp
[Huawei-ipsec-proposal-tran1] esp encryption-algorithm des
[Huawei-ipsec-proposal-tran1] esp authentication-algorithm sha1
[Huawei-ipsec-proposal-tran1] quit
```

在 RouterB 上配置安全提议。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] encapsulation-mode tunnel
[Huawei-ipsec-proposal-tran1] transform esp
[Huawei-ipsec-proposal-tran1] esp encryption-algorithm des
[Huawei-ipsec-proposal-tran1] esp authentication-algorithm sha1
[Huawei-ipsec-proposal-tran1] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec proposal** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec proposal
Number of Proposals: 1

IPSec proposal name: tran1
 Encapsulation mode: Tunnel
 Transform         : esp-new
 ESP protocol      : Authentication SHA1-HMAC-96
                     Encryption     DES
```

步骤 7 分别在 RouterA 和 RouterB 上创建安全策略。

在 RouterA 上配置安全策略。

```
[Huawei] ipsec policy map1 10 isakmp
[Huawei-ipsec-policy-isakmp-map1-10] ike-peer spub
[Huawei-ipsec-policy-isakmp-map1-10] proposal tran1
[Huawei-ipsec-policy-isakmp-map1-10] security acl 3101
[Huawei-ipsec-policy-isakmp-map1-10] quit
```

在 RouterB 上配置安全策略。

```
[Huawei] ipsec policy use1 10 isakmp
[Huawei-ipsec-policy-isakmp-use1-10] ike-peer spua
[Huawei-ipsec-policy-isakmp-use1-10] proposal tran1
[Huawei-ipsec-policy-isakmp-use1-10] security acl 3101
[Huawei-ipsec-policy-isakmp-use1-10] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec policy** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec policy
=========================================
IPSec policy group: "map1"
Using interface: {}
=========================================

    Sequence number: 10
    Security data flow: 3101
    Peer name:  spub
    Perfect forward secrecy: None
    Proposal name:  tran1
    IPSec SA local duration(time based): 3600 seconds
    IPSec SA local duration(traffic based): 1843200 kilobytes
    SA trigger mode: Automatic
    Route inject: None
```

步骤 8 分别在 RouterA 和 RouterB 的接口上应用各自的安全策略。

在 RouterA 的接口上引用安全策略。

```
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ipsec policy map1
[Huawei-Ethernet1/0/0] quit
```

在 RouterB 的接口上引用安全策略。

```
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ipsec policy use1
[Huawei-Ethernet1/0/0] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec sa** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec sa
==============================
Interface: Ethernet 1/0/0
    path MTU: 1500
==============================
  ------------------------------
  IPSec policy name: "map1"
  sequence number: 10
  mode: isakmp
  ------------------------------
    Connection id: 3
    encapsulation mode: tunnel
    tunnel local : 202.138.163.1    tunnel remote: 202.138.162.1
    [inbound ESP SAs]
      spi: 1406123142 (0x53cfbc86)
      proposal: ESP-ENCRYPT-DES ESP-AUTH-SHA1
      sa remaining key duration (bytes/sec): 1887436528/3575
```

```
    max received sequence-number: 4
    udp encapsulation used for nat traversal: N
  [outbound ESP SAs]
    spi: 3835455224 (0xe49c66f8)
    proposal: ESP-ENCRYPT-DES ESP-AUTH-SHA1
    sa remaining key duration (bytes/sec): 1887436464/3575
    max sent sequence-number: 5
    udp encapsulation used for nat traversal: N
```

步骤 9 检查配置结果

配置成功后，在主机 PCA 执行 **ping** 操作仍然可以 ping 通主机 PCB，它们之间的数据传输将被加密。

在 RouterA 上执行 **display ike sa** 操作，结果如下。

```
[Huawei] display ike sa
     Conn-ID      Peer            VPN    Flag(s)     Phase
  -------------------------------------------------------------
      14          202.138.162.1    0      RD|ST          1
      16          202.138.162.1    0      RD|ST          2
  Flag Description:
  RD--READY   ST--STAYALIVE   RL--REPLACED   FD--FADING   TO--TIMEOUT
  HRT--HEARTBEAT   LKG--LAST KNOWN GOOD SEQ NO.   BCK--BACKED UP
```

----结束

配置文件

- RouterA 的配置文件

```
#
acl number 3101
 rule 5 permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
#
ipsec proposal tran1
 esp authentication-algorithm sha1
#
ike proposal 1
 encryption-algorithm aes-cbc-128
 authentication-algorithm md5
#
 ike local-name huawei01
#
ike peer spub v1
 exchange-mode aggressive
 pre-shared-key huawei
 ike-proposal 1
 local-id-type name
 remote-name huawei02
 local-address 202.138.163.1
 remote-address 202.138.162.1
#
ipsec policy map1 10 isakmp
 security acl 3101
 ike-peer spub
 proposal tran1
#
 ip route-static 10.1.2.0 255.255.255.0 202.138.163.2
#
interface Ethernet1/0/0
 ip address 202.138.163.1 255.255.255.0
 ipsec policy map1
#
return
```

- RouterB 的配置文件

```
#
acl number 3101
```

```
  rule 5 permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
#
ipsec proposal tran1
 esp authentication-algorithm sha1
#
ike proposal 1
 encryption-algorithm aes-cbc-128
 authentication-algorithm md5
#
 ike local-name huawei02
#
ike peer spua v1
 exchange-mode aggressive
 pre-shared-key huawei
 ike-proposal 1
 local-id-type name
 remote-name huawei01
 local-address 202.138.162.1
 remote-address 202.138.163.1
#
ipsec policy use1 10 isakmp
 security acl 3101
 ike-peer spua
 proposal tran1
#
 ip route-static 10.1.1.0 255.255.255.0 202.138.162.2
#
interface Ethernet1/0/0
 ip address 202.138.162.1 255.255.255.0
 ipsec policy use1
#
return
```

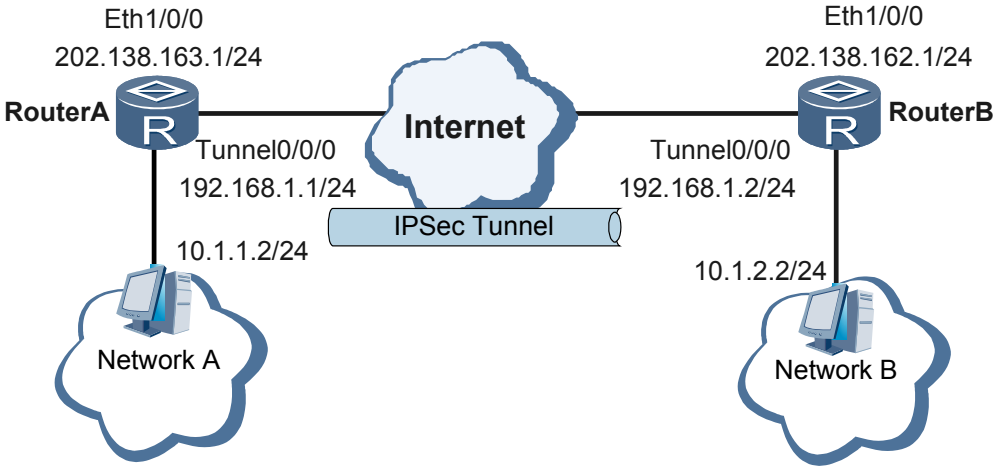
4.8.4 配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道示例

使用 IPSec 虚拟隧道接口建立 IPSec 隧道，简化了配置，减少了开销并使业务应用更加灵活。

组网需求

如图 4-6 所示，在 RouterA 和 RouterB 之间建立一个安全隧道，对 Tunnel 接口下的流量进行保护。安全协议采用 AH-ESP 协议，加密算法采用 3DES，认证算法采用 sha1。

图 4-6 配置采用 IPSec 虚拟隧道接口建立 IPSec 安全隧道组网图



配置思路

采用如下思路配置采用 IKE 协商方式建立安全联盟：

1. 配置接口的 IP 地址
2. 配置到对端的静态路由。
3. 配置 IKE 提议。
4. 配置 IKE 协商时需要的本机 ID 和 IKE Peer。
5. 配置安全提议。
6. 配置安全框架，并引用安全提议和 IKE 对等体。
7. 在接口上应用安全框架。

操作步骤

步骤 1 分别在 RouterA 和 RouterB 上配置各接口的 IP 地址。

在 RouterA 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 202.138.163.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

在 RouterB 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 202.138.162.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

步骤 2 分别在 RouterA 和 RouterB 上配置到对端的静态路由。

在 RouterA 上配置到目的端的静态路由，此处假设到达网络 B 的下一跳地址为 202.138.163.2。

```
[Huawei] ip route-static 10.1.2.0 255.255.255.0 202.138.163.2
```

在 RouterB 上配置到目的端的静态路由，此处假设到达网络 A 的下一跳地址为 202.138.162.2。

```
[Huawei] ip route-static 10.1.1.0 255.255.255.0 202.138.162.2
```

步骤 3 分别在 RouterA 和 RouterB 上配置 IKE 提议。

在 RouterA 上配置 IKE 提议。

```
[Huawei] ike proposal 1
[Huawei-ike-proposal-1] dh group5
[Huawei-ike-proposal-1] authentication-algorithm aes_xcbc_mac_96
[Huawei-ike-proposal-1] prf aes_xcbc_128
[Huawei-ike-proposal-1] quit
```

在 RouterB 上配置 IKE 提议。

```
[Huawei] ike proposal 1
[Huawei-ike-proposal-1] dh group5
[Huawei-ike-proposal-1] authentication-algorithm aes_xcbc_mac_96
[Huawei-ike-proposal-1] prf aes_xcbc_128
[Huawei-ike-proposal-1] quit
```

步骤 4 分别在 RouterA 和 RouterB 上配置本机 ID 和 IKE Peer。

在 RouterA 上配置进行 IKE 协商时需要的本机 ID 和 IKE Peer。

```
[Huawei] ike peer spub v2
[Huawei-ike-peer-spub] ike-proposal 1
[Huawei-ike-peer-spub] pre-shared-key huawei
[Huawei-ike-peer-spub] quit
```

在 RouterB 上配置进行 IKE 协商时需要的本机 ID 和 IKE Peer。

```
[Huawei] ike peer spua v2
[Huawei-ike-peer-spua] ike-proposal 1
[Huawei-ike-peer-spua] pre-shared-key huawei
[Huawei-ike-peer-spua] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ike peer** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ike peer name spub verbose
-----------------------------------------
  Peer name                : spub
  Pre-shared-key           : huawei
  proposal                 : 1
  Local ID type            :
  DPD                      : Disable
  DPD mode                 : Periodic
  DPD idle time            : 30
  DPD retransmit interval  : 15
  DPD retry limit          : 3
  Peer ID type             :
  Host name                :
  Peer IP address          :
  VPN name                 :
  Local IP address         : 202.138.163.1
  Remote name              :
  Nat-traversal            : Disable
  Configured IKE version   : Version two
  Auto-configure           : Disable
  PKI realm                : NULL
  Inband OCSP              : Disable
-----------------------------------------
```

步骤 5 分别在 RouterA 和 RouterB 上创建安全提议。

在 RouterA 上配置安全提议。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] transform ah-esp
[Huawei-ipsec-proposal-tran1] ah authentication-algorithm sha1
[Huawei-ipsec-proposal-tran1] esp authentication-algorithm sha1
[Huawei-ipsec-proposal-tran1] esp encryption-algorithm 3des
[Huawei-ipsec-proposal-tran1] quit
```

在 RouterB 上配置安全提议。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] transform ah-esp
[Huawei-ipsec-proposal-tran1] ah authentication-algorithm sha1
[Huawei-ipsec-proposal-tran1] esp authentication-algorithm sha1
[Huawei-ipsec-proposal-tran1] esp encryption-algorithm 3des
[Huawei-ipsec-proposal-tran1] quit
```

此时分别在 RouterA 和 RouterB 上执行 **display ipsec proposal** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec proposal
Number of Proposals: 1

IPSec proposal name: tran1
 Encapsulation mode: Tunnel
 Transform         : ah-esp-new
 AH protocol       : Authentication SHA1-HMAC-96
```

```
 ESP protocol      : Authentication SHA1-HMAC-96
                     Encryption     3DES
```

步骤 6 分别在 RouterA 和 RouterB 上创建安全框架。

在 RouterA 上配置安全框架。

```
[Huawei]  ipsec profile profile1
[Huawei-ipsec-profile-profile1] proposal tran1
[Huawei-ipsec-profile-profile1] ike-peer spub
[Huawei-ipsec-profile-profile1] quit
```

在 RouterB 上配置安全框架。

```
[Huawei]  ipsec profile profile2
[Huawei-ipsec-profile-profile1] proposal tran1
[Huawei-ipsec-profile-profile1] ike-peer spua
[Huawei-ipsec-profile-profile1] quit
```

步骤 7 分别在 RouterA 和 RouterB 的接口上应用各自的安全框架。

在 RouterA 的接口上引用安全框架。

```
[Huawei] interface tunnel 0/0/0
[Huawei-Tunnel0/0/0] ip address 192.168.1.1 24
[Huawei-Tunnel0/0/0] tunnel-protocol gre
[Huawei-Tunnel0/0/0] source 202.138.163.1
[Huawei-Tunnel0/0/0] destination 202.138.162.1
[Huawei-Tunnel0/0/0] ipsec profile profile1
[Huawei-Tunnel0/0/0] quit
```

在 RouterB 的接口上引用安全策略。

```
[Huawei] interface tunnel 0/0/0
[Huawei-Tunnel0/0/0] ip address 192.168.1.2 24
[Huawei-Tunnel0/0/0] tunnel-protocol gre
[Huawei-Tunnel0/0/0] source 202.138.162.1
[Huawei-Tunnel0/0/0] destination 202.138.163.1
[Huawei-Tunnel0/0/0] ipsec profile profile2
```

步骤 8 检查配置结果

配置成功后，分别在 RouterA 和 RouterB 上执行 **display ipsec profile** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec profile
=========================================
IPSec profile  : profile1
Using interface: Tunnel0/0/0
=========================================
 IPSec Profile Name        :profile1
 Peer Name                 :spub
 PFS   Group               :0 (0:Disable 1:Group1 2:Group2 5:Group5 14:Group14)
 SecondsFlag               :0 (0:Global 1:Local)
 SA Life Time Seconds      :3600
 KilobytesFlag             :0 (0:Global 1:Local)
 SA Life Kilobytes         :1843200
 Number of IPSec Proposals :1
 IPSec Proposals Name      :tran1
```

----结束

配置文件

- RouterA 的配置文件

```
#
ipsec proposal tran1
 transform ah-esp
```

```
 ah authentication-algorithm sha1
 esp authentication-algorithm sha1
 esp encryption-algorithm 3des
#
 ip route-static 10.1.2.0 255.255.255.0 202.138.163.2
#
ike proposal 1
 dh group5
 authentication-algorithm aes_xcbc_mac_96
 prf aes_xcbc_128
#
ike peer spub v2
 pre-shared-key huawei
 ike-proposal 1
#
 ipsec profile profile1
 ike-peer spub
 proposal tran1
#
interface Tunnel0/0/0
 ip address 192.168.1.1 255.255.255.0
 tunnel-protocol gre
 source 202.138.163.1
 destination 202.138.163.2
 ipsec profile profile1
#
interface Ethernet1/0/0
 ip address 202.138.163.1 255.255.255.0

#
return
```

- RouterB 的配置文件

```
#
ipsec proposal tran1
 transform ah-esp
 ah authentication-algorithm sha1
 esp authentication-algorithm sha1
 esp encryption-algorithm 3des
#
 ip route-static 10.1.1.0 255.255.255.0 202.138.162.2
#
ike proposal 1
 dh group5
 authentication-algorithm aes_xcbc_mac_96
 prf aes_xcbc_128

#
ike peer spua v2
 pre-shared-key huawei
 ike-proposal 1
#
 ipsec profile profile2
 ike-peer spua
 proposal tran1
#
interface Tunnel0/0/0
 ip address 192.168.1.2 255.255.255.0
 tunnel-protocol gre
 source 202.138.162.1
 destination 202.138.163.1
 ipsec profile profile2
#
interface Ethernet1/0/0
 ip address 202.138.162.1 255.255.255.0
#
return
```

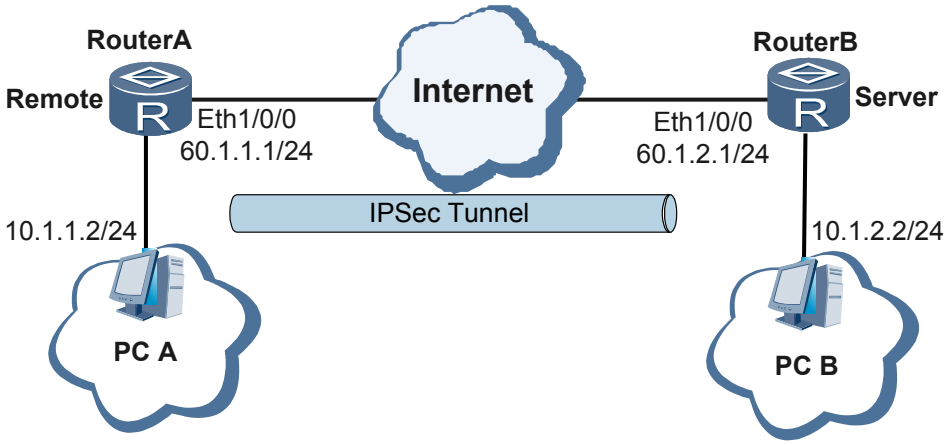
4.8.5 配置 Efficient VPN 采用 client 方式建立安全联盟示例

使用 Efficient VPN client 模式建立安全联盟在实际组网中的应用。

组网需求

如图 4-7 所示，在 RouterA 和 RouterB 之间建立一个安全隧道，对 PCA 代表的子网（10.1.1.0/24）与 PCB 代表的子网（10.1.2.0/24）之间做 NAT 后的数据流进行安全保护。Remote 与 Server 间自动地进行安全联盟的建立与密钥的交换，可以大幅度减化配置，提升效率。

图 4-7 配置 Efficient VPN 采用 client 方式建立安全联盟组网图



配置思路

RouterA 上采用如下思路配置 Efficient VPN 采用 client 方式建立安全联盟：

1. 配置接口的 IP 地址。
2. 配置静态路由。
3. 配置 client 模式的 Efficient VPN。
4. 配置 IKE 协商时的对端地址。
5. 配置预共享密钥。
6. 在接口上应用 Efficient VPN。

RouterB 上采用如下思路配置 Efficient VPN 采用 client 方式建立安全联盟：

1. 配置接口的 IP 地址。
2. 配置静态路由。
3. 配置要推送的资源属性。
4. 配置 IKE proposal 和 IKE peer。
5. 配置 IPSec proposal、模板策略和策略组。
6. 在接口上应用策略组。

操作步骤

步骤 1 在 RouterA 上作如下配置。

1. 在 RouterA 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 60.1.1.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

2. 在 RouterA 上配置到目的端的静态路由，此处假设到达网络 B 的下一跳地址为 60.1.1.2。

```
[Huawei] ip route-static 10.1.2.0 255.255.255.0 60.1.1.2
```

3. 配置 client 模式的 Efficient VPN。

```
[Huawei] ipsec efficient-vpn 2 mode client
```

4. 配置 IKE 协商时的对端地址。

```
[Huawei-ipsec-efficient-vpn-2] remote-address 60.1.2.1 v2
```

5. 配置预共享密钥。

```
[Huawei-ipsec-efficient-vpn-2] pre-shared-key huawei
[Huawei-ipsec-efficient-vpn-2] quit
```

6. 在接口上应用 Efficient VPN。

```
[Huawei] interface ethernet1/0/0
[Huawei-Ethernet1/0/0] ipsec efficient-vpn 2
```

步骤 2 在 RouterB 上作如下配置。

1. 在 RouterB 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 60.1.2.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
```

2. 在 RouterB 上配置到目的端的静态路由，此处假设到达网络 A 的下一跳地址为 60.1.2.2。

```
[Huawei] ip route-static 10.1.1.0 255.255.255.0 60.1.2.2
```

3. 配置要推送的资源属性，推送 IP 地址、DNS 服务器地址和 WINS 服务器地址。

```
[Huawei] ip pool pooltest
[Huawei-ip-pool-pooltest] network 100.1.1.0 mask 255.255.255.128
[Huawei-ip-pool-pooltest] quit
[Huawei] aaa
[Huawei-aaa] service-scheme schemetest
[Huawei-aaa-service-schemetest] dns 2.2.2.2
[Huawei-aaa-service-schemetest] dns 2.2.2.3 secondary
[Huawei-aaa-service-schemetest] ip-pool pooltest
[Huawei-aaa-service-schemetest] wins 3.3.3.2
[Huawei-aaa-service-schemetest] wins 3.3.3.3 secondary
[Huawei-aaa-service-schemetest] quit
[Huawei-aaa] quit
```

4. 配置 IKE proposal 和 IKE Peer。

```
[Huawei] ike proposal 5
[Huawei-ike-proposal-5] dh group2
[Huawei-ike-proposal-5] quit
[Huawei] ike peer rut3 v2
[Huawei-ike-peer-rut3] pre-shared-key huawei
[Huawei-ike-peer-rut3] ike-proposal 5
[Huawei-ike-peer-rut3] service-scheme schemetest
[Huawei-ike-peer-rut3] quit
```

5. 配置 IPSec proposal、模板策略和策略组。

```
[Huawei] ipsec proposal tran1
[Huawei-ipsec-proposal-tran1] quit
[Huawei] ipsec policy-template use1 10
[Huawei-ipsec-policy-templet-use1-10] ike-peer rut3
[Huawei-ipsec-policy-templet-use1-10] proposal tran1
[Huawei-ipsec-policy-templet-use1-10] sa duration time-based 600000
[Huawei-ipsec-policy-templet-use1-10] quit
[Huawei] ipsec policy policy1 10 isakmp template use1
```

6. 在接口上应用策略组。

```
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ipsec policy policy1
```

步骤 3 检查配置结果

1. 配置成功后，在主机 RouterA 执行 **ping** 操作仍然可以 ping 通主机 RouterB，它们之间的数据传输将被加密。

   在 RouterA 上执行 **display ike sa** 操作，结果如下。

```
[Huawei] display ike sa v2
     Conn-ID      Peer        VPN    Flag(s)     Phase
  -------------------------------------------------------------
       64         60.1.2.1    0      RD|ST         2
       62         60.1.2.1    0      RD|ST         1
  Flag Description:
  RD--READY   ST--STAYALIVE   RL--REPLACED   FD--FADING   TO--TIMEOUT
  HRT--HEARTBEAT   LKG--LAST KNOWN GOOD SEQ NO.   BCK--BACKED UP
```

2. 分别在 RouterA 和 RouterB 上执行 **display ipsec sa** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec sa
==============================
Interface: Ethernet 1/0/0
Path MTU: 1500
==============================
  ----------------------------
  IPSec efficient-vpn name: "2"
  Mode: EFFICIENTVPN-CLIENT MODE
  ----------------------------
    Connection ID     : 64
    Encapsulation mode: Tunnel
    Tunnel local      : 60.1.1.1
    Tunnel remote     : 60.1.2.1
    Flow source       : 100.1.1.126/255.255.255.255 0/0
    Flow destination  : 0.0.0.0/0.0.0.0 0/0
    [Outbound ESP SAs]
      SPI: 3752053811 (0xdfa3cc33)
      proposal: ESP-ENCRYPT-DES-64 ESP-AUTH-MD5
      SA remaining key duration (bytes/sec): 1887436800/1390
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
    [Inbound ESP SAs]
      SPI: 4182141148 (0xf94668dc)
      proposal: ESP-ENCRYPT-DES-64 ESP-AUTH-MD5
      SA remaining key duration (bytes/sec): 1887436800/1390
      Max received sequence-number: 0
      UDP encapsulation used for NAT traversal: N
```

3. 在 RouterA 上执行 **display ipsec efficient-vpn** 显示 Efficient VPN 策略的信息。

```
[Huawei] display ipsec efficient-vpn
=======================================
IPSec efficient-vpn name: 2
Using interface    : Ethernet1/0/0
=======================================
 IPSEC Efficient-vpn Name  : 2
 IPSEC Efficient-vpn Mode  : 1 (1:Client 2:Network)
 ACL Number                :
 Auth Method               : 8 (8:PSK 9:RSA)
 VPN name                  :
```

```
 Local ID Type            : 1 (1:IP 2:Name)
 Remote Address           : 60.1.2.1
 IKE Version              : 2 (1:IKEv1 2:IKEv2)
 FQDN                     :
 Pre Shared Key           : huawei
 PFS Type                 : 0 (0:Disable 1:Group1 2:Group2 5:Group5 14:Group14)
 Local Address            :
 Remote Name              :
 PKI Object               :
 Interface loopback       : LoopBack100
 Interface loopback IP    : 100.1.1.126/32
 Dns server IP            : 2.2.2.2, 2.2.2.3
 Wins server IP           : 3.3.3.2, 3.3.3.3
```

----结束

配置文件

- RouterA 的配置文件

```
#
ipsec efficient-vpn 2 mode client
 remote-address 60.1.2.1 v2
pre-shared-key huawei
#
interface Ethernet1/0/0
 ip address 60.1.1.1 255.255.255.0
 ipsec efficient-vpn 2
#
 ip route-static 10.1.2.0 255.255.255.0 60.1.1.2
#
return
```

- RouterB 的配置文件

```
#
ipsec proposal tran1
#
ike proposal 5
 dh
group2
#
ike peer rut3
v2
 pre-shared-key
huawei
 ike-proposal
5
 service-scheme
schemetest
#
ipsec policy-template use1
10
 ike-peer
rut3
 proposal
tran1
 sa duration time-based
600000
#
ipsec policy policy1 10 isakmp template
use1
#
ip pool
pooltest
 network 100.1.1.0 mask
255.255.255.128
#
aaa
 service-scheme
```

```
schemetest
  dns
2.2.2.2
  dns 2.2.2.3
secondary
  ip-pool
pooltest
  wins
3.3.3.2
  wins 3.3.3.3
secondary
#
interface
Ethernet1/0/0
 ip address 60.1.2.1
255.255.255.0
 ipsec policy policy1
#
 ip route-static 10.1.1.0 255.255.255.0 60.1.2.2
#
return
```

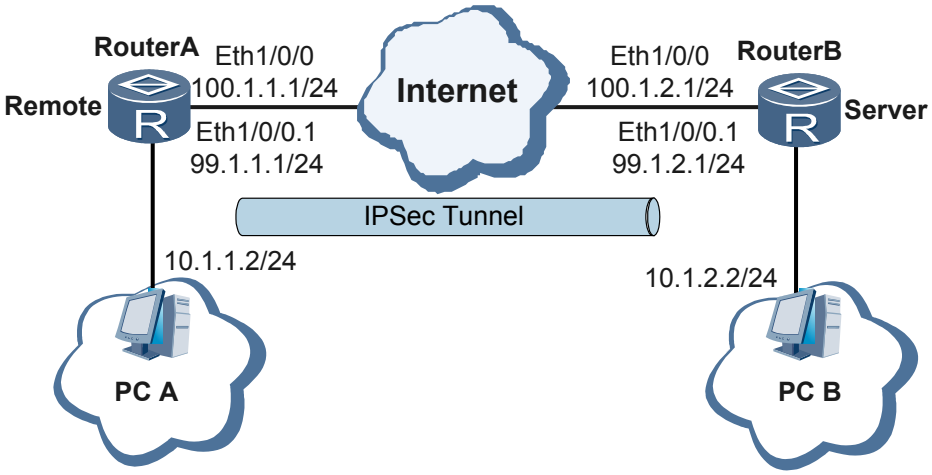
4.8.6 配置 Efficient VPN 采用 network 方式建立安全联盟示例

使用 Efficient VPN network 模式建立安全联盟在实际组网中的应用。

组网需求

如图 4-8 所示，在 RouterA 和 RouterB 之间建立一个安全隧道，对 PCA 代表的子网（10.1.1.0/24）与 PCB 代表的子网（10.1.2.0/24）之间符合 ACL 特征的数据流进行安全保护。Network 模式中，Remote 设备不向 Server 申请 IP 地址，不自动启用 NAT/PAT 功能。

图 4-8 配置 Efficient VPN 采用 network 方式建立安全联盟组网图



配置思路

采用如下思路配置 Efficient VPN 采用 Network 方式建立安全联盟：

1. 配置接口的 IP 地址。

2. 配置静态路由。
3. 配置 ACL，以定义要保护的数据流。
4. 配置 network 模式的 Efficient VPN。
5. 在接口上应用 Efficient VPN 策略。

操作步骤

步骤 1 分别在 RouterA 和 RouterB 上配置各接口的 IP 地址。

在 RouterA 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 100.1.1.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
[Huawei] interface ethernet 1/0/0.1
[Huawei-Ethernet1/0/0.1] ip address 99.1.1.1 255.255.255.0
[Huawei-Ethernet1/0/0.1] dot1q termination vid 1
[Huawei-Ethernet1/0/0.1] arp broadcast enable
[Huawei-Ethernet1/0/0.1] quit
```

在 RouterB 上配置接口的 IP 地址。

```
<Huawei> system-view
[Huawei] interface ethernet 1/0/0
[Huawei-Ethernet1/0/0] ip address 100.1.2.1 255.255.255.0
[Huawei-Ethernet1/0/0] quit
[Huawei] interface ethernet 1/0/0.1
[Huawei-Ethernet1/0/0.1] ip address 99.1.2.1 255.255.255.0
[Huawei-Ethernet1/0/0.1] dot1q termination vid 1
[Huawei-Ethernet1/0/0.1] arp broadcast enable
[Huawei-Ethernet1/0/0.1] quit
```

步骤 2 分别在 RouterA 和 RouterB 上配置到对端的静态路由。

在 RouterA 上配置到目的端的静态路由，此处假设到达网络 B 的下一跳地址为 100.1.1.2。

```
[Huawei] ip route-static 10.1.2.0 255.255.255.0 100.1.1.2
```

在 RouterB 上配置到目的端的静态路由，此处假设到达网络 A 的下一跳地址为 100.1.2.2。

```
[Huawei] ip route-static 10.1.1.0 255.255.255.0 100.1.2.2
```

步骤 3 分别在 RouterA 和 RouterB 上配置访问控制列表，定义各自要保护的数据流。

在 RouterA 上配置访问控制列表。

```
[Huawei] acl number 3000
[Huawei-acl-adv-3000] rule 5 permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
[Huawei-acl-adv-3000] quit
```

在 RouterB 上配置访问控制列表。

```
[Huawei] acl number 3000
[Huawei-acl-adv-3000] rule 5 permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
[Huawei-acl-adv-3000] quit
```

步骤 4 分别在 RouterA 和 RouterB 上分别配置 network 模式的 Efficient VPN。

在 RouterA 上配置 network 模式的 Efficient VPN。

```
[Huawei] ipsec efficient-vpn easyvpn_1 mode network
[Huawei-ipsec-efficient-vpn-easyvpn_1] remote-address 99.1.2.1 v1
```

```
[Huawei-ipsec-efficient-vpn-easyvpn_1] pre-shared-key htipl1.,;[-09876543211;'[]
[Huawei-ipsec-efficient-vpn-easyvpn_1] security acl 3000
[Huawei-ipsec-efficient-vpn-easyvpn_1] quit
```

在 RouterB 上配置 network 模式的 Efficient VPN。

```
[Huawei] ipsec efficient-vpn easyvpn_1 mode network
[Huawei-ipsec-efficient-vpn-easyvpn_1] remote-address 99.1.1.1 v1
[Huawei-ipsec-efficient-vpn-easyvpn_1] pre-shared-key htipl1.,;[-09876543211;'[]
[Huawei-ipsec-efficient-vpn-easyvpn_1] security acl 3000
[Huawei-ipsec-efficient-vpn-easyvpn_1] quit
```

步骤 5 分别在 RouterA 和 RouterB 的子接口上应用 Efficient VPN 策略。

在 RouterA 的子接口上应用 Efficient VPN 策略。

```
[Huawei] interface ethernet 1/0/0.1
[Huawei-Ethernet1/0/0.1] ipsec efficient-vpn easyvpn_1
```

在 RouterB 的子接口上应用 Efficient VPN 策略。

```
[Huawei] interface ethernet 1/0/0.1
[Huawei-Ethernet1/0/0.1] ipsec efficient-vpn easyvpn_1
```

步骤 6 检查配置结果

配置成功后，在 RouterA 执行 **ping** 操作可以 ping 通 RouterB，它们之间的数据传输将被加密。

- 分别在 RouterA 和 RouterB 上执行 **display ike sa** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ike sa
  Conn-ID         Peer       VPN   Flag(s)      Phase
  -------------------------------------------------------------
       3          99.1.2.1    0     RD|ST          2
       2          99.1.2.1    0     RD|ST          1
  Flag Description:
  RD--READY   ST--STAYALIVE   RL--REPLACED   FD--FADING   TO--TIMEOUT
  HRT--HEARTBEAT   LKG--LAST KNOWN GOOD SEQ NO.   BCK--BACKED UP
```

- 分别在 RouterA 和 RouterB 上执行 **display ipsec sa** 会显示所配置的信息，以 RouterA 为例。

```
[Huawei] display ipsec sa
==============================
Interface: Ethernet 1/0/0.1
Path MTU: 1500
==============================
  ----------------------------
  IPSec efficient-vpn name: "easyvpn_1"
  mode: EFFICIENTVPN-NETWORK MODE
  ----------------------------
    Connection ID: 3
    encapsulation mode: Tunnel
    tunnel local      : 99.1.1.1
    tunnel remote     : 99.1.2.1
    Flow source       : 100.1.1.1/0.0.0.0 0/0
    Flow destination  : 100.1.2.1/0.0.0.0 0/0
    [Outbound ESP SAs]
      SPI: 71167994 (0x43deffa)
      proposal: ESP-ENCRYPT-AES-256 SHA2-512-256
      SA remaining key duration (bytes/sec): 1887436800/1845
      Max sent sequence-number: 0
      UDP encapsulation used for NAT traversal: N
    [Intbound ESP SAs]
      SPI: 1488468104 (0x58b83888)
      Proposal: ESP-ENCRYPT-AES-256 SHA2-512-256
      SA remaining key duration (bytes/sec): 1887436800/1845
      Max received sequence-number: 0
      UDP encapsulation used for NAT traversal: N
```

----结束

配置文件

- RouterA 的配置文件

```
#
acl number 3000
 rule 5 permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255
#
ipsec efficient-vpn easyvpn_1 mode network
 remote-address 99.1.2.1 v1
 pre-shared-key htipl1.,;[-09876543211;'[]
 security acl 3000
#
interface Ethernet1/0/0
 ip address 100.1.1.1 255.255.255.0
#
 ip route-static 10.1.2.0 255.255.255.0 100.1.1.2
#
interface Ethernet1/0/0.1
 dot1q termination vid 1
 ip address 99.1.1.1 255.255.255.0
 ipsec efficient-vpn easyvpn_1
 arp broadcast enable
#
return
```

- RouterB 的配置文件

```
#
acl number 3000
 rule 5 permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255
#
ipsec efficient-vpn easyvpn_1 mode network
 remote-address 99.1.1.1 v1
 pre-shared-key htipl1.,;[-09876543211;'[]
 security acl 3000
#
interface Ethernet1/0/0
 ip address 100.1.2.1 255.255.255.0
#
 ip route-static 10.1.1.0 255.255.255.0 100.1.2.2
#
interface Ethernet1/0/0.1
 dot1q termination vid 1
 ip address 99.1.2.1 255.255.255.0
 ipsec efficient-vpn easyvpn_1
 arp broadcast enable
#
return
```

5 DSVPN 配置

关于本章

DSVPN 配置涉及到的网元包括源分支、目的分支、中心节点等路由器设备。

5.1 DSVPN 概述
DSVPN 是 Dynamic Smart VPN 的简称，是指 Hub-Spoke 网络模型中，分支和分支通过 NHRP 协议，动态建立数据转发通道的技术。

5.2 AR1200 支持的 DSVPN 特性
AR1200 支持的 DSVPN 特性主要依赖于：分支间路由部署方案和多点 GRE Tunnel 接口。

5.3 配置 DSVPN
DSVPN 部署时，可选择不启用 IPSec，如果启用 IPSec 对 GRE 流量进行保护，需将 NHRP Peer 信息告知 IPSec 特性。

5.4 维护 DSVPN
显示 DSVPN 的配置信息，清除 DSVPN 的统计信息。

5.5 配置举例
介绍使用不同路由方案部署 DSVPN 示例。

5.1 DSVPN 概述

DSVPN 是 Dynamic Smart VPN 的简称，是指 Hub-Spoke 网络模型中，分支和分支通过 NHRP 协议，动态建立数据转发通道的技术。

传统的 Hub-Spoke 网络模型中，数据流量主要集中于分支与中心之间。如果分支之间有流量互通时，并且应用了 IPSec 技术，中心需要在发送数据的分支隧道上解密，在接收数据的分支隧道上重新加密。分支到分支的流量跨越中心，耗费了中心的资源并引入延时。通过 DSVPN 技术，分支间可动态建立数据转发隧道，解决了上述问题。

要使分支间直接建立隧道，进行分支子网间的直接通信，则分支子网的路由下一跳不能是总部路由器，必须为其他分支，有以下路由部署方案：

- 分支间配置静态路由

  源分支通过配置静态路由协议，保存其他分支的子网路由，引导分支间隧道的建立。

- 分支间互相学习路由

  设备启动路由协议，实现分支与分支、分支与总部的路由学习。总部设备上，为实现分支间的路由通告，所有的分支，必须连接于总部的同一逻辑接口。如果启动的是 RIP 路由协议，需关闭 RIP 距离向量型路由协议的水平分割（Split Horizon）功能，实现分支间路由直接通告。如果启动的是 OSPF 路由协议，OSPF 是链路状态型路由协议，其本身不存在水平分割问题。

- 分支只有到总部的汇聚路由

  分支间互相学习路由，给分支路由器的性能、容量要求带来了挑战。分支节点可仅设置到总部的路径为默认转发路径，到目的分支的路由信息由 NHRP（NBMA Next Hop Resolution Protocol）解析应答进行添加，完成分支间的直接通信。

说明

DSVPN 部署时，可选择不启用 IPSec。如果需要启用 IPSec 对 GRE 流量进行保护，需将 peer 信息中对端的 IP 地址信息告知本端设备，用于建立 IPSec 隧道。

5.2 AR1200 支持的 DSVPN 特性

AR1200 支持的 DSVPN 特性主要依赖于：分支间路由部署方案和多点 GRE Tunnel 接口。

分支间路由学习部署 DSVPN 和分支只有到总部的汇聚路由部署 DSVPN，简要配置思路如下：

1. 创建 Tunnel 接口：指定 Tunnel 的源地址。
2. 配置各设备之间路由可达。
3. 配置 Peer 表项：在分支节点上配置中心节点的 peer 表项。

说明

DSVPN 功能使用 License 授权，缺省情况下，设备的 DSVPN 功能受限无法使用。如果需要使用 DSVPN 功能，请联系华为办事处申请并购买如下 License，

- AR1200 安全业务增值包
- AR1200 DSVPN（Dynamic Smart VPN）功能

5.3 配置 DSVPN

DSVPN 部署时，可选择不启用 IPSec，如果启用 IPSec 对 GRE 流量进行保护，需将 NHRP Peer 信息告知 IPSec 特性。

5.3.1 建立配置任务

在配置 DSVPN 之前，了解其应用环境，以及配置 DSVPN 需要提前完成的任务和准备的数据。

应用环境

传统的 Hub-Spoke 网络模型中，数据流量主要集中于分支与中心之间。如果分支之间有流量互通时，针对 IPSec，中心需要在发送数据分支的隧道上解密，在接收数据的分支隧道上重新加密。分支到分支的流量跨越中心，耗费了中心的资源并引入延时，通过配置 DSVPN，分支间可动态建立数据转发通道。

前置任务

在配置 DSVPN 之前，需要完成以下任务：

- 配置物理接口的链路层协议参数和 IP 地址，使物理接口的网络层协议状态为 Up。

数据准备

在配置 DSVPN 之前，需要准备以下数据。

序号	数据
1	Tunnel 接口的编号、Tunnel 的源地址或源端口、Tunnel 接口的 IP 地址
2	NHRP 认证字符串、NHRP 注册间隔时间、NHRP 表项保持时长
3	（可选）安全框架、IKE 对等体、安全提议

5.3.2 配置 MGRE

配置封装方式及 Tunnel 接口的源地址和网络地址。

背景信息

创建 Tunnel 接口后，需要指定封装方式为 MGRE、设置 Tunnel 接口的源地址，此外为使隧道支持动态路由协议，还要配置 Tunnel 接口的网络地址。在分支节点和中心节点的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，创建 Tunnel 接口，并进入 Tunnel 接口视图。

步骤 3 执行命令 **tunnel-protocol gre p2mp**,设置 Tunnel 封装为 MGRE 隧道。

说明

必须先指定隧道协议后才能进行隧道的源地址及其它参数的配置，修改隧道封装模式会删除该隧道下已配置的相关参数。

步骤 4 执行命令 **ip address** *ip-address* { *mask* | *mask-length* }，配置 Tunnel 接口的 IP 地址。

步骤 5 执行命令 **source** { *source-ip-address* | *interface-type interface-number* }，设置 Tunnel 接口的源地址或源接口。

----结束

5.3.3 配置 Tunnel 路由

DSVPN 依赖于分支间路由部署方案。

背景信息

在分支节点和中心节点上都必须存在经过 Tunnel 转发的路由，这样需要进行 MGRE 封装的报文才能正确转发。经过 Tunnel 接口的路由可以是静态路由，也可以是动态路由。在分支节点和中心节点的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 配置经过 Tunnel 接口的路由，选择如下方法之一：

- 执行命令 **ip route-static** *ip-address* { *mask* | *mask-length* } **tunnel** *interface-number* [**description** *text*]，配置静态路由。

  说明

  配置静态路由时，源端设备和目的端设备都需要配置。

- 配置动态路由。可以使用 OSPF、RIP 或 BGP，此处不再详述其配置方法。有关动态路由的配置，请参见《Huawei AR1200 系列配置指南-IP 路由》。

  说明

  - 如果动态路由配置为 OSPF，则在 Tunnel 接口下，OSPF 网络类型配置为广播型。
  - 如果动态路由配置为 RIP，则在 Tunnel 接口下，需关闭 RIP 水平分割进程。

----结束

5.3.4 配置分支 NHRP

分支节点上配置 NHRP 表项。

背景信息

NHRP 即下一跳解析协议用于解决 NBMA（Non-Broadcast Multiple Access）网络上的源节点如何获取到达目标节点下一跳的公网地址。在分支节点的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，进入 Tunnel 接口视图。

步骤 3 执行命令 **tunnel-protocol gre p2mp**，配置隧道模式为 MGRE。

步骤 4 执行命令 **nhrp entry** *protocol-address nbma-address* [**register**]，配置 NHRP 地址映射表。

步骤 5 （可选）执行命令 **nhrp authentication** *string*，配置 NHRP 协商的认证字符串。

缺省情况下，没有配置 NHRP 协商的认证字符串。

步骤 6 （可选）执行命令 **nhrp registration interval** *seconds*，配置 NHRP 注册间隔。

缺省情况下，分支节点向中心节点定时注册的时间间隔为 1800 秒。

步骤 7 （可选）执行命令 **nhrp entry holdtime seconds** *seconds*，配置 NHRP 表项保持时长。

缺省情况下，NHRP 表项保持时长为 7200s。

步骤 8 执行命令 **nhrp shortcut**，使能 nhrp shortcut 功能。

缺省情况下，未使能 nhrp shortcut 功能。

说明

如果需要实现分支只有到总部的汇聚路由部署 DSVPN 功能，需配置此步骤。

----结束

5.3.5 配置中心 NHRP

中心节点上配置 NHRP 表项。

背景信息

NHRP 下一跳解析协议用于解决 NBMA（Non-Broadcast Multiple Access）网络上的源节点如何获取到达目标节点下一跳的公网地址。在中心节点的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **interface tunnel** *interface-number*，进入 Tunnel 接口视图。

步骤 3 执行命令 **tunnel-protocol gre p2mp**，配置隧道模式为 MGRE。

步骤 4 （可选）执行命令 **nhrp authentication** *string*，配置 NHRP 协商的认证字符串。

缺省情况下，没有配置 NHRP 协商的认证字符串。

如果分支上配置了认证字符串但是中心节点上没有配置认证字符串，则不会进行认证字符串的认证。

步骤 5 执行命令 **nhrp entry multicast dynamic**，配置动态注册的分支加入 NHRP 组播成员表。

缺省情况下，没有配置动态注册的分支加入 NHRP 组播成员表。

说明

如果需要实现分支间路由学习部署 DSVPN 功能，需执行此步骤。

步骤 6 （可选）执行命令 **nhrp entry holdtime seconds** *seconds*，配置 NHRP 表项保持时长。

缺省情况下，NHRP 表项保持时长为 7200s。

步骤 7 执行命令 **nhrp registration no-unique**，配置 NHRP 注册时，允许覆盖冲突的 Peer 映射表项。

缺省情况下，NHRP 注册时不覆盖冲突的 Peer 映射表项。

步骤 8 执行命令 **nhrp redirect**，使能 nhrp redirect 功能。

缺省情况下，未使能 nhrp redirect 功能。

说明

如果需要实现分支只有到总部的汇聚路由部署 DSVPN 功能，需执行此步骤。

----结束

5.3.6 （可选）配置 IPSec 安全框架

为简化 IPSec 策略管理的复杂度，系统提供 IPSec 安全框架功能。

背景信息

DSVPN 部署时，可选择不启用 IPSec，如果需要启用 IPSec 对 GRE 流量进行保护，需将 NHRP Peer 信息告知 IPSec 特性。在分支和中心节点的路由器上进行如下配置。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **ipsec profile** *profile-name*，创建一个 IPSec 安全框架，并进入安全框架视图。

步骤 3 执行命令 **ike peer** *peer-name*，绑定 IKE 对等体。

缺省情况下，安全框架没有引用 IKE 对等体。

说明

配置 IKE 对等体的步骤请参见 **4.4.4 配置 IKE Peer**。

步骤 4 执行命令 **proposal** *proposal-name*，绑定 IPSec 提议。

缺省情况下，安全框架没有引用任何安全提议。

说明

配置 IPSec 安全提议的步骤请参见 **4.4.5 配置 IPSec 安全提议**。

步骤 5 执行命令 **pfs** { **dh-group1** | **dh-group2** | **dh-group5** | **dh-group14** }，设置协商时，使用完美的前向安全 PFS 特性。

缺省情况下，安全框架发起协商时没有使用 PFS 特性。

步骤 6 执行命令 **quit**，返回系统视图。

步骤 7 执行命令 **interface tunnel** *interface-number*，进入 Tunnel 接口视图。

步骤 8 执行命令 **tunnel-protocol** { **gre** [**p2mp**] | **ipsec** | **ipv4-ipv6** | **none** }，配置隧道模式。

隧道接口的封装模式需要设置为 IPSec、GRE 或者 MGRE 方式，才能在 Tunnel 口下绑定 IPSec 安全框架。

步骤 9 执行命令 **ipsec profile** *profile-name*，配置接口绑定 IPSec 安全框架。

----结束

5.3.7 检查配置结果

配置 DSVPN 完成之后，可以查看 NHRP Peer 信息、IPSec 安全框架的配置情况。

前提条件

已完成 DSVPN 的所有配置。

操作步骤

- 执行 **display nhrp peer** 命令,查看 NHRP Peer 表信息。
- 执行 **display ipsec profile** [**brief** | **name** *profile-name*]命令,查看 IPSec 框架的信息。

----结束

5.4 维护 DSVPN

显示 DSVPN 的配置信息，清除 DSVPN 的统计信息。

5.4.1 显示 DSVPN 配置

需要查看 DSVPN 的基本信息时，可以通过显示命令查看 NHRP Peer 表的相关信息、NHRP 统计信息。

前提条件

已经完成 DSVPN 的所有配置。

操作步骤

- 执行 **display nhrp peer** 命令显示 NHRP Peer 表信息。
- 执行 **display nhrp statistics interface** *interface-type interface-number* 命令显示 NHRP 统计信息。

----结束

5.4.2 清除 DSVPN 信息

清除 NHRP 报文统计信息。

背景信息

注意

清除信息后，以前的信息将无法恢复，务必仔细确认。

操作步骤

- 在确认需要清除的信息后，请在用户视图下执行 **reset nhrp statistics interface** *interface-type interface-number* 命令删除 Tunnel 接口下的 NHRP 报文统计信息。

----结束

5.5 配置举例

介绍使用不同路由方案部署 DSVPN 示例。

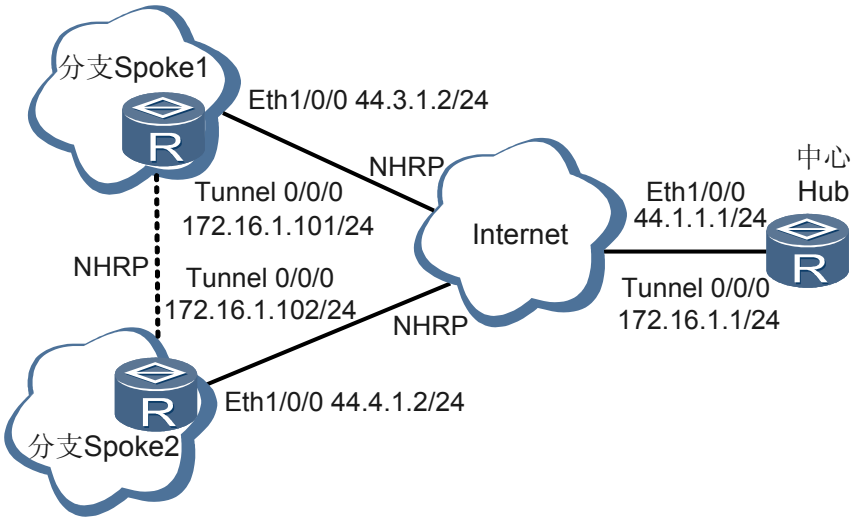
5.5.1 分支间进行路由学习部署 DSVPN 示例

采用分支间进行路由学习部署 DSVPN 在实际组网中的应用。

组网需求

如图 5-1 所示，中心 Hub、分支 Spoke1 和 Spoke2 属于同一自治系统，他们之间通过路由协议可达到 IP 网络互连。

图 5-1 配置采用分支间进行路由学习部署 DSVPN 组网图



配置思路

采用如下思路配置采用分支间进行路由学习部署 DSVPN：

1. 在各 Router 上运行路由协议，实现互通。
2. 在中心 Hub、分支 Spoke1 和 Spoke2 上创建 Tunnel 接口，指定 Tunnel 的源地址。
3. 在分支 Router 上配置 Hub 的 peer 表项。

数据准备

为完成此配置项，需准备如下的数据：

- 各 Router 之间路由可达。
- 各 Router 的 Tunnel 接口的源地址。

操作步骤

步骤 1 配置各接口 IP 地址。

按照图 5-1 配置各接口的 IP 地址，具体配置过程略。

步骤 2 配置各 Router 之间 NBMA 路由可达。

#在 Hub 的 Ethernet 接口上配置 OSPF 协议

```
[Huawei] ospf 2
[Huawei-ospf-2] area 0
[Huawei-ospf-2-area-0.0.0.0] network 44.1.1.0 0.0.0.255
[Huawei-ospf-2-area-0.0.0.0] quit
[Huawei-ospf-2] quit
```

#在 Spoke1 的 Ethernet 接口上配置 OSPF 协议

```
[Huawei] ospf 2
[Huawei-ospf-2] area 0
[Huawei-ospf-2-area-0.0.0.0]network 44.3.1.0 0.0.0.255
[Huawei-ospf-2-area-0.0.0.0] quit
[Huawei-ospf-2] quit
```

#在 Spoke2 的 Ethernet 接口上配置 OSPF 协议

```
[Huawei] ospf 2
[Huawei-ospf-2] area 0
[Huawei-ospf-2-area-0.0.0.0] network 44.4.1.0 0.0.0.255
[Huawei-ospf-2-area-0.0.0.0] quit
[Huawei-ospf-2] quit
```

步骤 3 配置各 Tunnel 接口的 OSPF 协议。

#配置 Hub

```
[Huawei] ospf 3
[Huawei-ospf-2] area 0
[Huawei-ospf-2-area-0.0.0.0] network 172.16.1.0 0.0.0.255
[Huawei-ospf-2-area-0.0.0.0] quit
[Huawei-ospf-2] quit
```

#配置 Spoke1

```
[Huawei] ospf 3
[Huawei-ospf-2] area 0
[Huawei-ospf-2-area-0.0.0.0] network 172.16.1.0 0.0.0.255
[Huawei-ospf-2-area-0.0.0.0] quit
[Huawei-ospf-2] quit
```

#配置 Spoke2

```
[Huawei] ospf 3
```

```
[Huawei-ospf-2] area 0
[Huawei-ospf-2-area-0.0.0.0] network 172.16.1.0 0.0.0.255
[Huawei-ospf-2-area-0.0.0.0] quit
[Huawei-ospf-2] quit
```

步骤 4 在各 Router 上配置 Tunnel 接口，并在分支 Spoke1 和分支 Spoke2 上分别配置 Hub 的 peer 表项。

#在 Hub 上配置 Tunnel 接口

```
[Huawei] system-view
[Huawei] interface tunnel 0/0/0
[Huawei-Tunnel0/0/0] tunnel-protocol gre p2mp
[Huawei-Tunnel0/0/0] source ethernet 1/0/0
[Huawei-Tunnel0/0/0] nhrp entry multicast dynamic
[Huawei-Tunnel0/0/0] ospf network-type broadcast
[Huawei-Tunnel0/0/0] ospf dr-priority 10
```

#在 Spoke1 上配置 Tunnel 接口和 Hub 的 peer 表项

```
[Huawei] system-view
[Huawei] interface tunnel 0/0/0
[Huawei-Tunnel0/0/0] tunnel-protocol gre p2mp
[Huawei-Tunnel0/0/0] source ethernet 1/0/0
[Huawei-Tunnel0/0/0] nhrp entry 172.16.1.1 44.1.1.1 register
[Huawei-Tunnel0/0/0] ospf network-type broadcast
[Huawei-Tunnel0/0/0] ospf dr-priority 8
```

#在 Spoke2 上配置 Tunnel 接口和 Hub 的 peer 表项

```
[Huawei] system-view
[Huawei] interface tunnel 0/0/0
[Huawei-Tunnel0/0/0] tunnel-protocol gre p2mp
[Huawei-Tunnel0/0/0] source ethernet 1/0/0
[Huawei-Tunnel0/0/0] nhrp entry 172.16.1.1 44.1.1.1 register
[Huawei-Tunnel0/0/0] ospf network-type broadcast
[Huawei-Tunnel0/0/0] ospf dr-priority 8
```

步骤 5 检查配置结果

配置成功后，检查 spoke 上的 peer 注册信息。

#在 Spoke1 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
  ---------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
---------------------------------------------------------------------------
172.16.1.1      32    44.1.1.1         172.16.1.1      static       hub
---------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-15:10:26
Expire time     : --
```

#在 Spoke2 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
  ---------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
---------------------------------------------------------------------------
172.16.1.1      32    44.1.1.1         172.16.1.1      static       hub
---------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-15:12:53
Expire time     : --
```

说明

完成上述配置后，执行 **display nhrp peer all** 命令，spoke1 和 spoke2 上只能看到 Hub 的静态 Peer 表项。

检查 Hub 上 Spoke1 和 Spoke2 的注册信息。

在 Hub 上执行 **display nhrp peer all** 操作，结果如下。

```
[Huawei] display nhrp peer all
  ----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.101    32    44.3.1.2         172.16.1.101    dynamic      route tunnel
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2008.01.07-18:07:45
Expire time     : 2008.01.07-20:07:52
----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.102    32    44.4.1.2         172.16.1.102    dynamic      route tunnel
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2008.01.07-18:11:51
Expire time     : 2008.01.07-20:11:57
```

步骤 6 执行 **ping** 操作，查看配置结果

在 spoke1 上 ping 分支 spoke2 的 IP 地址 172.10.1.102，则在 spoke1 和 spoke2 上可以分别看到彼此的 Peer 表项。

#在 Spoke1 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
  ----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.1      32    44.1.1.1         172.16.1.1      static       hub
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-15:10:26
Expire time     : --
----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.102    32    44.4.1.2         172.16.1.102    dynamic      route tunnel
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-16:09:31
Expire time     : 2011.08.18-18:09:31
```

#在 Spoke2 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
  ----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.1      32    44.1.1.1         172.16.1.1      static       hub
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-15:12:53
Expire time     : --
----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.101    32    44.3.1.2         172.16.1.101    dynamic      route tunnel
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-16:10:33
Expire time     : 2011.08.18-18:10:33
----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.102    32    44.4.1.2         172.16.1.102    dynamic      local
----------------------------------------------------------------------------
```

```
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-16:10:33
Expire time     : 2011.08.18-18:10:33
```

----结束

配置文件

- Router spoke1 的配置文件

```
#
interface Ethernet1/0/0
 ip address 44.3.1.2 255.255.255.0
#
interface Tunnel0/0/0
 ip address 172.16.1.101 255.255.255.0
 tunnel-protocol gre p2mp
 source Ethernet1/0/0
 nhrp entry 172.16.1.1 44.1.1.1 register
 ospf network-type broadcast
 ospf dr-priority 8
#
ospf 2
 area 0.0.0.0
  network 44.3.1.0 0.0.0.255
#
ospf 3
 area 0.0.0.0
  network 172.16.1.0 0.0.0.255
#
return
```

- Router spoke2 的配置文件

```
#
interface Ethernet1/0/0
 ip address 44.4.1.2 255.255.255.0
#
interface Tunnel0/0/0
 ip address 172.16.1.102 255.255.255.0
 tunnel-protocol gre p2mp
 source Ethernet1/0/0
 nhrp entry 172.16.1.1 44.1.1.1 register
 ospf network-type broadcast
 ospf dr-priority 8
#
ospf 2
 area 0.0.0.0
  network 44.4.1.0 0.0.0.255
#
ospf 3
 area 0.0.0.0
  network 172.16.1.0 0.0.0.255
#
return
```

- Router Hub 的配置文件

```
#
interface Ethernet1/0/0
 ip address 44.1.1.1 255.255.255.0
#
interface Tunnel0/0/0
 ip address 172.16.1.1 255.255.255.0
 tunnel-protocol gre p2mp
 source Ethernet1/0/0
 nhrp entry multicast dynamic
 ospf network-type broadcast
 ospf dr-priority 10
#
ospf 2
```

```
 area 0.0.0.0
  network 44.4.1.0 0.0.0.255
#
ospf 3
 area 0.0.0.0
  network 172.16.1.0 0.0.0.255
#
return
```

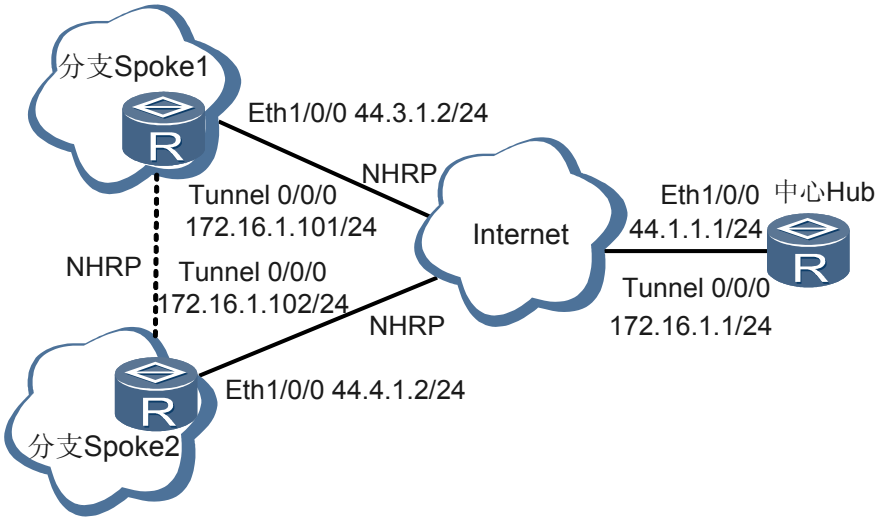
5.5.2 分支只有到总部的汇聚路由部署 DSVPN 示例

采用分支只有到总部的汇聚路由部署 DSVPN 在实际组网中的应用。

组网需求

如图 5-2 所示，中心 Hub、分支 Spoke1 和 Spoke2 属于同一自治系统，他们之间通过路由协议可达到 IP 网络互连。

图 5-2 配置采用分支只有到总部的汇聚路由部署 DSVPN 组网图



配置思路

采用如下思路配置采用分支只有到总部的汇聚路由部署 DSVPN：

1. 在各 Router 上运行路由协议，实现互通。
2. 在中心 Hub、分支 Spoke1 和 Spoke2 上创建 Tunnel 接口，指定 Tunnel 的源地址。
3. 在中心 Router 上使能 nhrp redirect 功能，在分支 Router 上使能 shortcut 功能。
4. 在分支 Router 上配置 Hub 的 peer 表项。

数据准备

为完成此配置项，需准备如下的数据：

- 各 Router 之间路由可达。
- 各 Router 的 Tunnel 接口的源地址。

操作步骤

步骤 1 配置各接口 IP 地址。

按照图 5-2 配置各接口的 IP 地址，具体配置过程略。

步骤 2 配置各 Router 之间 NBMA 路由可达。

#在 Hub 的 Ethernet 接口上配置 RIP 协议

```
[Huawei] rip
[Huawei-rip-1] network 44.0.0.0
[Huawei-rip-1] version 2
[Huawei-rip-1] quit
```

#在 Spoke1 的 Ethernet 接口上配置 RIP 协议

```
[Huawei] rip
[Huawei-rip-1] network 44.0.0.0
[Huawei-rip-1] version 2
[Huawei-rip-1] quit
```

#在 Spoke2 的 Ethernet 接口上配置 RIP 协议

```
[Huawei] rip
[Huawei-rip-1] network 44.0.0.0
[Huawei-rip-1] version 2
[Huawei-rip-1] quit
```

步骤 3 配置各 Tunnel 接口的 RIP 协议。

#配置 Hub

```
[Huawei] rip 2
[Huawei-rip-2] network 172.16.1.0
[Huawei-rip-2] version 2
[Huawei-rip-2] quit
```

#配置 Spoke1

```
[Huawei] rip 2
[Huawei-rip-2] network 172.16.1.0
[Huawei-rip-2] version 2
[Huawei-rip-2] quit
```

#配置 Spoke2

```
[Huawei] rip 2
[Huawei-rip-2] network 172.16.1.0
[Huawei-rip-2] version 2
[Huawei-rip-2] quit
```

步骤 4 在各 Router 上配置 Tunnel 接口，并分别在分支 Spoke1 和分支 Spoke2 上配置 Hub 的 peer 表项。

#在 Hub 上配置 Tunnel 接口，使能 nhrp redirect 功能

```
[Huawei] system-view
[Huawei] interface tunnel 0/0/0
[Huawei-Tunnel0/0/0] tunnel-protocol gre p2mp
[Huawei-Tunnel0/0/0] source ethernet 1/0/0
[Huawei-Tunnel0/0/0] nhrp redirect
[Huawei-Tunnel0/0/0] nhrp entry multicast dynamic
```

#在 Spoke1 上配置 Tunnel 接口和 Hub 的 peer 表项，使能 shortcut 功能

```
[Huawei] system-view
[Huawei] interface tunnel 0/0/0
[Huawei-Tunnel0/0/0] tunnel-protocol gre p2mp
[Huawei-Tunnel0/0/0] source ethernet 1/0/0
```

```
[Huawei-Tunnel0/0/0] nhrp shortcut
[Huawei-Tunnel0/0/0] nhrp entry 172.16.1.1 44.1.1.1 register
```

#在 Spoke2 上配置 Tunnel 接口和 Hub 的 peer 表项，使能 shortcut 功能

```
[Huawei] system-view
[Huawei] interface tunnel 0/0/0
[Huawei-Tunnel0/0/0] tunnel-protocol gre p2mp
[Huawei-Tunnel0/0/0] source ethernet 1/0/0
[Huawei-Tunnel0/0/0] nhrp shortcut
[Huawei-Tunnel0/0/0] nhrp entry 172.16.1.1 44.1.1.1 register
```

步骤 5 检查配置结果

配置成功后，检查 spoke 上的 peer 注册信息。

#在 Spoke1 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
  ----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.1      32    44.1.1.1         172.16.1.1      static       hub
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-15:10:26
Expire time     : --
```

#在 Spoke2 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
  ----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.1      32    44.1.1.1         172.16.1.1      static       hub
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-15:12:53
Expire time     : --
```

说明

完成上述配置后，执行 **display nhrp peer all** 命令，spoke1 和 spoke2 上只能看到 Hub 的静态 Peer 表项。

检查 Hub 上 Spoke1 和 Spoke2 的注册信息。

#在 Hub 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
 ----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.101    32    44.3.1.2         172.16.1.101    dynamic      route tunnel
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2008.01.07-18:07:45
Expire time     : 2008.01.07-20:07:52
----------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
----------------------------------------------------------------------------
172.16.1.102    32    44.4.1.2         172.16.1.102    dynamic      route tunnel
----------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2008.01.07-18:11:51
Expire time     : 2008.01.07-20:11:57
```

步骤 6 执行 ping 操作，查看配置结果

在 spoke1 上 ping 分支 spoke2 的 IP 地址 172.10.1.102，则在 spoke1 和 spoke2 上可以分别看到彼此的 Peer 表项。

#在 Spoke1 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
  ------------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
------------------------------------------------------------------------------
172.16.1.1      32    44.1.1.1         172.16.1.1      static       hub
------------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-15:10:26
Expire time     : --
------------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
------------------------------------------------------------------------------
172.16.1.102    32    44.4.1.2         172.16.1.102    dynamic      route tunnel
------------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-16:09:31
Expire time     : 2011.08.18-18:09:31
```

#在 Spoke2 上执行 **display nhrp peer all** 操作，结果如下

```
[Huawei] display nhrp peer all
  ------------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
------------------------------------------------------------------------------
172.16.1.1      32    44.1.1.1         172.16.1.1      static       hub
------------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-15:12:53
Expire time     : --
------------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
------------------------------------------------------------------------------
172.16.1.101    32    44.3.1.2         172.16.1.101    dynamic      route tunnel
------------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-16:10:33
Expire time     : 2011.08.18-18:10:33
------------------------------------------------------------------------------
Protocol-addr   Mask  NBMA-addr       NextHop-addr    Type         Flag
------------------------------------------------------------------------------
172.16.1.102    32    44.4.1.2         172.16.1.102    dynamic      local
------------------------------------------------------------------------------
Tunnel interface: Tunnel0/0/0
Created time    : 2011.08.18-16:10:33
Expire time     : 2011.08.18-18:10:33
```

----结束

配置文件

- Router spoke1 的配置文件

```
#
interface Ethernet1/0/0
 ip address 44.3.1.2 255.255.255.0
#
interface Tunnel0/0/0
 ip address 172.16.1.101 255.255.255.0
 tunnel-protocol gre p2mp
 source Ethernet1/0/0
 nhrp entry 172.16.1.1 44.1.1.1 register
 nhrp shortcut
#
rip 1
 version 2
 network 44.0.0.0
#
```

```
rip 2
 version 2
 network 172.16.1.0
#
return
```

- Router spoke2 的配置文件

```
#
interface Ethernet1/0/0
 ip address 44.4.1.2 255.255.255.0
#
interface Tunnel0/0/0
 ip address 172.16.1.102 255.255.255.0
 tunnel-protocol gre p2mp
 source Ethernet1/0/0
 nhrp entry 172.16.1.1 44.1.1.1 register
 nhrp shortcut
#
rip 1
 version 2
 network 44.0.0.0
#
rip 2
 version 2
 network 172.16.1.0
#
return
```

- Router Hub 的配置文件

```
#
interface Ethernet1/0/0
 ip address 44.1.1.1 255.255.255.0
#
interface Tunnel0/0/0
 ip address 172.16.1.1 255.255.255.0
 tunnel-protocol gre p2mp
 source Ethernet1/0/0
 nhrp redirect
 nhrp entry multicast dynamic
#
rip 1
 version 2
 network 44.0.0.0
#
rip 2
 version 2
 network 172.16.1.0
#
return
```

6 SSL VPN 配置

关于本章

SSL VPN（Secure Sockets Layer VPN）是以 HTTPS 为基础的安全接入的 VPN 技术，它利用 SSL 协议提供的数据加密、身份验证和消息完整性验证机制，为用户远程访问公司内部网络提供安全保障。

6.1 SSL VPN 介绍
通过 SSL VPN（Secure Sockets Layer VPN）技术，企业员工、客户和合作伙伴可以使用各种终端设备，在任何时间、任何地点通过 Internet 接入公司内部网络。

6.2 AR1200 支持的 SSL VPN 特性
AR1200 支持的 SSL VPN 特性包括：虚拟网关、SSL VPN 基本功能、管理 SSL VPN 用户和 SSL VPN 业务。

6.3 配置 SSL VPN 基本功能
SSL VPN 基本功能包括配置虚拟网关对应的接口和配置虚拟网关绑定 AAA 域。

6.4 管理 SSL VPN 用户
管理 SSL VPN 用户包括配置用户信息、最大在线用户数、用户最大在线时长和用户强制下线。

6.5 配置 SSL VPN 业务
用户使用 SSL VPN 的具体业务访问企业内网各种资源。

6.6 配置举例
介绍 SSL VPN 的配置举例。配置示例中包括组网需求、配置思路、操作步骤等。

6.1 SSL VPN 介绍

通过 SSL VPN（Secure Sockets Layer VPN）技术，企业员工、客户和合作伙伴可以使用各种终端设备，在任何时间、任何地点通过 Internet 接入公司内部网络。

随着 Internet 的普及，在家办公和移动办公也开始兴起，企业员工、客户和合作伙伴希望能够随时随地接入企业的内部网络，访问企业的内部资源。但是，远端用户访问企业的内部资源的过程中，会出现接入用户的身份可能不合法、远端接入主机可能不够安全等问题，这些都为企业内部网络带来了安全隐患。

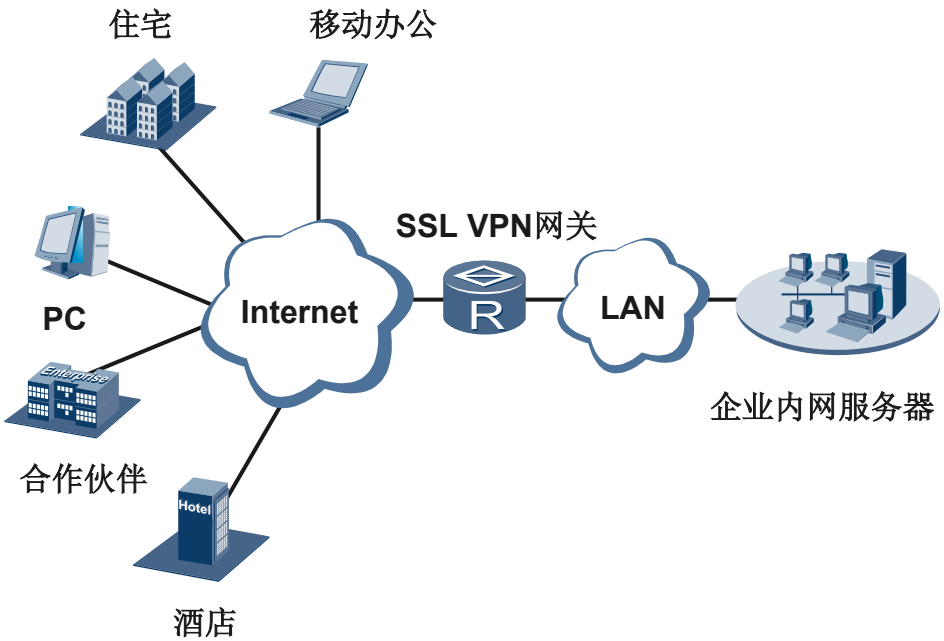
通过加密实现安全接入的 VPN 技术提供了一种安全机制，保护企业的内部网络不被攻击，内部资源不被窃取。

在实现安全接入的 VPN 技术中，SSL VPN 以 HTTPS 为基础，利用 SSL 协议提供的数据加密、身份验证和消息完整性验证机制，为用户远程访问企业内部网络提供了安全保证。

SSL VPN 在远程接入方面具有很多优势，适用于多种复杂的网络环境。如图 6-1 所示，SSL VPN 尤其适合应用于如下场景：

- 动态远程接入：用户使用各种远程终端设备，在任何时间、任何地点通过 Internet 接入公司内部网络。
- 不同的远程接入用户具有不同的访问权限：在使用 SSL VPN 网关时，远程接入用户可能是公司的员工、合作伙伴或其他人员，不同用户可以访问的资源可以不相同。
- 远程终端的运行环境具有多样性：远程终端可能安装不同的操作系统，使用不同的应用程序访问公司内部网络。

图 6-1 通过 SSL VPN 网关远程接入组网图



6.2 AR1200 支持的 SSL VPN 特性

AR1200 支持的 SSL VPN 特性包括：虚拟网关、SSL VPN 基本功能、管理 SSL VPN 用户和 SSL VPN 业务。

虚拟网关

AR1200 作为 SSL VPN 网关，可以划分为多个虚拟网关，AR1200 以虚拟网关为单位进行用户管理和业务配置。在 AR1200 部署 SSL VPN 功能时，管理员需要先创建虚拟网关，然后在所创建的虚拟网关下配置 SSL VPN 基本功能、管理 SSL VPN 用户和配置 SSL VPN 业务。

SSL VPN 基本功能

SSL VPN 基本功能包括配置虚拟网关对应的接口和配置虚拟网关绑定 AAA 域。

- AR1200 作为 SSL VPN 网关时，其虚拟网关对应的接口有两种类型：外网接口和内网接口。
  - 外网接口为 AR1200 连接 Internet 的接口，虚拟网关所管理的用户可以通过外网接口的 IP 地址获取 SSL VPN 网关的登录页面。
  - 内网接口为 AR1200 连接企业内网服务器的接口，通过该接口实现虚拟网关与企业内网服务器的通信。
- 虚拟网关需要对登录的用户进行 AAA 认证，以防止非法用户访问内网资源，保证企业信息安全。配置虚拟网关绑定 AAA 域后，用户必须通过 AAA 认证，才可以从虚拟网关获取准许访问内网资源的授权。

使用 AR1200 作为 SSL VPN 网关，必须配置基本功能并使基本功能生效。如果基本功能不生效，用户将无法使用 SSL VPN 网关访问企业内网服务器。

管理 SSL VPN 用户

管理 SSL VPN 用户包括：

- 配置用户信息

  为了能对合法用户进行本地认证，管理员可以配置用户登录虚拟网关的用户名和密码，并作为用户信息保存在虚拟网关中。虚拟网关对用户进行本地认证时，用户输入的用户名和密码与虚拟网关保存的用户名、密码一致，才允许用户登录虚拟网关。
- 配置最大在线用户数

  管理员可以配置虚拟网关支持的最大在线用户数，当虚拟网关在线用户数超过配置的最大用户数时，后面的用户无法登录。
- 配置用户最大在线时长

  用户成功登录虚拟网关以后，可能长时间不使用 SSL VPN 业务，从而造成资源的浪费。为了避免这种情况，管理员可以配置虚拟网关下的用户最大在线时长，在线时间超过最大在线时长的用户将被强制下线。用户强制下线后，其用户信息仍然保存在虚拟网关中。
- 配置用户强制下线

管理员可以根据用户名或者用户 ID 将虚拟网关下的指定用户或全部用户强制下线。用户强制下线后，其用户信息仍然保存在虚拟网关中。

SSL VPN 业务

AR1200 作为 SSL VPN 网关支持三种业务类型：Web 代理、端口转发和网络扩展。用户可以使用 SSL VPN 网关的这三种业务访问企业内网服务器中的各种资源。

- Web 代理业务，即用户使用浏览器以 HTTPS 方式、通过 SSL VPN 网关对内网 Web 服务器提供的资源进行访问。在这个过程中，SSL VPN 网关代理用户对内网 Web 服务器的访问，为用户访问内网 Web 服务器提供了安全的连接。
- 端口转发业务用于实现应用程序以 TCP 接入方式对内网服务器开放端口的安全访问。通过端口转发业务，用户可以访问内网中基于 TCP 的服务，包括远程访问服务（如 Telnet）、桌面共享服务、邮件服务等。
- 网络扩展业务可以使远程终端以 IP 接入方式与内网服务器在网络层实现安全通信，比如，在远程终端上 ping 内网服务器。

SSL VPN License

SSL VPN 功能使用 License 授权，缺省情况下，设备的 SSL VPN 功能受限无法使用。

如果需要使用 SSL VPN 功能，请联系华为办事处申请并购买如下 License：

- AR1200 安全业务增值包

说明

SSL VPN 支持的最大在线用户数受 License 控制，不同的 License 支持不同数目的在线用户，请用户按照需要组合购买。缺省情况下，新购买设备最大支持 2 个用户同时在线。

6.3 配置 SSL VPN 基本功能

SSL VPN 基本功能包括配置虚拟网关对应的接口和配置虚拟网关绑定 AAA 域。

6.3.1 建立配置任务

配置 SSL VPN 基本功能前了解此特性的应用环境、前置任务和数据准备，可以更快速、准确地完成配置任务。

应用环境

SSL VPN 基本功能包括配置虚拟网关对应的接口和配置虚拟网关绑定 AAA 域。

使用 AR1200 作为 SSL VPN 网关，必须配置基本功能并使基本功能生效。如果基本功能不生效，用户将无法使用 SSL VPN 网关访问企业内网服务器。

前置任务

在配置 SSL VPN 基本功能之前，需要完成以下任务：

- 已成功配置内外网接口的 IP 地址。
- 已成功配置 AAA 域。

数据准备

在配置 SSL VPN 基本功能之前，需准备以下数据。

序号	数据
1	虚拟网关的名称
2	虚拟网关对应的内外网接口的接口类型和接口编号
3	虚拟网关绑定 AAA 域的名称

6.3.2 创建 SSL VPN 虚拟网关

作为 SSL VPN 网关的 AR1200 是以虚拟网关为单位进行用户管理和业务配置。

背景信息

AR1200 作为 SSL VPN 网关，可以划分为多个虚拟网关，AR1200 以虚拟网关为单位进行用户管理和业务配置。在 AR1200 部署 SSL VPN 功能时，管理员需要先创建虚拟网关，然后在所创建的虚拟网关下配置 SSL VPN 基本功能、管理 SSL VPN 用户和配置 SSL VPN 业务。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **sslvpn gateway** *gateway-name*，创建虚拟网关并进入虚拟网关视图。

缺省情况下，AR1200 没有创建虚拟网关。

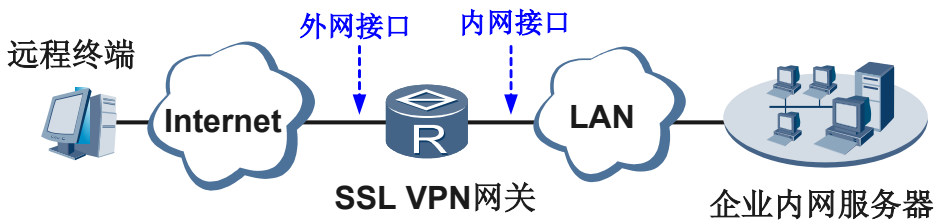
----结束

6.3.3 配置虚拟网关对应的接口

虚拟网关对应的接口包括外网接口和内网接口。

应用环境

图 6-2 虚拟网关对应接口的示意图



如图 6-2 所示，AR1200 作为 SSL VPN 网关时，其虚拟网关对应的接口有两种类型：外网接口和内网接口。

- 外网接口为 AR1200 连接 Internet 的接口，虚拟网关所管理的用户可以通过外网接口的 IP 地址获取 SSL VPN 网关的登录页面。
- 内网接口为 AR1200 连接企业内网服务器的接口，通过该接口实现虚拟网关与企业内网服务器的通信。

说明

虚拟网关对应的内外网接口为三层接口，且接口必须配置 IP 地址。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **sslvpn gateway** *gateway-name*，进入虚拟网关视图。

步骤 3 执行命令 **extranet interface** *interface-type interface-number*，配置虚拟网关对应的外网接口。

缺省情况下，虚拟网关没有配置对应的外网接口。

步骤 4 执行命令 **intranet interface** *interface-type interface-number*，配置虚拟网关对应的内网接口。

缺省情况下，虚拟网关没有配置对应的内网接口。

----结束

6.3.4 配置虚拟网关绑定 AAA 域

虚拟网关需要对用户进行 AAA 认证，以防止非法用户访问内网资源，保证企业信息安全。

背景信息

配置虚拟网关绑定 AAA 域后，用户必须通过 AAA 认证，才可以从虚拟网关获取准许访问内网资源的授权。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **sslvpn gateway** *gateway-name*，进入虚拟网关视图。

步骤 3 执行命令 **bind domain** *domain-name*，配置虚拟网关绑定 AAA 域。

缺省情况下，虚拟网关不绑定 AAA 域。

关于 AAA 域的详细配置信息请参见《Huawei AR1200 系列企业路由器 配置指南 安全配置》的 AAA 配置。

----结束

6.3.5 使能 SSL VPN 基本功能

使能虚拟网关的 SSL VPN 基本功能以后，虚拟网关的 SSL VPN 基本功能才可以生效。

前提条件

使能虚拟网关的 SSL VPN 基本功能前，SSL VPN 基本功能必须成功配置。包括：

- 已成功配置虚拟网关对应的外网接口和内网接口。参见 **6.3.3 配置虚拟网关对应的接口**。
- 虚拟网关已成功绑定 AAA 域。参见 **6.3.4 配置虚拟网关绑定 AAA 域**

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **sslvpn gateway** *gateway-name*，进入虚拟网关视图。

步骤 3 执行命令 **enable**，使能虚拟网关的 SSL VPN 基本功能。

缺省情况下，不使能虚拟网关的 SSL VPN 基本功能。

----结束

6.3.6 检查配置结果

虚拟网关基本功能配置成功后，可以查看虚拟网关的配置信息。

操作步骤

- 执行命令 **display sslvpn gateway** [*gateway-name*]，查看虚拟网关的配置信息。

----结束

6.4 管理 SSL VPN 用户

管理 SSL VPN 用户包括配置用户信息、最大在线用户数、用户最大在线时长和用户强制下线。

应用环境

管理 SSL VPN 用户包括：

- 配置用户信息

  为了能对合法用户进行本地认证，管理员可以配置用户登录虚拟网关的用户名和密码，并作为用户信息保存在虚拟网关中。虚拟网关对用户进行本地认证时，用户输入的用户名和密码与虚拟网关保存的用户名、密码一致，才允许用户登录虚拟网关。

  说明

  同一台主机不支持两个用户同时登录虚拟网关。

- 配置最大在线用户数

  管理员可以配置虚拟网关支持的最大在线用户数，当虚拟网关在线用户数超过配置的最大用户数时，后面的用户无法登录。

说明

AR1200 支持的在线用户数受 License 控制，不同级别的 License 支持不同数目的在线用户。缺省情况下，新购买设备最大支持 2 个用户同时在线。如果需要 AR1200 支持更多的用户同时在线，请根据需求联系华为办事处申请并购买相应的 License。

- 配置用户最大在线时长

  用户成功登录虚拟网关以后，可能长时间不使用 SSL VPN 业务，从而造成资源的浪费。为了避免这种情况，管理员可以配置虚拟网关下的用户最大在线时长，在线时间超过最大在线时长的用户将被强制下线。用户强制下线后，其用户信息仍然保存在虚拟网关中。

- 配置用户强制下线

  管理员可以根据用户名或者用户 ID 将虚拟网关下的指定用户或全部用户强制下线。用户强制下线后，其用户信息仍然保存在虚拟网关中。

前置任务

在配置管理 SSL VPN 用户之前，需要完成以下任务：

- SSL VPN 虚拟网关已成功创建。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行如下命令，配置用户登录虚拟网关的用户名和密码。

1. 执行命令 **aaa**，进入 AAA 视图。
2. 执行命令 **local-user** *user-name* **service-type sslvpn**，配置用户类型为 SSL VPN 虚拟网关用户。
3. 执行命令 **local-user** *user-name* **password** *password*，配置 SSL VPN 虚拟网关用户的密码。
4. 执行命令 **quit**，返回系统视图。

缺省情况下，AR1200 没有配置用户登录虚拟网关的用户名和密码。

步骤 3 执行命令 **sslvpn gateway** *gateway-name*，进入虚拟网关视图。

步骤 4 （可选）执行命令 **max-user** *number*，配置虚拟网关支持的最大在线用户数。

说明

AR1200 支持的最大在线用户数受 License 控制，不同级别的 License 支持不同数目的最大在线用户。缺省情况下，新购买设备最大支持 2 个用户同时在线。如果需要 AR1200 支持更多的用户同时在线，请根据需求联系华为办事处申请并购买相应的 License。

步骤 5 （可选）执行命令 **max-online-time** *number*，配置虚拟网关下的用户最大在线时长。

缺省情况下，虚拟网关下的用户最大在线时长为 120 分钟。

步骤 6 （可选）执行命令 **cut user** { **name** *user-name* | **id** *user-id* | **all** }，配置将虚拟网关下的用户强制下线。

----结束

检查配置结果

管理 SSL VPN 用户配置成功后，可以执行如下命令：

- 执行命令 **display sslvpn gateway** [*gateway-name*]，查看虚拟网关的配置信息。
- 执行命令 **display sslvpn gateway** *gateway-name* **access-user** [*user-name*]，查看指定虚拟网关下的用户信息。

6.5 配置 SSL VPN 业务

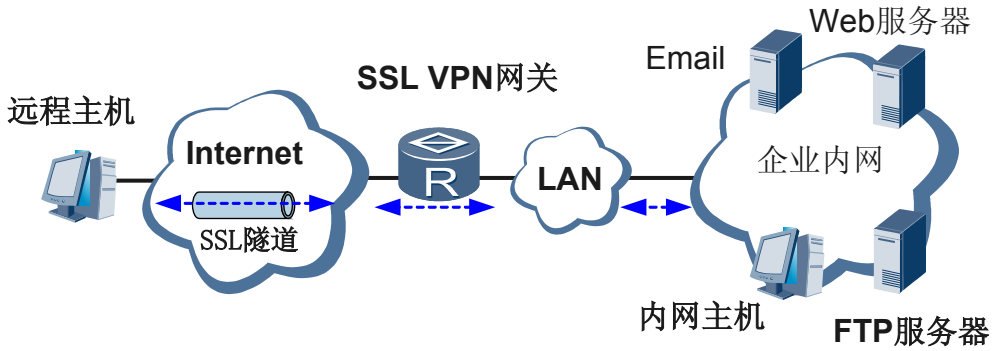
用户使用 SSL VPN 的具体业务访问企业内网各种资源。

6.5.1 建立配置任务

配置 SSL VPN 前了解此特性的应用环境、前置任务和数据准备，可以更快速、准确地完成配置任务。

应用环境

图 6-3 用户通过 SSL 网关访问企业内网服务器



如图 6-3 所示，SSL VPN 网关部署在企业网等内部网络的边缘，与安装在远程终端上的浏览器以及可以从浏览器自动下载的客户端软件配合，通过 SSL 协议保护在 Internet 上传输的用户数据，并代理用户对内网服务器的访问。

AR1200 作为 SSL VPN 网关支持三种业务类型：Web 代理、端口转发和网络扩展。用户可以使用 SSL VPN 网关的这三种业务访问企业内网服务器中的各种资源。

前置任务

在配置 SSL VPN 业务之前，需要完成以下任务：

- SSL VPN 虚拟网关已成功创建。

数据准备

在配置 SSL VPN 业务之前，需准备以下数据。

序号	数据
1	虚拟网关的名称

序号	数据
2	SSL VPN 业务的名称
3	SSL VPN 业务参数 ● Web 代理业务参数：内网 Web 服务器的 URL 地址 ● 端口转发业务参数：应用服务器的 IP 地址和端口号 ● 网络扩展业务参数：网络扩展业务使用的 IP 地址池、ACL、路由模式和隧道分离模式下的用户路由的目的 IP 地址、掩码

6.5.2 创建 SSL VPN 虚拟网关

作为 SSL VPN 网关的 AR1200 是以虚拟网关为单位进行用户管理和业务配置。

背景信息

AR1200 作为 SSL VPN 网关，可以划分为多个虚拟网关，AR1200 以虚拟网关为单位进行用户管理和业务配置。在 AR1200 部署 SSL VPN 功能时，管理员需要先创建虚拟网关，然后在所创建的虚拟网关下配置 SSL VPN 基本功能、管理 SSL VPN 用户和配置 SSL VPN 业务。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **sslvpn gateway** *gateway-name*，创建虚拟网关并进入虚拟网关视图。

缺省情况下，AR1200 没有创建虚拟网关。

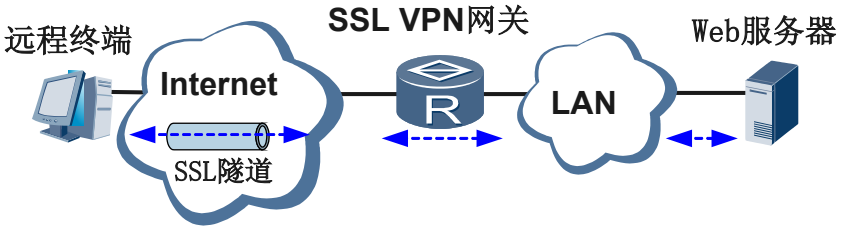
----结束

6.5.3 配置 web 代理业务

Web 代理业务，即用户使用浏览器以 HTTPS 方式,通过 SSL VPN 网关对内网 Web 服务器提供的资源进行访问。

背景信息

图 6-4 Web 代理业务的应用场景



如图 6-4 所示，用户使用浏览器以 HTTPS 方式，通过 SSL VPN 网关对内网 Web 服务器提供的资源进行访问。在这个过程中，SSL VPN 网关利用 Web 代理业务，代理用户对内网 Web 服务器的访问，为用户访问内网 Web 服务器提供了安全的连接。

管理员配置 Web 代理业务时，需要指定内网 Web 服务器 URL 地址，以便指定可以访问的内网 Web 服务器。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **sslvpn gateway** *gateway-name*，进入虚拟网关视图。

步骤 3 执行命令 **service-type web-proxy resource** *resource-name*，创建 Web 代理业务并进入 Web 代理业务视图。

缺省情况下，虚拟网关没有配置 Web 代理业务。

步骤 4 （可选）执行命令 **description** *description*，配置 Web 代理业务的描述信息。

步骤 5 执行命令 **link** *url* [**web-tunnel**]，配置内网 Web 服务器的 URL 地址。

缺省情况下，虚拟网关没有配置内网 Web 服务器的 URL 地址。

说明

作为 SSL VPN 网关的 AR1200 出现代理失败的情况时，用户可以选择以隧道模式访问内网 Web 服务器，但这样会降低访问的安全性。

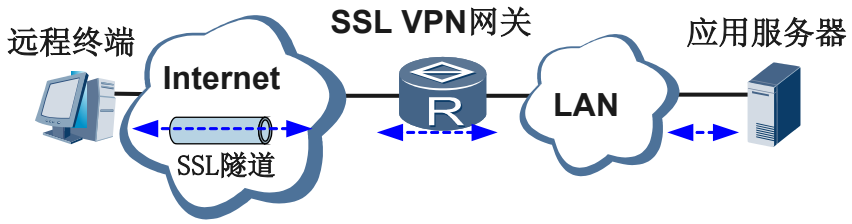
----结束

6.5.4 配置端口转发业务

端口转发业务用于实现应用程序以 TCP 接入方式对内网服务器开放端口的安全访问。

背景信息

图 6-5 端口转发业务的应用场景



如图 6-5 所示，通过端口转发业务，用户可以访问内网中基于 TCP 的服务，包括远程访问服务（如 Telnet）、桌面共享服务、邮件服务等。

说明

远程终端与应用服务器上基于 TCP 的应用程序的端口必须一致才能实现端口转发业务。

管理员配置端口转发业务时，需要指定内网应用服务器的 IP 地址和端口号，以便指定可以访问的内网应用服务器。

用户利用端口转发业务访问内网服务器时，不需要对现有的 TCP 应用程序进行升级，只需安装专用的客户端软件（该客户端软件从 Web 访问页面自动下载），由该软件实现使用 SSL连接传送应用层数据。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **sslvpn gateway** *gateway-name*，进入虚拟网关视图。

步骤 3 执行命令 **service-type port-forwarding resource** *resource-name*，创建端口转发业务并进入端口转发业务视图。

缺省情况下，虚拟网关没有配置端口转发业务。

步骤 4 （可选）执行命令 **description** *description*，配置端口转发业务的描述信息。

步骤 5 执行命令 **server ip-address** *ip-address* **port** *port-number*，配置端口转发业务可用的 IP 地址和端口号。

缺省情况下，虚拟网关没有配置端口转发业务可用的 IP 地址和端口号。

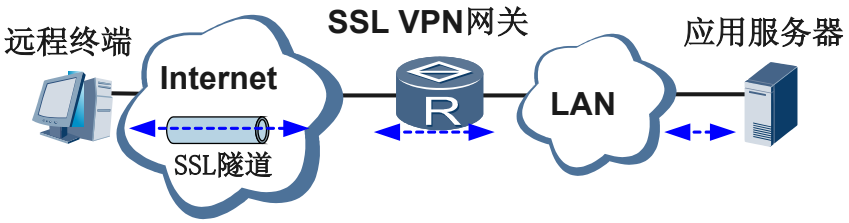
----结束

6.5.5 配置网络扩展业务

网络扩展业务用于实现远程终端以 IP 接入方式与内网服务器在网络层实现安全通信。

背景信息

图 6-6 网络扩展业务的应用场景



如图 6-6 所示，SSL VPN 网关通过网络扩展业务，可以使远程终端与内网服务器在网络层实现安全通信，比如：在远处终端与内网服务器之间实现文件共享。

用户通过网络扩展业务访问内网服务器前，需要安装专用的客户端软件（该客户端软件从 Web 访问页面自动下载），该客户端软件会在主机上安装一个虚拟网卡。客户端软件负责与 SSL VPN 网关建立 SSL 连接，为虚拟网卡申请 IP 地址，并设置以虚拟网卡为出接口的路由。

管理员配置网络扩展业务时，需要绑定网络扩展业务使用的 IP 地址池，客户端软件从该地址池中为虚拟网卡申请 IP 地址。

内网中某些重要的资源不希望用户访问，管理员可以执行 **bind acl** 命令绑定网络扩展业务使用的 ACL，也可以配置网络扩展业务使用的路由模式为隧道分离模式，在隧道分离模式下，远程终端只可以与内网指定网段的服务器进行通信。

操作步骤

步骤 1 执行命令 **system-view**，进入系统视图。

步骤 2 执行命令 **sslvpn gateway** *gateway-name*，进入虚拟网关视图。

步骤 3 执行命令 **service-type ip-forwarding resource** *resource-name*，创建网络扩展业务并进入网络扩展业务视图。

缺省情况下，虚拟网关没有配置网络扩展业务。

步骤 4 （可选）执行命令 **description** *description*，配置网络扩展业务的描述信息。

步骤 5 执行命令 **bind ip-pool** *pool-name*，绑定网络扩展业务使用的 IP 地址池。

缺省情况下，虚拟网关没有绑定网络扩展业务使用的 IP 地址池。

说明

若配置 IP 地址租期，则租期必须大于 SSL VPN 用户最大在线时长。

步骤 6 （可选）执行命令 **bind acl** *acl-number*，绑定网络扩展业务使用的 ACL。

步骤 7 （可选）配置网络扩展业务使用的路由模式。

- 执行命令 **route-mode full**，配置网络扩展业务使用的路由模式为全路由模式。
- 执行命令 **route-mode split**，配置网络扩展业务使用的路由模式为隧道分离模式。

缺省情况下，网络扩展业务使用的路由模式为全路由模式。

当网络扩展业务使用的路由模式配置为隧道分离模式时，请执行步骤 8。

步骤 8 （可选）执行命令 **route-split ip address** *ip-address* **mask** { *mask-length* | *mask* }，配置隧道分离模式下的用户路由。

说明

终端用户在应用网络扩展业务时，如果直接关掉 IE 等浏览器进程，程序的退出功能得不到执行而导致路由无法恢复。此时需要停止并重新启动网卡。

----结束

6.5.6 检查配置结果

虚拟网关业务配置成功后，可以查看业务的配置信息。

操作步骤

- 执行命令 **display sslvpn gateway** [*gateway-name*]，查看虚拟网关的配置信息。
- 执行命令 **display sslvpn gateway** *gateway-name* **resource class** { **web-proxy** | **port-forwarding** | **ip-forwarding** }，查看虚拟网关的资源信息。

----结束

6.6 配置举例

介绍 SSL VPN 的配置举例。配置示例中包括组网需求、配置思路、操作步骤等。

6.6.1 配置 SSL VPN 网关示例

以企业市场人员使用 SSL VPN 网关为例，实现企业市场人员访问企业各种内网资源。

组网环境

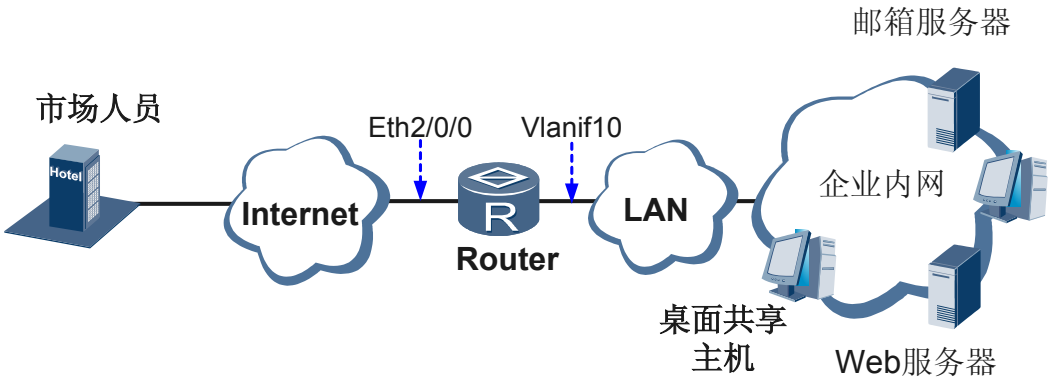
如图 6-7 所示，某企业通过 Router 与 Internet 相连接。Router 作为 SSL VPN 网关，处于外网的企业市场人员通过 Router 可以安全访问企业内网资源。

已知市场人员有如下访问需求：

- 市场人员需要访问企业内网的 Web 服务器和邮箱服务器，需要与企业内部主机（IP 地址：10.138.10.21）实现桌面共享，还需要 Ping 通企业内网部分主机（网段：10.138.10.64 ～ 10.138.10.95）。

要求管理员在 Router 上进行配置，满足企业市场人员的访问需求。

图 6-7 配置 SSL VPN 网关组网图



配置思路

采用如下的配置思路：

- 在 Router 上为企业市场人员创建一个虚拟网关，并在虚拟网关中配置相应的资源，以满足企业市场人员的访问需求。

数据准备

为完成配置举例，需准备如下的数据：

- 企业内网资源信息：

资源类型	IP 地址	端口号
Web 服务器	10.138.10.1	80
邮箱服务器	10.138.10.3	995
桌面共享主机	10.138.10.21	3389
远程主机	10.138.10.32 ～ 10.138.10.192	—

- 虚拟网关信息：

对应的用户	虚拟网关名称	虚拟网关对应的外网接口	虚拟网关对应的内网接口	AAA域	用户名和密码	所处的网段
市场人员	market	Ethernet 2/0/0	Vlanif10	default	liming 和 liming123456 wangjun 和 wangjun654321	10.135.30.0/24

说明

请用户根据实际情况选择合适的 AAA 域。关于 AAA 域的详细配置信息请参见《Huawei AR1200 系列企业路由器 配置指南 安全配置》的 AAA 配置。

- 外网接口 Ethernet2/0/0 的 IP 地址：1.1.1.1/24
- 内网接口 Vlanif10 的 IP 地址：10.138.10.254/24
- 网络扩展业务使用的 IP 地址池的网段：10.139.30.0/24

说明

在配置 SSL VPN 网关前，需要把 Router 配置为 HTTPS 服务器，并确保 Router、企业内网资源和用户之间路由可达。

操作步骤

步骤 1 配置 IP 地址池

```
<Huawei> system-view
[Huawei] sysname Router
[Router] ip pool market_pool
[Router-ip-pool-company_pool] network 10.139.30.0 mask 24
[Router-ip-pool-company_pool] quit
```

步骤 2 创建虚拟网关 market

```
[Router] sslvpn gateway market
```

步骤 3 配置基本功能，包括内外网接口和 AAA 域

```
[Router-sslvpn-market] extranet interface ethernet 2/0/0
[Router-sslvpn-market] intranet interface vlanif 10
[Router-sslvpn-market] bind domain default
[Router-sslvpn-market] enable
[Router-sslvpn-market] quit
```

步骤 4 配置用户信息

```
[Router] aaa
[Router-aaa] local-user liming service-type sslvpn
[Router-aaa] local-user liming password cipher liming123456
[Router-aaa] local-user wangjun service-type sslvpn
[Router-aaa] local-user wangjun password cipher wangjun654321
[Router-aaa] quit
```

步骤 5 配置业务

配置 Web 代理业务，实现市场人员访问 Web 服务器

```
[Router] sslvpn gateway market
[Router-sslvpn-market] service-type web-proxy resource market_web-proxy
[Router-sslvpn-market-wp-res-market_web-proxy] link http://10.138.10.1:80/
[Router-sslvpn-market-wp-res-market_web-proxy] quit
```

配置端口转发业务，实现市场人员访问邮箱服务器，并可以与企业内部主机（IP 地址：10.138.10.21）实现桌面共享

```
[Router-sslvpn-market] service-type port-forwarding resource market_port-forwarding
[Router-sslvpn-market-pf-res-market_port-forwarding] server ip-address 10.138.10.3 port 995
[Router-sslvpn-market-pf-res-market_port-forwarding] server ip-address 10.138.10.21 port 3389
[Router-sslvpn-market-pf-res-market_port-forwarding] quit
```

配置网络扩展业务，实现市场人员可以 Ping 通企业内网部分主机（网段：10.138.10.64 ～ 10.138.10.95）。

```
[Router-sslvpn-market] service-type ip-forwarding resource market_ip-forwarding
[Router-sslvpn-market-if-res-market_ip-forwarding] bind ip-pool market_pool
[Router-sslvpn-market-if-res-market_ip-forwarding] route-mode split
[Router-sslvpn-market-if-res-market_ip-forwarding] route-split ip address 10.138.10.64 mask 27
[Router-sslvpn-market-if-res-market_ip-forwarding] quit
[Router-sslvpn-market] quit
```

步骤 6 验证配置结果

市场人员在终端（比如 PC）打开 IE 浏览器，输入网址“https://1.1.1.1/sslvpn”，进入 Web 登录页面。市场人员输入用户名和密码，并选择名称为“market”的虚拟网关。认证成功后，市场人员在 Web 访问页面上查看可以访问的资源列表，包括 Web 服务器资源、邮箱服务器资源和共享桌面主机资源，并且可以 Ping 通企业内网部分主机（网段：10.138.10.64 ～ 10.138.10.95）。

----结束

配置文件

```
#
sysname Router
#
aaa
 local-user liming password cipher !9$"OZ$+1"-',917]_2Y71!!
 local-user liming service-type sslvpn
 local-user wangjun password cipher =S@P;D^[S_2)S(YTABROIQ!!
 local-user wangjun service-type sslvpn
#
interface Ethernet 2/0/0
 ip address 1.1.1.1 255.255.255.0
#
interface Vlanif 10
 ip address 10.138.10.254 255.255.255.0
#
ip pool company_pool
 network 10.139.30.0 mask 24
#
sslvpn gateway market
 extranet interface ethernet 2/0/0
 intranet interface vlanif 10
 bind domain default
 service-type web-proxy resource market_web-proxy
  link http://10.138.10.1:80/
 service-type port-forwarding resource market_port-forwarding
  server ip-address 10.138.10.3 port 995
  server ip-address 10.138.10.21 port 3389
 service-type ip-forwarding resource market_ip-forwarding
```

```
  bind ip-pool market_pool
  route-mode split
  route-split ip address 10.138.10.64 mask 27
#

return
```