CloudEngine 16800, 8800, 6800 系列交换机

典型配置案例（V300 版本）

文档版本 01

发布日期 2023-06-30

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暗示的声明或保证。

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： https://e.huawei.com

目 录

1 综合场景配置案例

1.1 基于 VRRP 的三层架构数据中心网络部署举例

适用产品和版本

- CloudEngine系列交换机V300R020C00或更高版本。
- USG5500系列产品V300R001版本。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

在数据中心场景中，采用接入层、汇聚层和核心层三层方式部署。用户希望：

- 考虑到业务的可靠性，接入层和汇聚层之间部署VRRP，在一条上行链路断开的时候，流量能切换到另外一条上行链路转发。
- 避免冗余备份链路导致的环网问题，消除接入层和汇聚层之间的环路。
- 核心层设备外挂防火墙，对业务流量提供安全过滤功能。
- 汇聚层和核心层部署OSPF协议实现三层互通。

图 1-1 基于 VRRP 的三层架构数据中心网络组网

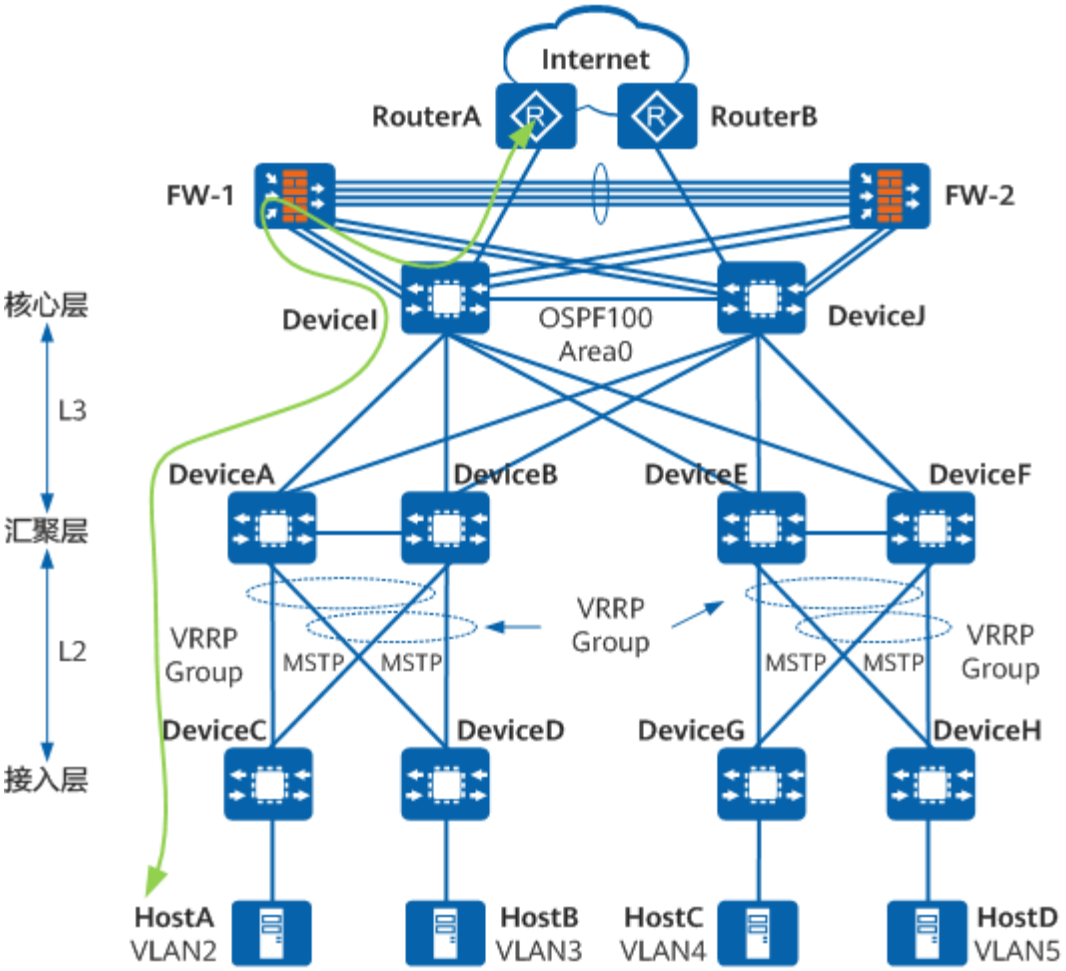


表 1-1 数据准备表（以 DeviceA、DeviceB、DeviceC 和 DeviceD 为例）

设备	VLAN及IP地址	接口编号	描述
DeviceA	VLAN：2 IP地址：10.1.2.102/24 虚拟IP地址：10.1.2.100	100GE1/0/1	TO-CE6800-DEVICEC
		100GE1/0/3	TO-CE16800-DEVICEB
	VLAN：3	100GE1/0/2	TO-CE6800-DEVICED

设备	VLAN及IP地址	接口编号	描述
	IP地址：10.1.3.102/24 虚拟IP地址：10.1.3.100	100GE1/0/3	TO-CE16800-DEVICEB
	VLAN：6 IP地址：10.1.6.102/24	100GE1/0/4	TO-CE16800-DEVICEI
	VLAN：7 IP地址：10.1.7.102/24	100GE1/0/5	TO-CE16800-DEVICEJ
DeviceB	VLAN：2 IP地址：10.1.2.103/24 虚拟IP地址：10.1.2.100	100GE1/0/2	TO-CE6800-DEVICEC
		100GE1/0/3	TO-CE16800-DEVICEA
	VLAN：3 IP地址：10.1.3.103/24 虚拟IP地址：10.1.3.100	100GE1/0/1	TO-CE6800-DEVICED
		100GE1/0/3	TO-CE16800-DEVICEA
	VLAN：6 IP地址：10.1.6.103/24	100GE1/0/4	TO-CE16800-DEVICEI
	VLAN：7 IP地址：10.1.7.103/24	100GE1/0/5	TO-CE16800-DEVICEJ
DeviceC	VLAN：2	100GE1/0/1	TO-CE16800-DEVICEA
		100GE1/0/2	TO-CE16800-DEVICEB
		100GE1/0/3	TO-HOSTA
DeviceD	VLAN：3	100GE1/0/1	TO-CE16800-DEVICEB
		100GE1/0/2	TO-CE16800-DEVICEA
		100GE1/0/3	TO-HOSTB
DeviceI	VLAN：6	100GE1/0/1	TO-CE16800-DEVICEA

设备	VLAN及IP地址	接口编号	描述
	IP地址：10.1.6.104/24	100GE1/0/2	TO-CE16800-DEVICEB
		100GE1/0/3	TO-CE16800-DEVICEE
		100GE1/0/4	TO-CE16800-DEVICEF
	VLAN：8 IP地址：10.1.8.104/24	100GE1/0/5	TO-ROUTERA
	VLAN：9 IP地址：172.16.1.2/24	100GE1/0/6	TO-FW-1
	VLAN：10 IP地址：172.16.2.2/24	100GE1/0/7	TO-FW-1
	VLAN：11 IP地址：172.16.3.2/24	100GE1/0/8	TO-FW-2
	VLAN：12 IP地址：172.16.4.2/24	100GE1/0/9	TO-FW-2
	VLAN：13 IP地址：10.1.13.102/24	100GE1/0/14	TO-CE16800-DEVICEJ
DeviceJ	VLAN：7 IP地址：10.1.7.104/24	100GE1/0/1	TO-CE16800-DEVICEA
		100GE1/0/2	TO-CE16800-DEVICEB
		100GE1/0/3	TO-CE16800-DEVICEE
		100GE1/0/4	TO-CE16800-DEVICEF
	VLAN：8 IP地址：10.1.8.105/24	100GE1/0/5	TO-ROUTERB
	VLAN：9 IP地址：172.16.6.2/24	100GE1/0/6	TO-FW-1

设备	VLAN及IP地址	接口编号	描述
	VLAN：10 IP地址：172.16.7.2/24	100GE1/0/7	TO-FW-1
	VLAN：11 IP地址：172.16.8.2/24	100GE1/0/8	TO-FW-2
	VLAN：12 IP地址：172.16.9.2/24	100GE1/0/9	TO-FW-2
	VLAN：13 IP地址：10.1.13.103/24	100GE1/0/14	TO-CE16800-DEVICEI
FW-1	172.16.1.1/24	GE1/0/1	TO-DEVICEI-Upstream
	172.16.2.1/24	GE1/0/2	TO-DEVICEI-Downstream
	172.16.3.1/24	GE1/0/3	TO-DEVICEJ-Upstream
	172.16.4.1/24	GE1/0/4	TO-DEVICEJ-Downstream
	172.16.5.1/24	Eth-Trunk1：GE2/0/0	TO-FW-2-HRP
		Eth-Trunk1：GE2/0/1	
		Eth-Trunk1：GE2/0/2	
		Eth-Trunk1：GE2/0/3	
	172.16.100.1/24	Loopback1	NA
	172.16.100.2/24	Loopback2	NA
	172.16.100.3/24	Loopback3	NA
	172.16.100.4/24	Loopback4	NA
FW-2	172.16.6.1/24	GE1/0/1	TO-DEVICEJ-Upstream
	172.16.7.1/24	GE1/0/2	TO-DEVICEJ-Downstream

设备	VLAN及IP地址	接口编号	描述
	172.16.8.1/24	GE1/0/3	TO-DEVICEI-Upstream
	172.16.9.1/24	GE1/0/4	TO-DEVICEI-Downstream
	172.16.10.1/24	Eth-Trunk1：GE2/0/0	TO-FW-1-HRP
		Eth-Trunk1：GE2/0/1	
		Eth-Trunk1：GE2/0/2	
		Eth-Trunk1：GE2/0/3	
	172.16.100.1/24	Loopback1	NA
	172.16.100.2/24	Loopback2	NA
	172.16.100.3/24	Loopback3	NA
	172.16.100.4/24	Loopback4	NA

配置思路

1. 通过在汇聚层设备DeviceA和DeviceB之间部署VRRP，实现链路冗余备份。
2. 通过在汇聚层设备DeviceA、汇聚层设备DeviceB和接入层设备DeviceC之间部署MSTP，消除网络中的环路。
3. 配置出口防火墙FW-1和FW-2双机热备，从核心层设备DeviceI或DeviceJ转发的流量经防火墙的安全策略处理，再分别流向数据中心或Internet。
4. 通过在汇聚层设备DeviceA、汇聚层设备DeviceB、核心层设备DeviceI和DeviceJ之间部署OSPF，实现网络三层互通。

操作步骤

步骤1 配置MSTP基本功能。

说明

只要两台设备的以下配置相同，这两台设备就属于同一个MST域。

- MST域的域名。
- 多生成树实例和VLAN的映射关系。
- MST域的修订级别。

1. 配置DeviceA、DeviceB、DeviceC到域名为RG1的域内，创建实例MSTI1和实例MSTI2。

   # 配置汇聚层设备DeviceA的MST域。

   ```
   <HUAWEI> system-view
   [~HUAWEI] sysname DeviceA
   ```

```
[*HUAWEI] commit
[~DeviceA] stp region-configuration
[~DeviceA-mst-region] region-name RG1
[*DeviceA-mst-region] instance 1 vlan 2
[*DeviceA-mst-region] instance 2 vlan 3
[*DeviceA-mst-region] commit
[~DeviceA-mst-region] quit
```

配置汇聚层设备DeviceB的MST域。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*HUAWEI] commit
[~DeviceB] stp region-configuration
[~DeviceB-mst-region] region-name RG1
[*DeviceB-mst-region] instance 1 vlan 2
[*DeviceB-mst-region] instance 2 vlan 3
[*DeviceB-mst-region] commit
[~DeviceB-mst-region] quit
```

配置接入层设备DeviceC的MST域。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceC
[*HUAWEI] commit
[~DeviceC] stp region-configuration
[~DeviceC-mst-region] region-name RG1
[*DeviceC-mst-region] instance 1 vlan 2
[*DeviceC-mst-region] commit
[~DeviceC-mst-region] quit
```

配置接入层设备DeviceD的MST域。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceD
[*HUAWEI] commit
[~DeviceD] stp region-configuration
[~DeviceD-mst-region] region-name RG1
[*DeviceD-mst-region] instance 2 vlan 3
[*DeviceD-mst-region] commit
[~DeviceD-mst-region] quit
```

2. 在域RG1内，配置MSTI1与MSTI2的根桥与备份根桥。
   - 配置MSTI1的根桥与备份根桥。

     # 配置汇聚层设备DeviceA为MSTI1的根桥。

     ```
     [~DeviceA] stp instance 1 root primary
     [*DeviceA] commit
     ```

     # 配置汇聚层设备DeviceB为MSTI1的备份根桥。

     ```
     [~DeviceB] stp instance 1 root secondary
     [*DeviceB] commit
     ```

   - 配置MSTI2的根桥与备份根桥。

     # 配置汇聚层设备DeviceB为MSTI2的根桥。

     ```
     [~DeviceB] stp instance 2 root primary
     [*DeviceB] commit
     ```

     # 配置汇聚层设备DeviceA为MSTI2的备份根桥。

     ```
     [~DeviceA] stp instance 2 root secondary
     [*DeviceA] commit
     ```

3. 配置实例MSTI1和MSTI2中将要被阻塞端口的路径开销值大于缺省值。

   说明

   - 端口路径开销值取值范围由路径开销计算方法决定，这里选择使用华为私有计算方法为例，配置实例MSTI1和MSTI2中将被阻塞端口的路径开销值为20000。
   - 同一网络内所有交换设备的端口路径开销应使用相同的计算方法。

   # 配置汇聚层设备DeviceA的端口路径开销的计算方法为华为私有计算方法。

```
[~DeviceA] stp pathcost-standard legacy
[*DeviceA] commit
```

配置汇聚层设备DeviceB的端口路径开销的计算方法为华为的私有计算方法。

```
[~DeviceB] stp pathcost-standard legacy
[*DeviceB] commit
```

配置接入层设备DeviceC的端口路径开销的计算方法为华为的私有计算方法，将端口100GE1/0/2在实例MSTI1中的路径开销值配置为20000。

```
[~DeviceC] stp pathcost-standard legacy
[*DeviceC] interface 100ge 1/0/2
[*DeviceC-100GE1/0/2] description TO-CE16800-DEVICEB
[*DeviceC-100GE1/0/2] stp instance 1 cost 20000
[*DeviceC-100GE1/0/2] commit
[~DeviceC-100GE1/0/2] quit
```

配置接入层设备DeviceD的端口路径开销的计算方法为华为的私有计算方法，将端口100GE1/0/2在实例MSTI2中的路径开销值配置为20000。

```
[~DeviceD] stp pathcost-standard legacy
[*DeviceD] interface 100ge 1/0/2
[*DeviceD-100GE1/0/2] description TO-CE16800-DEVICEA
[*DeviceD-100GE1/0/2] stp instance 2 cost 20000
[*DeviceD-100GE1/0/2] commit
[~DeviceD-100GE1/0/2] quit
```

4. 使能MSTP，实现破除环路。

   说明

   设备上MSTP功能默认使能。

   - 设备全局使能MSTP。

     # 在汇聚层设备DeviceA上启动MSTP。

     ```
     [~DeviceA] stp enable
     [*DeviceA] commit
     ```

     # 在汇聚层设备DeviceB上启动MSTP。

     ```
     [~DeviceB] stp enable
     [*DeviceB] commit
     ```

     # 在接入层设备DeviceC上启动MSTP。

     ```
     [~DeviceC] stp enable
     [*DeviceC] commit
     ```

     # 在接入层设备DeviceD上启动MSTP。

     ```
     [~DeviceD] stp enable
     [*DeviceD] commit
     ```

   - 将与Host相连的端口配置为边缘端口。

     # 配置接入层设备DeviceC端口100GE1/0/3为边缘端口。

     ```
     [~DeviceC] interface 100ge 1/0/3
     [*DeviceC-100GE1/0/3] description TO-HOSTA
     [*DeviceC-100GE1/0/3] stp edged-port enable
     [*DeviceC-100GE1/0/3] commit
     [~DeviceC-100GE1/0/3] quit
     ```

     # 配置接入层设备DeviceD端口100GE1/0/3为边缘端口。

     ```
     [~DeviceD] interface 100ge 1/0/3
     [*DeviceD-100GE1/0/3] description TO-HOSTB
     [*DeviceD-100GE1/0/3] stp edged-port enable
     [*DeviceD-100GE1/0/3] commit
     [~DeviceD-100GE1/0/3] quit
     ```

步骤2 配置保护功能，如在各实例的根桥设备的指定端口配置根保护功能。

在汇聚层设备DeviceA端口100GE1/0/1上启动根保护。

```
[~DeviceA] interface 100ge 1/0/1
[~DeviceA-100GE1/0/1] description TO-CE6800-DEVICEC
[*DeviceA-100GE1/0/1] stp root-protection
[*DeviceA-100GE1/0/1] commit
[~DeviceA-100GE1/0/1] quit
```

在汇聚层设备DeviceB端口100GE1/0/1上启动根保护。

```
[~DeviceB] interface 100ge 1/0/1
[~DeviceB-100GE1/0/1] description TO-CE6800-DEVICED
[*DeviceB-100GE1/0/1] stp root-protection
[*DeviceB-100GE1/0/1] commit
[~DeviceB-100GE1/0/1] quit
```

步骤3 配置处于环网中的设备的二层转发功能。

- 在交换设备DeviceA、DeviceB、DeviceC、DeviceD上创建VLAN2～3。

  # 在汇聚层设备DeviceA上创建VLAN2～3。

  ```
  [~DeviceA] vlan batch 2 to 3
  ```

  # 在汇聚层设备DeviceB上创建VLAN2～3。

  ```
  [~DeviceB] vlan batch 2 to 3
  ```

  # 在接入层设备DeviceC上创建VLAN2。

  ```
  [~DeviceC] vlan batch 2
  ```

  # 在接入层设备DeviceD上创建VLAN3。

  ```
  [~DeviceD] vlan batch 3
  ```

- 将交换设备上接入环路中的端口加入VLAN。

  # 将汇聚层设备DeviceA端口100GE1/0/1加入VLAN。

  ```
  [~DeviceA] interface 100ge 1/0/1
  [~DeviceA-100GE1/0/1] port link-type trunk
  [*DeviceA-100GE1/0/1] undo port trunk allow-pass vlan 1
  [*DeviceA-100GE1/0/1] port trunk allow-pass vlan 2
  [*DeviceA-100GE1/0/1] commit
  [~DeviceA-100GE1/0/1] quit
  ```

  # 将汇聚层设备DeviceA端口100GE1/0/2加入VLAN。

  ```
  [~DeviceA] interface 100ge 1/0/2
  [~DeviceA-100GE1/0/2] description TO-CE6800-DEVICED
  [*DeviceA-100GE1/0/2] port link-type trunk
  [*DeviceA-100GE1/0/2] port trunk allow-pass vlan 3
  [*DeviceA-100GE1/0/2] commit
  [~DeviceA-100GE1/0/2] quit
  ```

  # 将汇聚层设备DeviceA端口100GE1/0/3加入VLAN。

  ```
  [~DeviceA] interface 100ge 1/0/3
  [~DeviceA-100GE1/0/3] description TO-CE16800-DEVICEB
  [*DeviceA-100GE1/0/3] port link-type trunk
  [*DeviceA-100GE1/0/3] undo port trunk allow-pass vlan 1
  [*DeviceA-100GE1/0/3] port trunk allow-pass vlan 2 to 3
  [*DeviceA-100GE1/0/3] commit
  [~DeviceA-100GE1/0/3] quit
  ```

  # 将汇聚层设备DeviceB端口100GE1/0/1加入VLAN。

  ```
  [~DeviceB] interface 100ge 1/0/1
  [~DeviceB-100GE1/0/1] port link-type trunk
  [*DeviceB-100GE1/0/1] undo port trunk allow-pass vlan 1
  [*DeviceB-100GE1/0/1] port trunk allow-pass vlan 3
  [*DeviceB-100GE1/0/1] commit
  [~DeviceB-100GE1/0/1] quit
  ```

  # 将汇聚层设备DeviceB端口100GE1/0/2加入VLAN。

  ```
  [~DeviceB] interface 100ge 1/0/2
  [~DeviceB-100GE1/0/2] description TO-CE6800-DEVICEC
  [*DeviceB-100GE1/0/2] port link-type trunk
  ```



```
[*DeviceB-100GE1/0/2] undo port trunk allow-pass vlan 1
[*DeviceB-100GE1/0/2] port trunk allow-pass vlan 2
[*DeviceB-100GE1/0/2] commit
[~DeviceB-100GE1/0/2] quit
```

将汇聚层设备DeviceB端口100GE1/0/3加入VLAN。

```
[~DeviceB] interface 100ge 1/0/3
[~DeviceB-100GE1/0/3] description TO-CE16800-DEVICEA
[*DeviceB-100GE1/0/3] port link-type trunk
[*DeviceB-100GE1/0/3] undo port trunk allow-pass vlan 1
[*DeviceB-100GE1/0/3] port trunk allow-pass vlan 2 to 3
[*DeviceB-100GE1/0/3] commit
[~DeviceB-100GE1/0/3] quit
```

将接入层设备DeviceC端口100GE1/0/1加入VLAN。

```
[~DeviceC] interface 100ge 1/0/1
[~DeviceC-100GE1/0/1] description TO-CE16800-DEVICEA
[*DeviceC-100GE1/0/1] port link-type trunk
[*DeviceC-100GE1/0/1] undo port trunk allow-pass vlan 1
[*DeviceC-100GE1/0/1] port trunk allow-pass vlan 2
[*DeviceC-100GE1/0/1] commit
[~DeviceC-100GE1/0/1] quit
```

将接入层设备DeviceC端口100GE1/0/2加入VLAN。

```
[~DeviceC] interface 100ge 1/0/2
[~DeviceC-100GE1/0/2] port link-type trunk
[*DeviceC-100GE1/0/2] undo port trunk allow-pass vlan 1
[*DeviceC-100GE1/0/2] port trunk allow-pass vlan 2
[*DeviceC-100GE1/0/2] commit
[~DeviceC-100GE1/0/2] quit
```

将接入层设备DeviceC端口100GE1/0/3加入VLAN。

```
[~DeviceC] interface 100ge 1/0/3
[~DeviceC-100GE1/0/3] port link-type access
[*DeviceC-100GE1/0/3] port default vlan 2
[*DeviceC-100GE1/0/3] commit
[~DeviceC-100GE1/0/3] quit
```

将接入层设备DeviceD端口100GE1/0/1加入VLAN。

```
[~DeviceD] interface 100ge 1/0/1
[~DeviceD-100GE1/0/1] description TO-CE16800-DEVICEB
[*DeviceD-100GE1/0/1] port link-type trunk
[*DeviceD-100GE1/0/1] undo port trunk allow-pass vlan 1
[*DeviceD-100GE1/0/1] port trunk allow-pass vlan 3
[*DeviceD-100GE1/0/1] commit
[~DeviceD-100GE1/0/1] quit
```

将接入层设备DeviceD端口100GE1/0/2加入VLAN。

```
[~DeviceD] interface 100ge 1/0/2
[~DeviceD-100GE1/0/2] port link-type trunk
[*DeviceD-100GE1/0/2] undo port trunk allow-pass vlan 1
[*DeviceD-100GE1/0/2] port trunk allow-pass vlan 3
[*DeviceD-100GE1/0/2] commit
[~DeviceD-100GE1/0/2] quit
```

将接入层设备DeviceD端口100GE1/0/3加入VLAN。

```
[~DeviceD] interface 100ge 1/0/3
[~DeviceD-100GE1/0/3] port link-type access
[*DeviceD-100GE1/0/3] port default vlan 3
[*DeviceD-100GE1/0/3] commit
[~DeviceD-100GE1/0/3] quit
```

步骤4 配置VRRP备份组。

在汇聚层设备DeviceA和DeviceB上创建VRRP备份组1，配置DeviceA的优先级为120，抢占延时为20秒，作为Master设备；DeviceB的优先级为缺省值，作为Backup设备。

- DeviceA
```
[~DeviceA] interface vlanif 2
[*DeviceA-Vlanif2] vrrp vrid 1 virtual-ip 10.1.2.100
[*DeviceA-Vlanif2] vrrp vrid 1 priority 120
[*DeviceA-Vlanif2] vrrp vrid 1 preempt timer delay 20
[*DeviceA-Vlanif2] commit
[~DeviceA-Vlanif2] quit
```
- DeviceB
```
[~DeviceB] interface vlanif 2
[*DeviceB-Vlanif2] vrrp vrid 1 virtual-ip 10.1.2.100
[*DeviceB-Vlanif2] commit
[~DeviceB-Vlanif2] quit
```

在汇聚层设备DeviceA和DeviceB上创建VRRP备份组2，配置DeviceB的优先级为120，抢占延时为20秒，作为Master设备；DeviceA的优先级为缺省值，作为Backup设备。

- DeviceB
```
[~DeviceB] interface vlanif 3
[*DeviceB-Vlanif3] vrrp vrid 2 virtual-ip 10.1.3.100
[*DeviceB-Vlanif3] vrrp vrid 2 priority 120
[*DeviceB-Vlanif3] vrrp vrid 2 preempt timer delay 20
[*DeviceB-Vlanif3] commit
[~DeviceB-Vlanif3] quit
```
- DeviceA
```
[~DeviceA] interface vlanif 3
[*DeviceA-Vlanif3] vrrp vrid 2 virtual-ip 10.1.3.100
[*DeviceA-Vlanif3] commit
[~DeviceA-Vlanif3] quit
```

配置主机HostA的缺省网关为备份组1的虚拟IP地址10.1.2.100，配置主机HostB的缺省网关为备份组2的虚拟IP地址10.1.3.100。

步骤5 配置设备间的网络互连。

配置设备各接口的IP地址，以汇聚层设备DeviceA为例。DeviceB、DeviceI和DeviceJ的配置与之类似，详见配置脚本。

```
[~DeviceA] vlan batch 6 7
[*DeviceA] interface 100ge 1/0/4
[*DeviceA-100GE1/0/4] description TO-CE16800-DEVICEI
[*DeviceA-100GE1/0/4] port link-type trunk
[*DeviceA-100GE1/0/4] undo port trunk allow-pass vlan 1
[*DeviceA-100GE1/0/4] port trunk allow-pass vlan 6
[*DeviceA-100GE1/0/4] quit
[*DeviceA] interface 100ge 1/0/5
[*DeviceA-100GE1/0/5] description TO-CE16800-DEVICEJ
[*DeviceA-100GE1/0/5] port link-type trunk
[*DeviceA-100GE1/0/5] undo port trunk allow-pass vlan 1
[*DeviceA-100GE1/0/5] port trunk allow-pass vlan 7
[*DeviceA-100GE1/0/5] quit
[*DeviceA] interface vlanif 2
[*DeviceA-Vlanif2] ip address 10.1.2.102 24
[*DeviceA-Vlanif2] quit
[*DeviceA] interface vlanif 3
[*DeviceA-Vlanif3] ip address 10.1.3.102 24
[*DeviceA-Vlanif3] quit
[*DeviceA] interface vlanif 6
[*DeviceA-Vlanif6] ip address 10.1.6.102 24
[*DeviceA-Vlanif6] quit
[*DeviceA] interface vlanif 7
[*DeviceA-Vlanif7] ip address 10.1.7.102 24
[*DeviceA-Vlanif7] quit
[*DeviceA] commit
```

配置汇聚层设备DeviceA、汇聚层设备DeviceB、核心层设备DeviceI、核心层设备DeviceJ和出口路由器间采用OSPF协议进行互连。以汇聚层设备DeviceA为例。DeviceB、DeviceI和DeviceJ的配置与之类似，详见配置脚本。

```
[~DeviceA] ospf 1
[*DeviceA-ospf-1] area 0
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.2.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.3.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.6.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.7.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] quit
[*DeviceA-ospf-1] quit
[*DeviceA] commit
```

步骤6 配置防火墙。

配置FW-1和FW-2进行双机热备，从DeviceI、DeviceJ转发的报文经FW-1或FW-2的安全策略处理，再分别流向数据中心或Internet。

FW-1和FW-2进行负载分担，均同时转发流量，当一台FW故障时，业务可以平滑切换到另一台FW。

以下FW-1和FW-2以华为USG统一安全网关为例，介绍FW双机热备负载分担配置步骤。

1. 在出口防火墙FW-1上完成基础配置，包括配置设备名称、接口、安全区域等。

```
<USG> system-view
[USG] sysname FW-1
[FW-1] interface GigabitEthernet 1/0/1
[FW-1-GigabitEthernet1/0/1] ip address 172.16.1.1 24
[FW-1-GigabitEthernet1/0/1] quit
[FW-1] interface GigabitEthernet 1/0/2
[FW-1-GigabitEthernet1/0/2] ip address 172.16.2.1 24
[FW-1-GigabitEthernet1/0/2] quit
[FW-1] interface GigabitEthernet 1/0/3
[FW-1-GigabitEthernet1/0/3] ip address 172.16.3.1 24
[FW-1-GigabitEthernet1/0/3] quit
[FW-1] interface GigabitEthernet 1/0/4
[FW-1-GigabitEthernet1/0/4] ip address 172.16.4.1 24
[FW-1-GigabitEthernet1/0/4] quit

[FW-1] interface Eth-Trunk 1
[FW-1-Eth-Trunk1] trunkport GigabitEthernet 2/0/0 2/0/1 2/0/2 2/0/3
[FW-1-Eth-Trunk1] ip address 172.16.5.1 24
[FW-1-Eth-Trunk1] quit

[FW-1] firewall zone trust
[FW-1-zone-trust] add interface GigabitEthernet 1/0/1
[FW-1-zone-trust] add interface GigabitEthernet 1/0/3
[FW-1-zone-trust] quit
[FW-1] firewall zone untrust
[FW-1-zone-untrust] add interface GigabitEthernet 1/0/2
[FW-1-zone-untrust] add interface GigabitEthernet 1/0/4
[FW-1-zone-untrust] quit
[FW-1] firewall zone dmz
[FW-1-zone-dmz] add interface Eth-Trunk 1
[FW-1-zone-dmz] quit

[FW-1] interface LoopBack 1
[FW-1-LoopBack1] ip address 172.16.100.1 32
[FW-1-LoopBack1] quit
[FW-1] interface LoopBack 2
[FW-1-LoopBack2] ip address 172.16.100.2 32
[FW-1-LoopBack2] quit
[FW-1] interface LoopBack 3
[FW-1-LoopBack3] ip address 172.16.100.3 32
[FW-1-LoopBack3] quit
```

```
[FW-1] interface LoopBack 4
[FW-1-LoopBack4] ip address 172.16.100.4 32
[FW-1-LoopBack4] quit
```

2. 在出口防火墙FW-2上完成基础配置，包括配置设备名称、接口、安全区域等。

```
<USG> system-view
[USG] sysname FW-2
[FW-2] interface GigabitEthernet 1/0/1
[FW-2-GigabitEthernet1/0/1] ip address 172.16.6.1 24
[FW-2-GigabitEthernet1/0/1] quit
[FW-2] interface GigabitEthernet 1/0/2
[FW-2-GigabitEthernet1/0/2] ip address 172.16.7.1 24
[FW-2-GigabitEthernet1/0/2] quit
[FW-2] interface GigabitEthernet 1/0/3
[FW-2-GigabitEthernet1/0/3] ip address 172.16.8.1 24
[FW-2-GigabitEthernet1/0/3] quit
[FW-2] interface GigabitEthernet 1/0/4
[FW-2-GigabitEthernet1/0/4] ip address 172.16.9.1 24
[FW-2-GigabitEthernet1/0/4] quit

[FW-2] interface Eth-Trunk 1
[FW-2-Eth-Trunk1] trunkport GigabitEthernet 2/0/0 2/0/1 2/0/2 2/0/3
[FW-2-Eth-Trunk1] ip address 172.16.10.1 24
[FW-2-Eth-Trunk1] quit

[FW-2] firewall zone trust
[FW-2-zone-trust] add interface GigabitEthernet 1/0/1
[FW-2-zone-trust] add interface GigabitEthernet 1/0/3
[FW-2-zone-trust] quit
[FW-2] firewall zone untrust
[FW-2-zone-untrust] add interface GigabitEthernet 1/0/2
[FW-2-zone-untrust] add interface GigabitEthernet 1/0/4
[FW-2-zone-untrust] quit
[FW-2] firewall zone dmz
[FW-2-zone-dmz] add interface Eth-Trunk 1
[FW-2-zone-dmz] quit

[FW-2] interface LoopBack 1
[FW-2-LoopBack1] ip address 172.16.100.1 32
[FW-2-LoopBack1] quit
[FW-2] interface LoopBack 2
[FW-2-LoopBack2] ip address 172.16.100.2 32
[FW-2-LoopBack2] quit
[FW-2] interface LoopBack 3
[FW-2-LoopBack3] ip address 172.16.100.3 32
[FW-2-LoopBack3] quit
[FW-2] interface LoopBack 4
[FW-2-LoopBack4] ip address 172.16.100.4 32
[FW-2-LoopBack4] quit
```

3. 分别在出口防火墙FW-1、FW-2上配置OSPF。配置router-id时，需要为不同的进程指定不同的router-id。另外，主备防火墙也需要为OSPF进程指定不同的router-id，防止OSPF路由震荡。

```
[FW-1] ospf 1 router-id 172.16.100.1
[FW-1-ospf-1] area 0
[FW-1-ospf-1-area-0.0.0.0] network 172.16.1.0 0.0.0.255
[FW-1-ospf-1-area-0.0.0.0] network 172.16.100.1 0.0.0.0
[FW-1-ospf-1-area-0.0.0.0] quit
[FW-1-ospf-1] quit
[FW-1] ospf 2 router-id 172.16.100.2
[FW-1-ospf-2] area 0
[FW-1-ospf-2-area-0.0.0.0] network 172.16.2.0 0.0.0.255
[FW-1-ospf-2-area-0.0.0.0] network 172.16.100.2 0.0.0.0
[FW-1-ospf-2-area-0.0.0.0] quit
[FW-1-ospf-2] quit
[FW-1] ospf 3 router-id 172.16.100.3
[FW-1-ospf-3] area 0
[FW-1-ospf-3-area-0.0.0.0] network 172.16.3.0 0.0.0.255
[FW-1-ospf-3-area-0.0.0.0] network 172.16.100.3 0.0.0.0
```

```
[FW-1-ospf-3-area-0.0.0.0] quit
[FW-1-ospf-3] quit
[FW-1] ospf 4 router-id 172.16.100.4
[FW-1-ospf-4] area 0
[FW-1-ospf-4-area-0.0.0.0] network 172.16.4.0 0.0.0.255
[FW-1-ospf-4-area-0.0.0.0] network 172.16.100.4 0.0.0.0
[FW-1-ospf-4-area-0.0.0.0] quit
[FW-1-ospf-4] quit

[FW-2] ospf 1 router-id 172.16.100.6
[FW-2-ospf-1] area 0
[FW-2-ospf-1-area-0.0.0.0] network 172.16.6.0 0.0.0.255
[FW-2-ospf-1-area-0.0.0.0] network 172.16.100.1 0.0.0.0
[FW-2-ospf-1-area-0.0.0.0] quit
[FW-2-ospf-1] quit
[FW-2] ospf 2 router-id 172.16.100.7
[FW-2-ospf-2] area 0
[FW-2-ospf-2-area-0.0.0.0] network 172.16.7.0 0.0.0.255
[FW-2-ospf-2-area-0.0.0.0] network 172.16.100.2 0.0.0.0
[FW-2-ospf-2-area-0.0.0.0] quit
[FW-2-ospf-2] quit
[FW-2] ospf 3 router-id 172.16.100.8
[FW-2-ospf-3] area 0
[FW-2-ospf-3-area-0.0.0.0] network 172.16.8.0 0.0.0.255
[FW-2-ospf-3-area-0.0.0.0] network 172.16.100.3 0.0.0.0
[FW-2-ospf-3-area-0.0.0.0] quit
[FW-2-ospf-3] quit
[FW-2] ospf 4 router-id 172.16.100.9
[FW-2-ospf-4] area 0
[FW-2-ospf-4-area-0.0.0.0] network 172.16.9.0 0.0.0.255
[FW-2-ospf-4-area-0.0.0.0] network 172.16.100.4 0.0.0.0
[FW-2-ospf-4-area-0.0.0.0] quit
[FW-2-ospf-4] quit
```

4. 分别在出口防火墙FW-1、FW-2配置双机热备。
   - 在FW-1上配置双机热备。

     ```
     [FW-1] hrp track interface GigabitEthernet 1/0/1
     [FW-1] hrp track interface GigabitEthernet 1/0/2
     [FW-1] hrp track interface GigabitEthernet 1/0/3
     [FW-1] hrp track interface GigabitEthernet 1/0/4
     [FW-1] hrp adjust ospf-cost enable
     [FW-1] hrp interface Eth-Trunk 1 remote 172.16.10.1
     [FW-1] hrp enable
     [FW-1] hrp mirror session enable
     ```

   - 在FW-2上配置双机热备。

     ```
     [FW-2] hrp track interface GigabitEthernet 1/0/1
     [FW-2] hrp track interface GigabitEthernet 1/0/2
     [FW-2] hrp track interface GigabitEthernet 1/0/3
     [FW-2] hrp track interface GigabitEthernet 1/0/4
     [FW-2] hrp adjust ospf-cost enable
     [FW-2] hrp interface Eth-Trunk 1 remote 172.16.5.1
     [FW-2] hrp enable
     [FW-2] hrp mirror session enable
     ```

5. 配置安全策略和入侵防御。

   ```
   HRP_M[FW-1] policy interzone trust untrust outbound
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound] policy 1
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound-1] policy source 10.1.2.0 mask 24
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound-1] policy source 10.1.3.0 mask 24
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound-1] policy source 10.1.4.0 mask 24
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound-1] policy source 10.1.5.0 mask 24
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound-1] action permit
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound-1] profile ips default
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound-1] quit
   HRP_M[FW-1-policy-interzone-trust-untrust-outbound] quit
   HRP_M[FW-1] policy interzone trust untrust inbound
   HRP_M[FW-1-policy-interzone-trust-untrust-inbound] policy 1
   HRP_M[FW-1-policy-interzone-trust-untrust-inbound-1] policy destination 10.1.2.0 mask 24
   ```

```
HRP_M[FW-1-policy-interzone-trust-untrust-inbound-1] policy destination 10.1.3.0 mask 24
HRP_M[FW-1-policy-interzone-trust-untrust-inbound-1] policy destination 10.1.4.0 mask 24
HRP_M[FW-1-policy-interzone-trust-untrust-inbound-1] policy destination 10.1.5.0 mask 24
HRP_M[FW-1-policy-interzone-trust-untrust-inbound-1] policy service service-set ftp http
HRP_M[FW-1-policy-interzone-trust-untrust-inbound-1] action permit
HRP_M[FW-1-policy-interzone-trust-untrust-inbound-1] profile ips default
HRP_M[FW-1-policy-interzone-trust-untrust-inbound-1] quit
HRP_M[FW-1-policy-interzone-trust-untrust-inbound] quit
HRP_M[FW-1] ips enable
```

6. 配置攻击防范。

   说明

   本举例中的攻击防范阈值仅供参考，实际配置时，请管理员根据网络实际流量进行配置。

```
HRP_M[FW-1] firewall defend syn-flood enable
HRP_M[FW-1] firewall defend syn-flood enable
HRP_M[FW-1] firewall defend syn-flood zone untrust max-rate 20000
HRP_M[FW-1] firewall defend udp-flood enable
HRP_M[FW-1] firewall defend udp-flood zone untrust max-rate 1500
HRP_M[FW-1] firewall defend icmp-flood enable
HRP_M[FW-1] firewall defend icmp-flood zone untrust max-rate 20000
HRP_M[FW-1] firewall blacklist enable
HRP_M[FW-1] firewall defend ip-sweep enable
HRP_M[FW-1] firewall defend ip-sweep max-rate 4000
HRP_M[FW-1] firewall defend port-scan enable
HRP_M[FW-1] firewall defend port-scan max-rate 4000
HRP_M[FW-1] firewall defend ip-fragment enable
HRP_M[FW-1] firewall defend ip-spoofing enable
```

步骤7 配置策略路由将所有流经核心层设备DeviceI和DeviceJ的流量通过策略路由重定向到防火墙，防火墙对流量进行过滤。

以核心层设备DeviceI的配置为例，核心层设备DeviceJ配置与之类似，详见配置文件。

```
[~DeviceI] acl 3001
[*DeviceI-acl4-advance-3001] rule 5 permit ip source 10.1.2.0 24
[*DeviceI-acl4-advance-3001] rule 10 permit ip source 10.1.3.0 24
[*DeviceI-acl4-advance-3001] rule 15 permit ip source 10.1.4.0 24
[*DeviceI-acl4-advance-3001] rule 20 permit ip source 10.1.5.0 24
[*DeviceI-acl4-advance-3001] commit
[~DeviceI-acl4-advance-3001] quit
[~DeviceI] traffic classifier c1
[*DeviceI-classifier-c1] if-match acl 3001
[*DeviceI-classifier-c1] quit
[*DeviceI] commit
[~DeviceI] traffic behavior b1
[*DeviceI-behavior-b1] redirect load-balance nexthop 172.16.100.1 172.16.100.3
[*DeviceI-behavior-b1] quit
[*DeviceI] commit
[~DeviceI] traffic policy p1
[*DeviceI-trafficpolicy-p1] classifier c1 behavior b1
[*DeviceI-trafficpolicy-p1] quit
[*DeviceI] commit
[~DeviceI] interface 100ge 1/0/1
[~DeviceI-100GE1/0/1] traffic-policy p1 inbound
[*DeviceI-100GE1/0/1] quit
[*DeviceI] commit
[~DeviceI] interface 100ge 1/0/2
[~DeviceI-100GE1/0/2] traffic-policy p1 inbound
[*DeviceI-100GE1/0/2] quit
[*DeviceI] commit
[~DeviceI] interface 100ge 1/0/3
[~DeviceI-100GE1/0/3] traffic-policy p1 inbound
[*DeviceI-100GE1/0/3] quit
[*DeviceI] commit
[~DeviceI] interface 100ge 1/0/4
[~DeviceI-100GE1/0/4] traffic-policy p1 inbound
```

```
[*DeviceI-100GE1/0/4] quit
[*DeviceI] commit
[~DeviceI] interface 100ge 1/0/14
[~DeviceI-100GE1/0/14] traffic-policy p1 inbound
[*DeviceI-100GE1/0/14] quit
[*DeviceI] commit
[~DeviceI] acl 3003
[*DeviceI-acl4-advance-3003] rule 5 permit ip destination 10.1.2.0 24
[*DeviceI-acl4-advance-3003] rule 10 permit ip destination 10.1.3.0 24
[*DeviceI-acl4-advance-3003] rule 15 permit ip destination 10.1.4.0 24
[*DeviceI-acl4-advance-3003] rule 20 permit ip destination 10.1.5.0 24
[*DeviceI-acl4-advance-3003] commit
[~DeviceI-acl4-advance-3003] quit
[~DeviceI] traffic classifier c3
[*DeviceI-classifier-c3] if-match acl 3003
[*DeviceI-classifier-c3] quit
[*DeviceI] commit
[~DeviceI] traffic behavior b3
[*DeviceI-behavior-b3] redirect load-balance nexthop 172.16.100.2 172.16.100.4
[*DeviceI-behavior-b3] quit
[*DeviceI] commit
[~DeviceI] traffic policy p2
[*DeviceI-trafficpolicy-p2] classifier c3 behavior b3
[*DeviceI-trafficpolicy-p2] quit
[*DeviceI] commit
[~DeviceI] interface 100ge 1/0/5
[~DeviceI-100GE1/0/5] traffic-policy p2 inbound
[*DeviceI-100GE1/0/5] quit
[*DeviceI] commit
```

----结束

检查配置结果

1. 完成上述配置后，在汇聚层设备DeviceA上执行**display vrrp**命令，可以看到DeviceA在备份组1中作为Master设备，在备份组2中作为Backup设备。

```
<DeviceA> display vrrp verbose
Vlanif2 | Virtual Router 1
State         : Master
Virtual IP    : 10.1.2.100
Master IP     : 10.1.2.102
PriorityRun   : 120
PriorityConfig : 120
MasterPriority : 120
Preempt       : YES   Delay Time : 20s   Remain : --
Hold Multiplier: 3
TimerRun      : 1s
TimerConfig   : 1s
Auth Type     : NONE
Virtual MAC   : 0000-5e00-0101
Check TTL     : YES
Config Type   : Normal
Create Time       : 2023-04-14 09:57:22
Last Change Time  : 2023-04-14 09:58:37

Vlanif3 | Virtual Router 2
State         : Backup
Virtual IP    : 10.1.3.100
Master IP     : 10.1.3.103
PriorityRun   : 100
PriorityConfig : 100
MasterPriority : 120
Preempt       : YES   Delay Time : 0s   Remain : --
Hold Multiplier: 3
TimerRun      : 1s
TimerConfig   : 1s
Auth Type     : NONE
Virtual MAC   : 0000-5e00-0102
```

```
Check TTL      : YES
Config Type    : Normal
Create Time       : 2023-04-14 09:57:22
Last Change Time  : 2023-04-14 10:38:00
```

2. 在汇聚层设备DeviceB上执行**display vrrp**命令，可以看到DeviceB在备份组1中作为Backup设备，在备份组2中作为Master设备。

```
<DeviceB> display vrrp verbose
Vlanif2 | Virtual Router 1
State          : Backup
Virtual IP     : 10.1.2.100
Master IP      : 10.1.2.102
PriorityRun    : 100
PriorityConfig : 100
MasterPriority : 120
Preempt        : YES   Delay Time : 0s   Remain : --
Hold Multiplier: 3
TimerRun       : 1s
TimerConfig    : 1s
Auth Type      : NONE
Virtual MAC    : 0000-5e00-0101
Check TTL      : YES
Config Type    : Normal
Create Time       : 2023-04-14 10:00:37
Last Change Time  : 2023-04-14 10:30:13

Vlanif3 | Virtual Router 2
State          : Master
Virtual IP     : 10.1.3.100
Master IP      : 10.1.3.103
PriorityRun    : 120
PriorityConfig : 120
MasterPriority : 120
Preempt        : YES   Delay Time : 20s   Remain : --
Hold Multiplier: 3
TimerRun       : 1s
TimerConfig    : 1s
Auth Type      : NONE
Virtual MAC    : 0000-5e00-0102
Check TTL      : YES
Config Type    : Normal
Create Time       : 2023-04-14 10:00:37
Last Change Time  : 2023-04-14 10:30:34
```

配置脚本

- 汇聚层设备DeviceA的配置脚本

```
#
sysname DeviceA
#
vlan batch 2 to 3 6 to 7
#
stp instance 1 root primary
stp instance 2 root secondary
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
#
interface Vlanif2
 ip address 10.1.2.102 255.255.255.0
 vrrp vrid 1 virtual-ip 10.1.2.100
 vrrp vrid 1 priority 120
 vrrp vrid 1 preempt timer delay 20
#
interface Vlanif3
```

```
 ip address 10.1.3.102 255.255.255.0
 vrrp vrid 2 virtual-ip 10.1.3.100
#
interface Vlanif6
 ip address 10.1.6.102 255.255.255.0
#
interface Vlanif7
 ip address 10.1.7.102 255.255.255.0
#
interface 100GE1/0/1
 description TO-CE6800-DEVICEC
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2
 stp root-protection
#
interface 100GE1/0/2
 description TO-CE6800-DEVICED
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3
#
interface 100GE1/0/3
 description TO-CE16800-DEVICEB
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2 to 3
#
interface 100GE1/0/4
 description TO-CE16800-DEVICEI
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 6
#
interface 100GE1/0/5
 description TO-CE16800-DEVICEJ
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 7
#
ospf 1
 area 0.0.0.0
  network 10.1.2.0 0.0.0.255
  network 10.1.3.0 0.0.0.255
  network 10.1.6.0 0.0.0.255
  network 10.1.7.0 0.0.0.255
#
return
```

- 汇聚层设备DeviceB的配置脚本

```
#
sysname DeviceB
#
vlan batch 2 to 3 6 to 7
#
stp instance 1 root secondary
stp instance 2 root primary
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
#
interface Vlanif2
 ip address 10.1.2.103 255.255.255.0
 vrrp vrid 1 virtual-ip 10.1.2.100
#
interface Vlanif3
 ip address 10.1.3.103 255.255.255.0
```



```
 vrrp vrid 2 virtual-ip 10.1.3.100
 vrrp vrid 2 priority 120
 vrrp vrid 2 preempt timer delay 20
#
interface Vlanif6
 ip address 10.1.6.103 255.255.255.0
#
interface Vlanif7
 ip address 10.1.7.103 255.255.255.0
#
interface 100GE1/0/1
 description TO-CE6800-DEVICED
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3
 stp root-protection
#
interface 100GE1/0/2
 description TO-CE6800-DEVICEC
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2
#
interface 100GE1/0/3
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2 to 3
#
interface 100GE1/0/4
 description TO-CE16800-DEVICEI
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 6
#
interface 100GE1/0/5
 description TO-CE16800-DEVICEJ
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 7
#
ospf 1
 area 0.0.0.0
  network 10.1.2.0 0.0.0.255
  network 10.1.3.0 0.0.0.255
  network 10.1.6.0 0.0.0.255
  network 10.1.7.0 0.0.0.255
#
return
```

- 接入层设备DeviceC的配置脚本

```
#
sysname DeviceC
#
vlan batch 2
#
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
#
interface 100GE1/0/1
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2
#
interface 100GE1/0/2
 description TO-CE16800-DEVICEB
```

```
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2
 stp instance 1 cost 20000
#
interface 100GE1/0/3
 description TO-HOSTA
 port default vlan 2
 stp disable
#
return
```

- 接入层设备DeviceD的配置脚本

```
#
sysname DeviceD
#
vlan batch 3
#
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 2 vlan 3
#
interface 100GE1/0/1
 description TO-CE16800-DEVICEB
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3
#
interface 100GE1/0/2
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3
 stp instance 2 cost 20000
#
interface 100GE1/0/3
 description TO-HOSTB
 port default vlan 3
 stp disable
#
return
```

- 核心层设备DeviceI的配置脚本

```
#
sysname DeviceI
#
vlan batch 6 8 to 13
#
acl number 3001
 rule 5 permit ip source 10.1.2.0 0.0.0.255
 rule 10 permit ip source 10.1.3.0 0.0.0.255
 rule 15 permit ip source 10.1.4.0 0.0.0.255
 rule 20 permit ip source 10.1.5.0 0.0.0.255
#
acl number 3003
 rule 5 permit ip destination 10.1.2.0 0.0.0.255
 rule 10 permit ip destination 10.1.3.0 0.0.0.255
 rule 15 permit ip destination 10.1.4.0 0.0.0.255
 rule 20 permit ip destination 10.1.5.0 0.0.0.255
#
traffic classifier c1 type or
 if-match acl 3001
#
traffic classifier c3 type or
 if-match acl 3003
#
traffic behavior b1
 redirect load-balance nexthop 172.16.100.1 172.16.100.3
```



```
#
traffic behavior b3
 redirect load-balance nexthop 172.16.100.2 172.16.100.4
#
traffic policy p1
 classifier c1 behavior b1 precedence 5
#
traffic policy p2
 classifier c3 behavior b3 precedence 5
#
interface Vlanif6
 ip address 10.1.6.104 255.255.255.0
#
interface Vlanif8
 ip address 10.1.8.104 255.255.255.0
#
interface Vlanif9
 ip address 172.16.1.2 255.255.255.0
#
interface Vlanif10
 ip address 172.16.2.2 255.255.255.0
#
interface Vlanif11
 ip address 172.16.3.2 255.255.255.0
#
interface Vlanif12
 ip address 172.16.4.2 255.255.255.0
#
interface Vlanif13
 ip address 10.1.13.102 255.255.255.0
#
interface 100GE1/0/1
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 6
 traffic-policy p1 inbound
#
interface 100GE1/0/2
 description TO-CE16800-DEVICEB
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 6
 traffic-policy p1 inbound
#
interface 100GE1/0/3
 description TO-CE16800-DEVICEE
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 6
 traffic-policy p1 inbound
#
interface 100GE1/0/4
 description TO-CE16800-DEVICEF
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 6
 traffic-policy p1 inbound
#
interface 100GE1/0/5
 description TO-ROUTERA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 8
 traffic-policy p2 inbound
#
interface 100GE1/0/6
 description TO-FW-1
 port link-type trunk
```

```
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 9
#
interface 100GE1/0/7
 description TO-FW-1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10
#
interface 100GE1/0/8
 description TO-FW-2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
#
interface 100GE1/0/9
 description TO-FW-2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 12
#
interface 100GE1/0/14
 description TO-CE16800-DEVICEJ
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 13
 traffic-policy p1 inbound
#
ospf 1
 area 0.0.0.0
  network 10.1.6.0 0.0.0.255
  network 10.1.8.0 0.0.0.255
  network 10.1.13.0 0.0.0.255
  network 172.16.1.0 0.0.0.255
  network 172.16.2.0 0.0.0.255
  network 172.16.3.0 0.0.0.255
  network 172.16.4.0 0.0.0.255
#
return
```

- 核心层设备DeviceJ的配置脚本

```
#
sysname DeviceJ
#
vlan batch 7 to 13
#
acl number 3001
 rule 5 permit ip source 10.1.2.0 0.0.0.255
 rule 10 permit ip source 10.1.3.0 0.0.0.255
 rule 15 permit ip source 10.1.4.0 0.0.0.255
 rule 20 permit ip source 10.1.5.0 0.0.0.255
#
acl number 3003
 rule 5 permit ip destination 10.1.2.0 0.0.0.255
 rule 10 permit ip destination 10.1.3.0 0.0.0.255
 rule 15 permit ip destination 10.1.4.0 0.0.0.255
 rule 20 permit ip destination 10.1.5.0 0.0.0.255
#
traffic classifier c1 type or
 if-match acl 3001
#
traffic classifier c3 type or
 if-match acl 3003
#
traffic behavior b1
 redirect load-balance nexthop 172.16.100.1 172.16.100.3
#
traffic behavior b3
 redirect load-balance nexthop 172.16.100.2 172.16.100.4
#
```

```
traffic policy p1
 classifier c1 behavior b1 precedence 5
#
traffic policy p2
 classifier c3 behavior b3 precedence 5
#
interface Vlanif7
 ip address 10.1.7.105 255.255.255.0
#
interface Vlanif8
 ip address 10.1.8.105 255.255.255.0
#
interface Vlanif9
 ip address 172.16.6.2 255.255.255.0
#
interface Vlanif10
 ip address 172.16.7.2 255.255.255.0
#
interface Vlanif11
 ip address 172.16.8.2 255.255.255.0
#
interface Vlanif12
 ip address 172.16.9.2 255.255.255.0
#
interface Vlanif13
 ip address 10.1.13.103 255.255.255.0
#
interface 100GE1/0/1
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 7
 traffic-policy p1 inbound
#
interface 100GE1/0/2
 description TO-CE16800-DEVICEB
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 7
 traffic-policy p1 inbound
#
interface 100GE1/0/3
 description TO-CE16800-DEVICEE
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 7
 traffic-policy p1 inbound
#
interface 100GE1/0/4
 description TO-CE16800-DEVICEF
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 7
 traffic-policy p1 inbound
#
interface 100GE1/0/5
 description TO-ROUTERB
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 8
 traffic-policy p2 inbound
#
interface 100GE1/0/6
 description TO-FW-1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 9
#
interface 100GE1/0/7
```

```
 description TO-FW-1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10
#
interface 100GE1/0/8
 description TO-FW-2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
#
interface 100GE1/0/9
 description TO-FW-2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 12
#
interface 100GE1/0/14
 description TO-CE16800-DEVICEI
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 13
 traffic-policy p1 inbound
#
ospf 1
 area 0.0.0.0
  network 10.1.7.0 0.0.0.255
  network 10.1.8.0 0.0.0.255
  network 10.1.13.0 0.0.0.255
  network 172.16.6.0 0.0.0.255
  network 172.16.7.0 0.0.0.255
  network 172.16.8.0 0.0.0.255
  network 172.16.9.0 0.0.0.255
#
return
```

- 出口防火墙FW-1的配置脚本

```
#
 sysname FW-1
#
 firewall packet-filter default permit interzone local dmz direction inbound
 firewall packet-filter default permit interzone local dmz direction outbound
#
 firewall defend port-scan enable
 firewall defend ip-sweep enable
 firewall defend ip-fragment enable
 firewall defend icmp-flood enable
 firewall defend udp-flood enable
 firewall defend syn-flood enable
 firewall defend ip-spoofing enable
 firewall defend action discard
 firewall defend icmp-flood zone untrust max-rate 20000
 firewall defend udp-flood zone untrust max-rate 1500
 firewall defend syn-flood zone untrust max-rate 20000
#
 hrp enable
 hrp adjust ospf-cost enable
 hrp interface Eth-Trunk1 remote 172.16.10.1
 hrp mirror session enable
 hrp track interface GigabitEthernet 1/0/1
 hrp track interface GigabitEthernet 1/0/2
 hrp track interface GigabitEthernet 1/0/3
 hrp track interface GigabitEthernet 1/0/4
#
 ips enable
#
interface Eth-Trunk1
 ip address 172.16.5.1 255.255.255.0
#
interface GigabitEthernet1/0/1
```

```
 description TO-CE16800-DeviceI-Upstream
 ip address 172.16.1.1 255.255.255.0
#
interface GigabitEthernet1/0/2
 description TO-CE16800-DeviceI-Downstream
 ip address 172.16.2.1 255.255.255.0
#
interface GigabitEthernet1/0/3
 description TO-CE16800-DeviceJ-Upstream
 ip address 172.16.3.1 255.255.255.0
#
interface GigabitEthernet1/0/4
 description TO-CE16800-DeviceJ-Downstream
 ip address 172.16.4.1 255.255.255.0
#
interface GigabitEthernet2/0/0
 description TO-FW-2-HRP
 eth-trunk 1
#
interface GigabitEthernet2/0/1
 description TO-FW-2-HRP
 eth-trunk 1
#
interface GigabitEthernet2/0/2
 description TO-FW-2-HRP
 eth-trunk 1
#
interface GigabitEthernet2/0/3
 description TO-FW-2-HRP
 eth-trunk 1
#
interface LoopBack 1
 ip address 172.16.100.1 32
#
interface LoopBack 2
 ip address 172.16.100.2 32
#
interface LoopBack 3
 ip address 172.16.100.3 32
#
interface LoopBack 4
 ip address 172.16.100.4 32
#
profile type ips name default
 signature-set name default
  os both
  target both
  severity low medium high
  protocol all
  category all
#
firewall zone trust
 set priority 85
 add interface GigabitEthernet 1/0/1
 add interface GigabitEthernet 1/0/3
#
firewall zone untrust
 set priority 5
 add interface GigabitEthernet 1/0/2
 add interface GigabitEthernet 1/0/4
#
firewall zone dmz
 set priority 50
 add interface Eth-Trunk1
#
firewall interzone trust untrust
 detect ftp
#
policy interzone trust untrust inbound
```

```
 policy 1
  action permit
  profile ips default
  policy service service-set ftp
  policy service service-set http
  policy destination 10.1.2.0 mask 24
  policy destination 10.1.3.0 mask 24
  policy destination 10.1.4.0 mask 24
  policy destination 10.1.5.0 mask 24
#
policy interzone trust untrust outbound
 policy 1
  action permit
  profile ips default
  policy source 10.1.2.0 mask 24
  policy source 10.1.3.0 mask 24
  policy source 10.1.4.0 mask 24
  policy source 10.1.5.0 mask 24
#
ospf 1 router-id 172.16.100.1
 area 0.0.0.0
  network 172.16.1.0 0.0.0.255
  network 172.16.100.1 0.0.0.0
#
ospf 2 router-id 172.16.100.2
 area 0.0.0.0
  network 172.16.2.0 0.0.0.255
  network 172.16.100.2 0.0.0.0
#
ospf 3 router-id 172.16.100.3
 area 0.0.0.0
  network 172.16.3.0 0.0.0.255
  network 172.16.100.3 0.0.0.0
#
ospf 4 router-id 172.16.100.4
 area 0.0.0.0
  network 172.16.4.0 0.0.0.255
  network 172.16.100.4 0.0.0.0
#
return
```

- 出口防火墙FW-2的配置脚本

```
#
 sysname FW-2
#
 firewall packet-filter default permit interzone local dmz direction inbound
 firewall packet-filter default permit interzone local dmz direction outbound
#
 firewall defend port-scan enable
 firewall defend ip-sweep enable
 firewall defend ip-fragment enable
 firewall defend icmp-flood enable
 firewall defend udp-flood enable
 firewall defend syn-flood enable
 firewall defend ip-spoofing enable
 firewall defend action discard
 firewall defend icmp-flood zone untrust max-rate 20000
 firewall defend udp-flood zone untrust max-rate 1500
 firewall defend syn-flood zone untrust max-rate 20000
#
 hrp enable
 hrp adjust ospf-cost enable
 hrp interface Eth-Trunk1 remote 172.16.5.1
 hrp mirror session enable
 hrp track interface GigabitEthernet 1/0/1
 hrp track interface GigabitEthernet 1/0/2
 hrp track interface GigabitEthernet 1/0/3
 hrp track interface GigabitEthernet 1/0/4
#
 ips enable
```

```
#
interface Eth-Trunk1
 ip address 172.16.10.1 255.255.255.0
#
interface GigabitEthernet1/0/1
 description TO-CE16800-DeviceI-Upstream
 ip address 172.16.6.1 255.255.255.0
#
interface GigabitEthernet1/0/2
 description TO-CE16800-DeviceI-Downstream
 ip address 172.16.7.1 255.255.255.0
#
interface GigabitEthernet1/0/3
 description TO-CE16800-DeviceJ-Upstream
 ip address 172.16.8.1 255.255.255.0
#
interface GigabitEthernet1/0/4
 description TO-CE16800-DeviceJ-Downstream
 ip address 172.16.9.1 255.255.255.0
#
interface GigabitEthernet2/0/0
 description TO-FW-1-HRP
 eth-trunk 1
#
interface GigabitEthernet2/0/1
 description TO-FW-1-HRP
 eth-trunk 1
#
interface GigabitEthernet2/0/2
 description TO-FW-1-HRP
 eth-trunk 1
#
interface GigabitEthernet2/0/3
 description TO-FW-1-HRP
 eth-trunk 1
#
interface LoopBack 1
 ip address 172.16.100.1 32
#
interface LoopBack 2
 ip address 172.16.100.2 32
#
interface LoopBack 3
 ip address 172.16.100.3 32
#
interface LoopBack 4
 ip address 172.16.100.4 32
#
profile type ips name default
 signature-set name default
  os both
  target both
  severity low medium high
  protocol all
  category all
#
firewall zone trust
 set priority 85
 add interface GigabitEthernet 1/0/1
 add interface GigabitEthernet 1/0/3
#
firewall zone untrust
 set priority 5
 add interface GigabitEthernet 1/0/2
 add interface GigabitEthernet 1/0/4
#
firewall zone dmz
 set priority 50
 add interface Eth-Trunk1
```

```
#
firewall interzone trust untrust
 detect ftp
#
policy interzone trust untrust inbound
 policy 1
  action permit
  profile ips default
  policy service service-set ftp
  policy service service-set http
  policy destination 10.1.2.0 mask 24
  policy destination 10.1.3.0 mask 24
  policy destination 10.1.4.0 mask 24
  policy destination 10.1.5.0 mask 24
#
policy interzone trust untrust outbound
 policy 1
  action permit
  profile ips default
  policy source 10.1.2.0 mask 24
  policy source 10.1.3.0 mask 24
  policy source 10.1.4.0 mask 24
  policy source 10.1.5.0 mask 24
#
ospf 1 router-id 172.16.100.6
 area 0.0.0.0
  network 172.16.6.0 0.0.0.255
  network 172.16.100.1 0.0.0.0
#
ospf 2 router-id 172.16.100.7
 area 0.0.0.0
  network 172.16.7.0 0.0.0.255
  network 172.16.100.2 0.0.0.0
#
ospf 3 router-id 172.16.100.8
 area 0.0.0.0
  network 172.16.8.0 0.0.0.255
  network 172.16.100.3 0.0.0.0
#
ospf 4 router-id 172.16.100.9
 area 0.0.0.0
  network 172.16.9.0 0.0.0.255
  network 172.16.100.4 0.0.0.0
#
return
```

1.2 基于 VRRP 的二层架构数据中心网络部署举例

适用产品和版本

- CloudEngine系列交换机V300R020C00或更高版本。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

在数据中心场景中，接入层交换机以双上行方式接入核心层。用户希望：

- 考虑到业务的可靠性，部署冗余链路，在一条上行链路断开的时候，流量能切换到另外一条上行链路转发。
- 避免冗余备份链路导致的环网问题，消除网络中的环路。

图 1-2 基于 VRRP 的二层架构数据中心网络组网

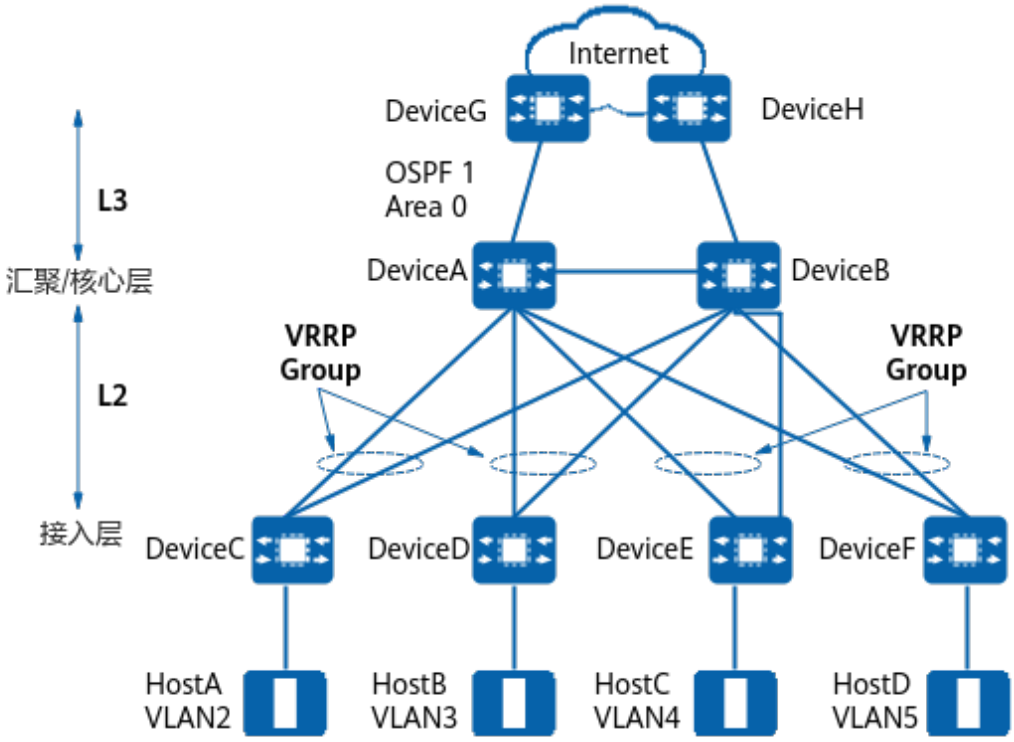


表 1-2 数据准备表

设备	VLAN及IP地址	接口	描述
DeviceA	VLAN：2 IP地址：10.1.2.102/24 虚拟IP地址：10.1.2.100	100GE1/0/1	TO-CE6800-DEVICEC
		100GE1/0/5	TO-CE16800-DEVICEB
	VLAN：3 IP地址：10.1.3.102/24 虚拟IP地址：10.1.3.100	100GE1/0/2	TO-CE6800-DEVICED
		100GE1/0/5	TO-CE16800-DEVICEB
	VLAN：4 IP地址：10.1.4.102/24 虚拟IP地址：10.1.4.100	100GE1/0/3	TO-CE6800-DEVICEE
		100GE1/0/5	TO-CE16800-DEVICEB
	VLAN：5 IP地址：10.1.5.102/24 虚拟IP地址：10.1.5.100	100GE1/0/4	TO-CE6800-DEVICEF
		100GE1/0/5	TO-CE16800-DEVICEB

设备	VLAN及IP地址	接口	描述
	VLAN：6 IP地址：10.1.6.103/24	100GE1/0/6	TO-DEVICEG
DeviceB	VLAN：2 IP地址：10.1.2.103/24 虚拟IP地址：10.1.2.100	100GE1/0/4	TO-CE6800-DEVICEC
		100GE1/0/5	TO-CE16800-DEVICEA
	VLAN：3 IP地址：10.1.3.103/24 虚拟IP地址：10.1.3.100	100GE1/0/3	TO-CE6800-DEVICED
		100GE1/0/5	TO-CE16800-DEVICEA
	VLAN：4 IP地址：10.1.4.103/24 虚拟IP地址：10.1.4.100	100GE1/0/2	TO-CE6800-DEVICEE
		100GE1/0/5	TO-CE16800-DEVICEA
	VLAN：5 IP地址：10.1.5.103/24 虚拟IP地址：10.1.5.100	100GE1/0/1	TO-CE6800-DEVICEF
		100GE1/0/5	TO-CE16800-DEVICEA
	VLAN：7 IP地址：10.1.7.103/24	100GE1/0/6	TO-DEVICEH
DeviceC	VLAN：2	100GE1/0/1	TO-CE16800-DEVICEA
		100GE1/0/2	TO-CE16800-DEVICEB
		100GE1/0/3	TO-HOSTA
DeviceD	VLAN：3	100GE1/0/1	TO-CE16800-DEVICEB
		100GE1/0/2	TO-CE16800-DEVICEA
		100GE1/0/3	TO-HOSTB
DeviceE	VLAN：4	100GE1/0/1	TO-CE16800-DEVICEA

设备	VLAN及IP地址	接口	描述
		100GE1/0/2	TO-CE16800-DEVICEB
		100GE1/0/3	TO-HOSTC
DeviceF	VLAN：5	100GE1/0/1	TO-CE16800-DEVICEB
		100GE1/0/2	TO-CE16800-DEVICEA
		100GE1/0/3	TO-HOSTD

配置思路

- 通过在核心层设备DeviceA和DeviceB之间部署VRRP，实现链路冗余备份。
- 通过在接入层和核心层设备之间部署MSTP，消除网络中的环路。

操作步骤

步骤1 配置MSTP基本功能（以DeviceA、DeviceB、DeviceC和DeviceD为例）

说明

只要两台设备的以下配置相同，这两台设备就属于同一个MST域。

- MST域的域名。
- 多生成树实例和VLAN的映射关系。
- MST域的修订级别。

1. 配置DeviceA、DeviceB、DeviceC、DeviceD到域名为RG1的域内，创建实例MSTI1和实例MSTI2

   # 配置核心层设备DeviceA的MST域。

   ```
   <HUAWEI> system-view
   [~HUAWEI] sysname DeviceA
   [*HUAWEI] commit
   [~DeviceA] stp region-configuration
   [~DeviceA-mst-region] region-name RG1
   [*DeviceA-mst-region] instance 1 vlan 2
   [*DeviceA-mst-region] instance 2 vlan 3
   [*DeviceA-mst-region] commit
   [~DeviceA-mst-region] quit
   ```

   # 配置核心层设备DeviceB的MST域。

   ```
   <HUAWEI> system-view
   [~HUAWEI] sysname DeviceB
   [*HUAWEI] commit
   [~DeviceB] stp region-configuration
   [~DeviceB-mst-region] region-name RG1
   [*DeviceB-mst-region] instance 1 vlan 2
   [*DeviceB-mst-region] instance 2 vlan 3
   [*DeviceB-mst-region] commit
   [~DeviceB-mst-region] quit
   ```

   # 配置接入层设备DeviceC的MST域。

   ```
   <HUAWEI> system-view
   [~HUAWEI] sysname DeviceC
   ```

```
[*HUAWEI] commit
[~DeviceC] stp region-configuration
[~DeviceC-mst-region] region-name RG1
[*DeviceC-mst-region] instance 1 vlan 2
[*DeviceC-mst-region] instance 2 vlan 3
[*DeviceC-mst-region] commit
[~DeviceC-mst-region] quit
```

配置接入层设备DeviceD的MST域。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceD
[*HUAWEI] commit
[~DeviceD] stp region-configuration
[~DeviceD-mst-region] region-name RG1
[*DeviceD-mst-region] instance 1 vlan 2
[*DeviceD-mst-region] instance 2 vlan 3
[*DeviceD-mst-region] commit
[~DeviceD-mst-region] quit
```

2. 在域RG1内，配置MSTI1与MSTI2的根桥与备份根桥
   - 配置MSTI1的根桥与备份根桥

     # 配置核心层设备DeviceA为MSTI1的根桥。

     ```
     [~DeviceA] stp instance 1 root primary
     [*DeviceA] commit
     ```

     # 配置核心层设备DeviceB为MSTI1的备份根桥。

     ```
     [~DeviceB] stp instance 1 root secondary
     [*DeviceB] commit
     ```

   - 配置MSTI2的根桥与备份根桥

     # 配置核心层设备DeviceB为MSTI2的根桥。

     ```
     [~DeviceB] stp instance 2 root primary
     [*DeviceB] commit
     ```

     # 配置核心层设备DeviceA为MSTI2的备份根桥。

     ```
     [~DeviceA] stp instance 2 root secondary
     [*DeviceA] commit
     ```

3. 配置实例MSTI1和MSTI2中将要被阻塞端口的路径开销值大于缺省值

   说明

   - 端口路径开销值取值范围由路径开销计算方法决定，这里选择使用华为私有计算方法为例，配置实例MSTI1和MSTI2中将被阻塞端口的路径开销值为20000。
   - 同一网络内所有交换设备的端口路径开销应使用相同的计算方法。

   # 配置核心层设备DeviceA的端口路径开销的计算方法为华为私有计算方法。

   ```
   [~DeviceA] stp pathcost-standard legacy
   [*DeviceA] commit
   ```

   # 配置核心层设备DeviceB的端口路径开销的计算方法为华为的私有计算方法。

   ```
   [~DeviceB] stp pathcost-standard legacy
   [*DeviceB] commit
   ```

   # 配置接入层设备DeviceC的端口路径开销的计算方法为华为的私有计算方法，将端口100GE1/0/2在实例MSTI1中的路径开销值配置为20000。

   ```
   [~DeviceC] stp pathcost-standard legacy
   [*DeviceC] interface 100ge 1/0/2
   [*DeviceC-100GE1/0/2] description TO-CE16800-DEVICEB
   [*DeviceC-100GE1/0/2] stp instance 1 cost 20000
   [*DeviceC-100GE1/0/2] commit
   [~DeviceC-100GE1/0/2] quit
   ```

   # 配置接入层设备DeviceD的端口路径开销的计算方法为华为的私有计算方法，将端口100GE1/0/2在实例MSTI2中的路径开销值配置为20000。

```
[~DeviceD] stp pathcost-standard legacy
[*DeviceD] interface 100ge 1/0/2
[*DeviceD-100GE1/0/2] description TO-CE16800-DEVICEA
[*DeviceD-100GE1/0/2] stp instance 2 cost 20000
[*DeviceD-100GE1/0/2] commit
[~DeviceD-100GE1/0/2] quit
```

4. 使能MSTP，实现破除环路

   说明

   设备上MSTP功能默认使能。

   - 设备全局使能MSTP

     # 在核心层设备DeviceA上启动MSTP。

     ```
     [~DeviceA] stp enable
     [*DeviceA] commit
     ```

     # 在核心层设备DeviceB上启动MSTP。

     ```
     [~DeviceB] stp enable
     [*DeviceB] commit
     ```

     # 在接入层设备DeviceC上启动MSTP。

     ```
     [~DeviceC] stp enable
     [*DeviceC] commit
     ```

     # 在接入层设备DeviceD上启动MSTP。

     ```
     [~DeviceD] stp enable
     [*DeviceD] commit
     ```

   - 将与Host相连的端口配置为边缘端口

     # 配置接入层设备DeviceC端口100GE1/0/3为边缘端口。

     ```
     [~DeviceC] interface 100ge 1/0/3
     [~DeviceC-100GE1/0/3] description TO-HOSTA
     [*DeviceC-100GE1/0/3] stp edged-port enable
     [*DeviceC-100GE1/0/3] commit
     [~DeviceC-100GE1/0/3] quit
     ```

     # 配置接入层设备DeviceD端口100GE1/0/3为边缘端口。

     ```
     [~DeviceD] interface 100ge 1/0/3
     [~DeviceD-100GE1/0/3] description TO-HOSTB
     [*DeviceD-100GE1/0/3] stp edged-port enable
     [*DeviceD-100GE1/0/3] commit
     [~DeviceD-100GE1/0/3] quit
     ```

步骤2 配置保护功能，如在各实例的根桥设备的指定端口配置根保护功能（以DeviceA、DeviceB、DeviceC和DeviceD为例）

在核心层设备DeviceA端口100GE1/0/1上启动根保护。

```
[~DeviceA] interface 100ge 1/0/1
[~DeviceA-100GE1/0/1] description TO-CE6800-DEVICEC
[*DeviceA-100GE1/0/1] stp root-protection
[*DeviceA-100GE1/0/1] commit
[~DeviceA-100GE1/0/1] quit
```

在核心层设备DeviceB端口100GE1/0/3上启动根保护。

```
[~DeviceB] interface 100ge 1/0/3
[~DeviceB-100GE1/0/3] description TO-CE6800-DEVICED
[*DeviceB-100GE1/0/3] stp root-protection
[*DeviceB-100GE1/0/3] commit
[~DeviceB-100GE1/0/3] quit
```

步骤3 配置处于环网中的设备的二层转发功能（以DeviceA、DeviceB、DeviceC和DeviceD为例）

- 在交换设备DeviceA、DeviceB、DeviceC上创建VLAN2～3

  # 在核心层设备DeviceA上创建VLAN2～3。

  ```
  [~DeviceA] vlan batch 2 to 3
  ```

  # 在核心层设备DeviceB上创建VLAN2～3。

  ```
  [~DeviceB] vlan batch 2 to 3
  ```

  # 在接入层设备DeviceC上创建VLAN2。

  ```
  [~DeviceC] vlan batch 2
  ```

  # 在接入层设备DeviceD上创建VLAN3。

  ```
  [~DeviceD] vlan batch 3
  ```

- 将交换设备上接入环路中的端口加入VLAN

  # 将核心层设备DeviceA端口100GE1/0/1加入VLAN。

  ```
  [~DeviceA] interface 100ge 1/0/1
  [~DeviceA-100GE1/0/1] port link-type trunk
  [*DeviceA-100GE1/0/1] port trunk allow-pass vlan 2
  [*DeviceA-100GE1/0/1] undo port trunk allow-pass vlan 1
  [*DeviceA-100GE1/0/1] commit
  [~DeviceA-100GE1/0/1] quit
  ```

  # 将核心层设备DeviceA端口100GE1/0/2加入VLAN。

  ```
  [~DeviceA] interface 100ge 1/0/2
  [~DeviceA-100GE1/0/2] description TO-CE6800-DEVICED
  [*DeviceA-100GE1/0/2] port link-type trunk
  [*DeviceA-100GE1/0/2] port trunk allow-pass vlan 3
  [*DeviceA-100GE1/0/2] undo port trunk allow-pass vlan 1
  [*DeviceA-100GE1/0/2] commit
  [~DeviceA-100GE1/0/2] quit
  ```

  # 将核心层设备DeviceA端口100GE1/0/5加入VLAN。

  ```
  [~DeviceA] interface 100ge 1/0/5
  [~DeviceA-100GE1/0/5] description TO-CE16800-DEVICEB
  [*DeviceA-100GE1/0/5] port link-type trunk
  [*DeviceA-100GE1/0/5] port trunk allow-pass vlan 2 to 3
  [*DeviceA-100GE1/0/5] undo port trunk allow-pass vlan 1
  [*DeviceA-100GE1/0/5] commit
  [~DeviceA-100GE1/0/5] quit
  ```

  # 将核心层设备DeviceB端口100GE1/0/3加入VLAN。

  ```
  [~DeviceB] interface 100ge 1/0/3
  [~DeviceB-100GE1/0/3] port link-type trunk
  [*DeviceB-100GE1/0/3] port trunk allow-pass vlan 3
  [*DeviceB-100GE1/0/3] undo port trunk allow-pass vlan 1
  [*DeviceB-100GE1/0/3] commit
  [~DeviceB-100GE1/0/3] quit
  ```

  # 将核心层设备DeviceB端口100GE1/0/4加入VLAN。

  ```
  [~DeviceB] interface 100ge 1/0/4
  [~DeviceB-100GE1/0/4] description TO-CE6800-DEVICEC
  [*DeviceB-100GE1/0/4] port link-type trunk
  [*DeviceB-100GE1/0/4] port trunk allow-pass vlan 2
  [*DeviceB-100GE1/0/4] undo port trunk allow-pass vlan 1
  [*DeviceB-100GE1/0/4] commit
  [~DeviceB-100GE1/0/4] quit
  ```

  # 将核心层设备DeviceB端口100GE1/0/5加入VLAN。

  ```
  [~DeviceB] interface 100ge 1/0/5
  [~DeviceB-100GE1/0/5] description TO-CE16800-DEVICEB
  [*DeviceB-100GE1/0/5] port link-type trunk
  [*DeviceB-100GE1/0/5] port trunk allow-pass vlan 2 to 3
  [*DeviceB-100GE1/0/5] undo port trunk allow-pass vlan 1
  [*DeviceB-100GE1/0/5] commit
  [~DeviceB-100GE1/0/5] quit
  ```

  # 将接入层设备DeviceC端口100GE1/0/1加入VLAN。

```
[~DeviceC] interface 100ge 1/0/1
[~DeviceC-100GE1/0/1] description TO-CE16800-DEVICEA
[*DeviceC-100GE1/0/1] port trunk allow-pass vlan 2
[*DeviceC-100GE1/0/1] undo port trunk allow-pass vlan 1
[*DeviceC-100GE1/0/1] commit
[~DeviceC-100GE1/0/1] quit
```

将接入层设备DeviceC端口100GE1/0/2加入VLAN。

```
[~DeviceC] interface 100ge 1/0/2
[~DeviceC-100GE1/0/2] port link-type trunk
[*DeviceC-100GE1/0/2] port trunk allow-pass vlan 2
[*DeviceC-100GE1/0/2] undo port trunk allow-pass vlan 1
[*DeviceC-100GE1/0/2] commit
[~DeviceC-100GE1/0/2] quit
```

将接入层设备DeviceC端口100GE1/0/3加入VLAN。

```
[~DeviceC] interface 100ge 1/0/3
[~DeviceC-100GE1/0/3] port link-type access
[*DeviceC-100GE1/0/3] port default vlan 2
[*DeviceC-100GE1/0/3] commit
[~DeviceC-100GE1/0/3] quit
```

将接入层设备DeviceD端口100GE1/0/1加入VLAN。

```
[~DeviceD] interface 100ge 1/0/1
[~DeviceD-100GE1/0/1] description TO-CE16800-DEVICEB
[*DeviceD-100GE1/0/1] port link-type trunk
[*DeviceD-100GE1/0/1] port trunk allow-pass vlan 3
[*DeviceD-100GE1/0/1] undo port trunk allow-pass vlan 1
[*DeviceD-100GE1/0/1] commit
[~DeviceD-100GE1/0/1] quit
```

将接入层设备DeviceD端口100GE1/0/2加入VLAN。

```
[~DeviceD] interface 100ge 1/0/2
[~DeviceD-100GE1/0/2] port link-type trunk
[*DeviceD-100GE1/0/2] port trunk allow-pass vlan 3
[*DeviceD-100GE1/0/2] commit
[~DeviceD-100GE1/0/2] quit
```

将接入层设备DeviceD端口100GE1/0/3加入VLAN。

```
[~DeviceD] interface 100ge 1/0/3
[~DeviceD-100GE1/0/3] port link-type access
[*DeviceD-100GE1/0/3] port default vlan 3
[*DeviceD-100GE1/0/3] commit
[~DeviceD-100GE1/0/3] quit
```

步骤4 配置设备间的网络互连

配置设备各接口的IP地址，以核心层设备DeviceA为例。其他设备的配置与之类似，详见配置脚本。

```
[~DeviceA] vlan batch 6
[*DeviceA] interface 100ge 1/0/6
[*DeviceA-100GE1/0/6] description TO-DEVICEG
[*DeviceA-100GE1/0/6] port link-type trunk
[*DeviceA-100GE1/0/6] port trunk allow-pass vlan 6
[*DeviceA-100GE1/0/6] undo port trunk allow-pass vlan 1
[*DeviceA-100GE1/0/6] quit
[*DeviceA] interface vlanif 2
[*DeviceA-Vlanif2] ip address 10.1.2.102 24
[*DeviceA-Vlanif2] quit
[*DeviceA] interface vlanif 3
[*DeviceA-Vlanif3] ip address 10.1.3.102 24
[*DeviceA-Vlanif3] quit
[*DeviceA] interface vlanif 4
[*DeviceA-Vlanif4] ip address 10.1.4.102 24
[*DeviceA-Vlanif4] quit
[*DeviceA] interface vlanif 5
[*DeviceA-Vlanif5] ip address 10.1.5.102 24
[*DeviceA-Vlanif5] quit
```

```
[*DeviceA] interface vlanif 6
[*DeviceA-Vlanif6] ip address 10.1.6.102 24
[*DeviceA-Vlanif6] quit
[*DeviceA] commit
```

配置核心层设备和出口路由器之间采用OSPF协议进行互连。以核心层设备DeviceA为例，其他设备的配置与之类似，详见配置脚本。

```
[~DeviceA] ospf 1
[*DeviceA-ospf-1] area 0
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.2.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.3.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.4.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.5.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.6.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] quit
[*DeviceA-ospf-1] quit
[*DeviceA] commit
```

步骤5 配置VRRP备份组

在核心层设备DeviceA和DeviceB上创建VRRP备份组1，配置DeviceA的优先级为120，抢占延时为20秒，作为Master设备；DeviceB的优先级为缺省值，作为Backup设备。

- DeviceA
  ```
  [~DeviceA] interface vlanif 2
  [~DeviceA-Vlanif2] vrrp vrid 1 virtual-ip 10.1.2.100
  [*DeviceA-Vlanif2] vrrp vrid 1 priority 120
  [*DeviceA-Vlanif2] vrrp vrid 1 preempt timer delay 20
  [*DeviceA-Vlanif2] commit
  [~DeviceA-Vlanif2] quit
  ```
- DeviceB
  ```
  [~DeviceB] interface vlanif 2
  [~DeviceB-Vlanif2] vrrp vrid 1 virtual-ip 10.1.2.100
  [*DeviceB-Vlanif2] commit
  [~DeviceB-Vlanif2] quit
  ```

在核心层设备DeviceA和DeviceB上创建VRRP备份组2，配置DeviceB的优先级为120，抢占延时为20秒，作为Master设备；DeviceA的优先级为缺省值，作为Backup设备。

- DeviceB
  ```
  [~DeviceB] interface vlanif 3
  [~DeviceB-Vlanif3] vrrp vrid 2 virtual-ip 10.1.3.100
  [*DeviceB-Vlanif3] vrrp vrid 2 priority 120
  [*DeviceB-Vlanif3] vrrp vrid 2 preempt timer delay 20
  [*DeviceB-Vlanif3] commit
  [~DeviceB-Vlanif3] quit
  ```
- DeviceA
  ```
  [~DeviceA] interface vlanif 3
  [~DeviceA-Vlanif3] vrrp vrid 2 virtual-ip 10.1.3.100
  [*DeviceA-Vlanif3] commit
  [~DeviceA-Vlanif3] quit
  ```

在核心层设备DeviceA和DeviceB上创建VRRP备份组3，配置DeviceA的优先级为120，抢占延时为20秒，作为Master设备；DeviceB的优先级为缺省值，作为Backup设备。

- DeviceA
  ```
  [~DeviceA] interface vlanif 4
  [~DeviceA-Vlanif4] vrrp vrid 3 virtual-ip 10.1.4.100
  [*DeviceA-Vlanif4] vrrp vrid 3 priority 120
  [*DeviceA-Vlanif4] vrrp vrid 3 preempt timer delay 20
  ```

```
[*DeviceA-Vlanif4] commit
[~DeviceA-Vlanif4] quit
```

- DeviceB

```
[~DeviceB] interface vlanif 4
[~DeviceB-Vlanif4] vrrp vrid 3 virtual-ip 10.1.4.100
[*DeviceB-Vlanif4] commit
[~DeviceB-Vlanif4] quit
```

在核心层设备DeviceA和DeviceB上创建VRRP备份组4，配置DeviceB的优先级为120，抢占延时为20秒，作为Master设备；DeviceA的优先级为缺省值，作为Backup设备。

- DeviceB

```
[~DeviceB] interface vlanif 5
[~DeviceB-Vlanif5] vrrp vrid 4 virtual-ip 10.1.5.100
[*DeviceB-Vlanif5] vrrp vrid 4 priority 120
[*DeviceB-Vlanif5] vrrp vrid 4 preempt timer delay 20
[*DeviceB-Vlanif5] commit
[~DeviceB-Vlanif5] quit
```

- DeviceA

```
[~DeviceA] interface vlanif 5
[~DeviceA-Vlanif5] vrrp vrid 4 virtual-ip 10.1.3.100
[*DeviceA-Vlanif5] commit
[~DeviceA-Vlanif5] quit
```

配置主机HostA的缺省网关为备份组1的虚拟IP地址10.1.2.100，配置主机HostB的缺省网关为备份组2的虚拟IP地址10.1.3.100，配置主机HostC的缺省网关为备份组3的虚拟IP地址10.1.4.100，配置主机HostD的缺省网关为备份组4的虚拟IP地址10.1.5.100。

----结束

检查配置结果

1. 完成上述配置后，在核心层设备DeviceA上执行**display vrrp**命令，可以看到DeviceA在备份组1中作为Master设备，在备份组2中作为Backup设备。

```
<DeviceA> display vrrp verbose
 Vlanif2 | Virtual Router 1
   State : Master
   Virtual IP : 10.1.2.100
   Master IP : 10.1.2.102
   PriorityRun : 120
   PriorityConfig : 120
   MasterPriority : 120
   Preempt : YES   Delay Time : 20 s   Remain : --
   TimerRun : 1 s
   TimerConfig : 1 s
   Auth Type : NONE
   Virtual MAC : 0000-5e00-0101
   Check TTL : YES
   Config Type : Normal
   Create Time : 2022-05-11 11:39:18
   Last Change Time : 2022-05-26 11:38:58

 Vlanif3 | Virtual Router 2
   State : Backup
   Virtual IP : 10.1.3.100
   Master IP : 10.1.3.103
   PriorityRun : 100
   PriorityConfig : 100
   MasterPriority : 120
   Preempt : YES   Delay Time : 0 s   Remain : --
   TimerRun : 1 s
   TimerConfig : 1 s
   Auth type : NONE
```

```
  Virtual MAC : 0000-5e00-0102
  Check TTL : YES
  Config Type : Normal
  Create Time : 2022-05-11 11:40:18
  Last Change Time : 2022-05-26 11:48:58
```

2. 在核心层设备DeviceB上执行**display vrrp**命令，可以看到DeviceB在备份组1中作为Backup设备，在备份组2中作为Master设备。

```
<DeviceB> display vrrp verbose
 Vlanif2 | Virtual Router 1
   State : Backup
   Virtual IP : 10.1.2.100
   Master IP : 10.1.2.102
   PriorityRun : 100
   PriorityConfig : 100
   MasterPriority : 120
   Preempt : YES   Delay Time : 0 s   Remain : --
   TimerRun : 1 s
   TimerConfig : 1 s
   Auth Type : NONE
   Virtual MAC : 0000-5e00-0101
   Check TTL : YES
   Config Type : Normal
   Create Time : 2022-05-11 11:39:18
   Last Change Time : 2022-05-26 11:38:58

 Vlanif3 | Virtual Router 2
   State : Master
   Virtual IP : 10.1.3.100
   Master IP : 10.1.3.103
   PriorityRun : 120
   PriorityConfig : 120
   MasterPriority : 120
   Preempt : YES   Delay Time : 20 s   Remain : --
   TimerRun : 1 s
   TimerConfig : 1 s
   Auth type : NONE
   Virtual MAC : 0000-5e00-0102
   Check TTL : YES
   Config Type : Normal
   Create Time : 2022-05-11 11:40:18
   Last Change Time : 2022-05-26 11:48:58
```

配置脚本

- 核心层设备DeviceA的配置脚本

```
#
sysname DeviceA
#
vlan batch 2 to 6
#
stp instance 1 root primary
stp instance 2 root secondary
stp instance 3 root primary
stp instance 4 root secondary
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
 instance 3 vlan 4
 instance 4 vlan 5
#
interface Vlanif2
 ip address 10.1.2.102 255.255.255.0
 vrrp vrid 1 virtual-ip 10.1.2.100
 vrrp vrid 1 priority 120
```

```
 vrrp vrid 1 preempt timer delay 20
#
interface Vlanif3
 ip address 10.1.3.102 255.255.255.0
 vrrp vrid 2 virtual-ip 10.1.3.100
#
interface Vlanif4
 ip address 10.1.4.102 255.255.255.0
 vrrp vrid 3 virtual-ip 10.1.4.100
 vrrp vrid 3 priority 120
 vrrp vrid 3 preempt timer delay 20
#
interface Vlanif5
 ip address 10.1.5.102 255.255.255.0
 vrrp vrid 4 virtual-ip 10.1.5.100
#
interface Vlanif6
 ip address 10.1.6.102 255.255.255.0
#
interface 100GE1/0/1
 description TO-CE6800-DEVICEC
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2
 stp root-protection
#
interface 100GE1/0/2
 description TO-CE6800-DEVICED
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3
#
interface 100GE1/0/3
 description TO-CE6800-DEVICEE
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 4
 stp root-protection
#
interface 100GE1/0/4
 description TO-CE6800-DEVICEF
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 5
#
interface 100GE1/0/5
 description TO-CE16800-DEVICEB
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2 to 5
#
interface 100GE1/0/6
 description TO-DEVICEG
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 6
#
ospf 1
 area 0.0.0.0
  network 10.1.2.0 0.0.0.255
  network 10.1.3.0 0.0.0.255
  network 10.1.4.0 0.0.0.255
  network 10.1.5.0 0.0.0.255
  network 10.1.6.0 0.0.0.255
#
return
```

- 核心层设备DeviceB的配置脚本

```
#
sysname DeviceB
```



```
#
vlan batch 2 to 5 7
#
stp instance 1 root secondary
stp instance 2 root primary
stp instance 3 root secondary
stp instance 4 root primary
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
 instance 3 vlan 4
 instance 4 vlan 5
#
interface Vlanif2
 ip address 10.1.2.103 255.255.255.0
 vrrp vrid 1 virtual-ip 10.1.2.100
#
interface Vlanif3
 ip address 10.1.3.103 255.255.255.0
 vrrp vrid 2 virtual-ip 10.1.3.100
 vrrp vrid 2 priority 120
 vrrp vrid 2 preempt timer delay 20
#
interface Vlanif4
 ip address 10.1.4.103 255.255.255.0
 vrrp vrid 3 virtual-ip 10.1.4.100
#
interface Vlanif5
 ip address 10.1.5.103 255.255.255.0
 vrrp vrid 4 virtual-ip 10.1.5.100
 vrrp vrid 4 priority 120
 vrrp vrid 4 preempt timer delay 20
#
interface Vlanif7
 ip address 10.1.7.103 255.255.255.0
#
interface 100GE1/0/1
 description TO-CE6800-DEVICEF
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 5
 stp root-protection
#
interface 100GE1/0/2
 description TO-CE6800-DEVICEE
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 4
#
interface 100GE1/0/3
 description TO-CE6800-DEVICED
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3
 stp root-protection
#
interface 100GE1/0/4
 description TO-CE6800-DEVICEC
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2
#
interface 100GE1/0/5
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
```

```
 port trunk allow-pass vlan 2 to 5
#
interface 100GE1/0/6
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 7
#
ospf 1
 area 0.0.0.0
  network 10.1.2.0 0.0.0.255
  network 10.1.3.0 0.0.0.255
  network 10.1.4.0 0.0.0.255
  network 10.1.5.0 0.0.0.255
  network 10.1.7.0 0.0.0.255
#
return
```

- 接入层设备DeviceC的配置脚本

```
#
sysname DeviceC
#
vlan batch 2
#
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
 instance 3 vlan 4
 instance 4 vlan 5
#
interface 100GE1/0/1
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2
#
interface 100GE1/0/2
 description TO-CE16800-DEVICEB
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 2
 stp instance 1 cost 20000
#
interface 100GE1/0/3
 description TO-HOSTA
 port default vlan 2
 stp edged-port enable
#
return
```

- 接入层设备DeviceD的配置脚本

```
#
sysname DeviceD
#
vlan batch 3
#
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
 instance 3 vlan 4
 instance 4 vlan 5
#
interface 100GE1/0/1
 description TO-CE16800-DEVICEB
```



```
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3
#
interface 100GE1/0/2
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3
 stp instance 2 cost 20000
#
interface 100GE1/0/3
 description TO-HOSTB
 port default vlan 3
 stp edged-port enable
#
return
```

- 接入层DeviceE的配置脚本

```
#
sysname DeviceE
#
vlan batch 4
#
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
 instance 3 vlan 4
 instance 4 vlan 5
#
interface 100GE1/0/1
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 4
#
interface 100GE1/0/2
 description TO-CE16800-DEVICEB
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 4
 stp instance 1 cost 20000
#
interface 100GE1/0/3
 description TO-HOSTC
 port default vlan 4
 stp edged-port enable
#
return
```

- 接入层DeviceF的配置脚本

```
#
sysname DeviceF
#
vlan batch 5
#
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
 instance 3 vlan 4
 instance 4 vlan 5
#
interface 100GE1/0/1
 description TO-CE16800-DEVICEB
```



```
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 5
#
interface 100GE1/0/2
 description TO-CE16800-DEVICEA
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 5
 stp instance 2 cost 20000
#
interface 100GE1/0/3
 description TO-HOSTD
 port default vlan 5
 stp edged-port enable
#
return
```

1.3 配置分布式网关部署方式的 IPv4 VXLAN 示例

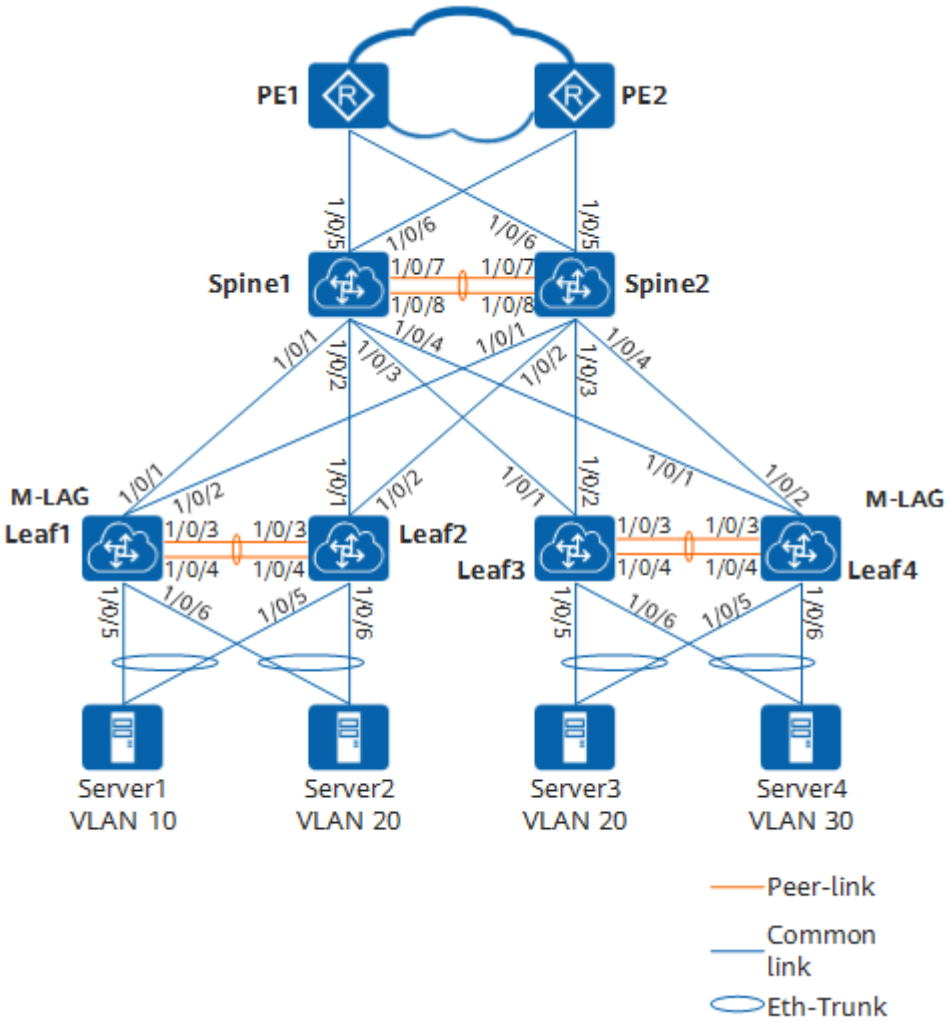
适用产品和版本

- CE16800（除X系列单板外）、CE8800、CE6800（除CE6820H、CE6820H-K、CE6820S外）系列产品V300R020C00或更高版本。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

如**图1-3**所示，某企业新建的数据中心网络采用分布式网关部署方式，其中Underlay基础网络为IPv4，Overlay网络为IPv4/IPv6。Leaf作为三层网关与服务器对接；Spine同时作为东西向流量的汇聚设备和网络出口网关。为了保证高可靠性，Spine、Leaf采用M-LAG部署方式。

图 1-3 分布式网关部署方式的 VXLAN 网络示意图



说明

上图中“1/0/1”为接口编号，接口速率为100GE，即“1/0/1”表示接口“100GE1/0/1”。其他接口类似。

表 1-3 接口地址表

设备名称	接口	IP地址	设备名称	接口	IP地址
Spine 1	100GE1/0/1	192.168.1.1/24	Spine 2	100GE1/0/1	192.168.5.1/24
	100GE1/0/2	192.168.2.1/24		100GE1/0/2	192.168.6.1/24
	100GE1/0/3	192.168.3.1/24		100GE1/0/3	192.168.7.1/24

设备名称	接口	IP地址	设备名称	接口	IP地址
	100GE1/0/4	192.168.4.1/24		100GE1/0/4	192.168.8.1/24
	100GE1/0/5	IPv4: 10.1.10.1/24 IPv6: fc00:10::1/64		100GE1/0/5	IPv4: 10.1.30.1/24 IPv6: fc00:30::1/64
	100GE1/0/6	IPv4: 10.1.20.1/24 IPv6: fc00:20::1/64		100GE1/0/6	IPv4: 10.1.40.1/24 IPv6: fc00:40::1/64
	Loopback0	4.4.4.4/32		Loopback0	5.5.5.5/32
	Loopback1	1.1.1.1/32		Loopback1	1.1.1.1/32
	Loopback2	10.10.10.10/32		Loopback2	11.11.11.11/32
Leaf1	100GE1/0/1	192.168.1.2/24	Leaf2	100GE1/0/1	192.168.2.2/24
	100GE1/0/2	192.168.5.2/24		100GE1/0/2	192.168.6.2/24
	Loopback0	6.6.6.6/32		Loopback0	7.7.7.7/32
	Loopback1	2.2.2.2/32		Loopback1	2.2.2.2/32
	Loopback2	12.12.12.12/32		Loopback2	13.13.13.13/32
Leaf3	100GE1/0/1	192.168.3.2/24	Leaf4	100GE1/0/1	192.168.4.2/24
	100GE1/0/2	192.168.7.2/24		100GE1/0/2	192.168.8.2/24
	Loopback0	8.8.8.8/32		Loopback0	9.9.9.9/32
	Loopback1	3.3.3.3/32		Loopback1	3.3.3.3/32
	Loopback2	14.14.14.14/32		Loopback2	15.15.15.15/32

配置思路

采用如下思路配置分布式网关部署方式的VXLAN网络：

1. 配置路由协议，保证Underlay网络三层互通。
2. 配置M-LAG，实现服务器双活接入。

3. 配置BGP EVPN建立VXLAN隧道。

操作步骤

步骤1 配置路由协议，实现Underlay网络三层互通。

配置Leaf1。其他设备的配置与Leaf1类似，这里不再赘述，具体配置请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf1
[*HUAWEI] commit
[~Leaf1] interface 100ge 1/0/1
[~Leaf1-100GE1/0/1] undo portswitch
[*Leaf1-100GE1/0/1] ip address 192.168.1.2 24
[*Leaf1-100GE1/0/1] ospf network-type p2p
[*Leaf1-100GE1/0/1] quit
[*Leaf1] interface 100ge 1/0/2
[*Leaf1-100GE1/0/2] undo portswitch
[*Leaf1-100GE1/0/2] ip address 192.168.5.2 24
[*Leaf1-100GE1/0/2] ospf network-type p2p
[*Leaf1-100GE1/0/2] quit
[*Leaf1] interface loopback 0
[*Leaf1-LoopBack0] ip address 6.6.6.6 32
[*Leaf1-LoopBack0] quit
[*Leaf1] interface loopback 1
[*Leaf1-LoopBack1] ip address 2.2.2.2 32
[*Leaf1-LoopBack1] quit
[*Leaf1] ospf
[*Leaf1-ospf-1] area 0
[*Leaf1-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[*Leaf1-ospf-1-area-0.0.0.0] network 192.168.5.0 0.0.0.255
[*Leaf1-ospf-1-area-0.0.0.0] network 6.6.6.6 0.0.0.0
[*Leaf1-ospf-1-area-0.0.0.0] network 2.2.2.2 0.0.0.0
[*Leaf1-ospf-1-area-0.0.0.0] quit
[*Leaf1-ospf-1] quit
[*Leaf1] commit
```

OSPF成功配置后，Leaf、Spine之间可通过OSPF协议发现对方的Loopback接口的地址，并能互相ping通。

步骤2 配置Leaf组成M-LAG系统。本示例中Leaf1、Leaf2组成M-LAG系统，Leaf3、Leaf4组成M-LAG系统，具体配置与此类似，不再赘述。

配置Leaf1。

```
[~Leaf1] stp mode rstp
[*Leaf1] stp v-stp enable
[*Leaf1] dfs-group 1
[*Leaf1-dfs-group-1] dual-active detection source ip 6.6.6.6 peer 7.7.7.7
[*Leaf1-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Leaf1-dfs-group-1] priority 150
[*Leaf1-dfs-group-1] quit
[*Leaf1] interface eth-trunk 1
[*Leaf1-Eth-Trunk1] trunkport 100ge 1/0/3
[*Leaf1-Eth-Trunk1] trunkport 100ge 1/0/4
[*Leaf1-Eth-Trunk1] mode lacp-static
[*Leaf1-Eth-Trunk1] peer-link 1
[*Leaf1-Eth-Trunk1] quit
[*Leaf1] interface eth-trunk 2
[*Leaf1-Eth-Trunk2] trunkport 100ge 1/0/5
[*Leaf1-Eth-Trunk2] mode lacp-static
[*Leaf1-Eth-Trunk2] dfs-group 1 m-lag 1
[*Leaf1-Eth-Trunk2] stp edged-port enable
[*Leaf1-Eth-Trunk2] quit
[*Leaf1] interface eth-trunk 3
[*Leaf1-Eth-Trunk3] trunkport 100ge 1/0/6
```

```
[*Leaf1-Eth-Trunk3] mode lacp-static
[*Leaf1-Eth-Trunk3] dfs-group 1 m-lag 2
[*Leaf1-Eth-Trunk3] stp edged-port enable
[*Leaf1-Eth-Trunk3] quit
[*Leaf1] commit
[~Leaf1] monitor-link group 1
[*Leaf1-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf1-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf1-mtlk-group1] port eth-trunk 2 downlink 1
[*Leaf1-mtlk-group1] port eth-trunk 3 downlink 2
[*Leaf1-mtlk-group1] quit
[*Leaf1] commit
```

#配置Leaf2。

```
[~Leaf2] stp mode rstp
[*Leaf2] stp v-stp enable
[*Leaf2] dfs-group 1
[*Leaf2-dfs-group-1] dual-active detection source ip 7.7.7.7 peer 6.6.6.6
[*Leaf2-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Leaf2-dfs-group-1] quit
[*Leaf2] interface eth-trunk 1
[*Leaf2-Eth-Trunk1] trunkport 100ge 1/0/3
[*Leaf2-Eth-Trunk1] trunkport 100ge 1/0/4
[*Leaf2-Eth-Trunk1] mode lacp-static
[*Leaf2-Eth-Trunk1] peer-link 1
[*Leaf2-Eth-Trunk1] quit
[*Leaf2] interface eth-trunk 2
[*Leaf2-Eth-Trunk2] mode lacp-static
[*Leaf2-Eth-Trunk2] trunkport 100ge 1/0/5
[*Leaf2-Eth-Trunk2] dfs-group 1 m-lag 1
[*Leaf2-Eth-Trunk2] stp edged-port enable
[*Leaf2-Eth-Trunk2] quit
[*Leaf2] interface eth-trunk 3
[*Leaf2-Eth-Trunk3] mode lacp-static
[*Leaf2-Eth-Trunk3] trunkport 100ge 1/0/6
[*Leaf2-Eth-Trunk3] dfs-group 1 m-lag 2
[*Leaf2-Eth-Trunk3] stp edged-port enable
[*Leaf2-Eth-Trunk3] quit
[*Leaf2] commit
[~Leaf2] monitor-link group 1
[*Leaf2-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf2-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf2-mtlk-group1] port eth-trunk 2 downlink 1
[*Leaf2-mtlk-group1] port eth-trunk 3 downlink 2
[*Leaf2-mtlk-group1] quit
[*Leaf2] commit
```

步骤3 配置Spine1、Spine2组成M-LAG系统。

配置Spine1。

```
[~Spine1] stp mode rstp
[*Spine1] stp v-stp enable
[*Spine1] dfs-group 1
[*Spine1-dfs-group-1] dual-active detection source ip 4.4.4.4 peer 5.5.5.5
[*Spine1-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Spine1-dfs-group-1] priority 150
[*Spine1-dfs-group-1] quit
[*Spine1] interface eth-trunk 1
[*Spine1-Eth-Trunk1] trunkport 100ge 1/0/7
[*Spine1-Eth-Trunk1] trunkport 100ge 1/0/8
[*Spine1-Eth-Trunk1] mode lacp-static
[*Spine1-Eth-Trunk1] peer-link 1
[*Spine1-Eth-Trunk1] quit
[*Spine1] commit
```

#配置Spine2。

```
[~Spine2] stp mode rstp
[*Spine2] stp v-stp enable
```



```
[*Spine2] dfs-group 1
[*Spine2-dfs-group-1] dual-active detection source ip 5.5.5.5 peer 4.4.4.4
[*Spine2-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Spine2-dfs-group-1] quit
[*Spine2] interface eth-trunk 1
[*Spine2-Eth-Trunk1] trunkport 100ge 1/0/7
[*Spine2-Eth-Trunk1] trunkport 100ge 1/0/8
[*Spine2-Eth-Trunk1] mode lacp-static
[*Spine2-Eth-Trunk1] peer-link 1
[*Spine2-Eth-Trunk1] quit
[*Spine2] commit
```

步骤4 配置BGP EVPN，建立VXLAN隧道。

1. 配置业务接入点。

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述。

   ```
   [~Leaf1] bridge-domain 10
   [*Leaf1-bd10] quit
   [*Leaf1] bridge-domain 20
   [*Leaf1-bd20] quit
   [*Leaf1] interface eth-trunk 2.10 mode l2
   [*Leaf1-Eth-Trunk2.1] encapsulation dot1q vid 10
   [*Leaf1-Eth-Trunk2.1] bridge-domain 10
   [*Leaf1-Eth-Trunk2.1] quit
   [*Leaf1] interface eth-trunk 3.20 mode l2
   [*Leaf1-Eth-Trunk3.1] encapsulation dot1q vid 20
   [*Leaf1-Eth-Trunk3.1] bridge-domain 20
   [*Leaf1-Eth-Trunk3.1] quit
   [*Leaf1] commit
   ```

2. 配置BGP EVPN对等体关系。Spine1、Spine2作为路由反射器。

   # 配置Spine1。Spine2的配置与Spine1类似，这里不再赘述，具体配置请参考配置脚本。

   ```
   [~Spine1] evpn-overlay enable
   [*Spine1] bgp 100
   [*Spine1-bgp] peer 6.6.6.6 as-number 100
   [*Spine1-bgp] peer 6.6.6.6 connect-interface LoopBack0
   [*Spine1-bgp] peer 7.7.7.7 as-number 100
   [*Spine1-bgp] peer 7.7.7.7 connect-interface LoopBack0
   [*Spine1-bgp] peer 8.8.8.8 as-number 100
   [*Spine1-bgp] peer 8.8.8.8 connect-interface LoopBack0
   [*Spine1-bgp] peer 9.9.9.9 as-number 100
   [*Spine1-bgp] peer 9.9.9.9 connect-interface LoopBack0
   [*Spine1-bgp] l2vpn-family evpn
   [*Spine1-bgp-af-evpn] peer 6.6.6.6 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Spine1-bgp-af-evpn] peer 6.6.6.6 advertise irb
   [*Spine1-bgp-af-evpn] peer 6.6.6.6 advertise irbv6
   [*Spine1-bgp-af-evpn] peer 6.6.6.6 reflect-client
   [*Spine1-bgp-af-evpn] peer 7.7.7.7 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Spine1-bgp-af-evpn] peer 7.7.7.7 advertise irb
   [*Spine1-bgp-af-evpn] peer 7.7.7.7 advertise irbv6
   [*Spine1-bgp-af-evpn] peer 7.7.7.7 reflect-client
   [*Spine1-bgp-af-evpn] peer 8.8.8.8 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Spine1-bgp-af-evpn] peer 8.8.8.8 advertise irb
   [*Spine1-bgp-af-evpn] peer 8.8.8.8 advertise irbv6
   [*Spine1-bgp-af-evpn] peer 8.8.8.8 reflect-client
   [*Spine1-bgp-af-evpn] peer 9.9.9.9 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Spine1-bgp-af-evpn] peer 9.9.9.9 advertise irb
   [*Spine1-bgp-af-evpn] peer 9.9.9.9 advertise irbv6
   [*Spine1-bgp-af-evpn] peer 9.9.9.9 reflect-client
   [*Spine1-bgp-af-evpn] undo policy vpn-target
   [*Spine1-bgp-af-evpn] quit
   [*Spine1-bgp] quit
   [*Spine1] commit
   ```

配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述，具体配置请参考配置脚本。

```
[~Leaf1] evpn-overlay enable
[*Leaf1] bgp 100
[*Leaf1-bgp] peer 4.4.4.4 as-number 100
[*Leaf1-bgp] peer 4.4.4.4 connect-interface LoopBack0
[*Leaf1-bgp] peer 5.5.5.5 as-number 100
[*Leaf1-bgp] peer 5.5.5.5 connect-interface LoopBack0
[*Leaf1-bgp] l2vpn-family evpn
[*Leaf1-bgp-af-evpn] peer 4.4.4.4 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf1-bgp-af-evpn] peer 4.4.4.4 advertise irb
[*Leaf1-bgp-af-evpn] peer 4.4.4.4 advertise irbv6
[*Leaf1-bgp-af-evpn] peer 5.5.5.5 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf1-bgp-af-evpn] peer 5.5.5.5 advertise irb
[*Leaf1-bgp-af-evpn] peer 5.5.5.5 advertise irbv6
[*Leaf1-bgp-af-evpn] quit
[*Leaf1-bgp] quit
[*Leaf1] commit
```

3. 配置VPN实例及EVPN实例。

   # 配置Spine1。Spine2的配置与Spine1类似，这里不再赘述，具体配置请参考配置脚本。

```
[~Spine1] ip vpn-instance vpn1
[*Spine1-vpn-instance-vpn1] vxlan vni 5000
[*Spine1-vpn-instance-vpn1] ipv4-family
[*Spine1-vpn-instance-vpn1-af-ipv4] route-distinguisher 4.4.4.4:1
[*Spine1-vpn-instance-vpn1-af-ipv4] vpn-target 0:1 evpn
[*Spine1-vpn-instance-vpn1-af-ipv4] quit
[*Spine1-vpn-instance-vpn1] ipv6-family
[*Spine1-vpn-instance-vpn1-af-ipv6] route-distinguisher 4.4.4.4:1
[*Spine1-vpn-instance-vpn1-af-ipv6] vpn-target 0:1 evpn
[*Spine1-vpn-instance-vpn1-af-ipv6] quit
[*Spine1-vpn-instance-vpn1] quit
[*Spine1] bgp 100
[*Spine1-bgp] ipv4-family vpn-instance vpn1
[*Spine1-bgp-vpn1] import-route static
[*Spine1-bgp-vpn1] advertise l2vpn evpn
[*Spine1-bgp-vpn1] quit
[*Spine1-bgp] ipv6-family vpn-instance vpn1
[*Spine1-bgp-6-vpn1] import-route static
[*Spine1-bgp-6-vpn1] advertise l2vpn evpn
[*Spine1-bgp-6-vpn1] quit
[*Spine1-bgp] quit
[*Spine1] interface nve 1   //配置NVE
[*Spine1-Nve1] source 1.1.1.1   //Spine1和Spine2作为M-LAG双活系统，这两台设备上配置的NVE接口的IP地址和MAC地址需要相同
[*Spine1-Nve1] mac-address 00e0-fc00-0101
[*Spine1-Nve1] quit
[*Spine1] commit
```

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述，具体配置请参考配置脚本。

```
[~Leaf1] ip vpn-instance vpn1
[*Leaf1-vpn-instance-vpn1] vxlan vni 5000
[*Leaf1-vpn-instance-vpn1] ipv4-family
[*Leaf1-vpn-instance-vpn1-af-ipv4] route-distinguisher 6.6.6.6:1
[*Leaf1-vpn-instance-vpn1-af-ipv4] vpn-target 0:1 evpn
[*Leaf1-vpn-instance-vpn1-af-ipv4] quit
[*Leaf1-vpn-instance-vpn1] ipv6-family
[*Leaf1-vpn-instance-vpn1-af-ipv6] route-distinguisher 6.6.6.6:1
[*Leaf1-vpn-instance-vpn1-af-ipv6] vpn-target 0:1 evpn
[*Leaf1-vpn-instance-vpn1-af-ipv6] quit
[*Leaf1-vpn-instance-vpn1] quit
[*Leaf1] bridge-domain 10
[*Leaf1-bd10] vxlan vni 10
[*Leaf1-bd10] evpn
```

```
[*Leaf1-bd10-evpn] route-distinguisher 6.6.6.6:10
[*Leaf1-bd10-evpn] vpn-target 0:10
[*Leaf1-bd10-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd10-evpn] quit
[*Leaf1-bd10] quit
[*Leaf1] bridge-domain 20
[*Leaf1-bd20] vxlan vni 20
[*Leaf1-bd20] evpn
[*Leaf1-bd20-evpn] route-distinguisher 6.6.6.6:20
[*Leaf1-bd20-evpn] vpn-target 0:20
[*Leaf1-bd20-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd20-evpn] quit
[*Leaf1-bd20] quit
[*Leaf1] bgp 100
[*Leaf1-bgp] ipv4-family vpn-instance vpn1
[*Leaf1-bgp-vpn1] import-route direct
[*Leaf1-bgp-vpn1] advertise l2vpn evpn
[*Leaf1-bgp-vpn1] quit
[*Leaf1-bgp] ipv6-family vpn-instance vpn1
[*Leaf1-bgp-6-vpn1] import-route direct
[*Leaf1-bgp-6-vpn1] advertise l2vpn evpn
[*Leaf1-bgp-6-vpn1] quit
[*Leaf1-bgp] quit
[*Leaf1] interface nve 1      //配置NVE
[*Leaf1-Nve1] source 2.2.2.2  //Leaf1和Leaf2作为M-LAG双活系统，这两台设备上配置的NVE接口的IP地址和MAC地址需要相同
[*Leaf1-Nve1] mac-address 00e0-fc00-0102
[*Leaf1-Nve1] vni 10 head-end peer-list protocol bgp
[*Leaf1-Nve1] vni 20 head-end peer-list protocol bgp
[*Leaf1-Nve1] quit
[*Leaf1] commit
```

4. 在Leaf1、Leaf2、Leaf3、Leaf4上配置三层网关。

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述，具体配置请参考配置脚本。

```
[~Leaf1] interface vbdif 10
[*Leaf1-Vbdif10] ip binding vpn-instance vpn1
[*Leaf1-Vbdif10] ip address 10.1.1.1 24  //Leaf1和Leaf2作为M-LAG双活系统，这两台设备上配置的VBDIF接口的IP地址和MAC地址需要相同
[*Leaf1-Vbdif10] ipv6 enable
[*Leaf1-Vbdif10] ipv6 address fc00:1::1 64
[*Leaf1-Vbdif10] mac-address 00e0-fc00-0105
[*Leaf1-Vbdif10] vxlan anycast-gateway enable
[*Leaf1-Vbdif10] arp collect host enable
[*Leaf1-Vbdif10] arp broadcast-detect enable
[*Leaf1-Vbdif10] ipv6 nd collect host enable
[*Leaf1-Vbdif10] ipv6 nd na glean
[*Leaf1-Vbdif10] quit
[*Leaf1] interface vbdif 20
[*Leaf1-Vbdif20] ip binding vpn-instance vpn1
[*Leaf1-Vbdif20] ip address 10.1.2.1 24
[*Leaf1-Vbdif20] ipv6 enable
[*Leaf1-Vbdif20] ipv6 address fc00:2::1 64
[*Leaf1-Vbdif20] mac-address 00e0-fc00-0106
[*Leaf1-Vbdif20] vxlan anycast-gateway enable
[*Leaf1-Vbdif20] arp collect host enable
[*Leaf1-Vbdif20] arp broadcast-detect enable
[*Leaf1-Vbdif20] ipv6 nd collect host enable
[*Leaf1-Vbdif20] ipv6 nd na glean
[*Leaf1-Vbdif20] quit
[*Leaf1] commit
```

步骤5 在M-LAG设备中配置静态Bypass VXLAN隧道。

在M-LAG双归接入VXLAN的场景中，当下行一条链路发生故障时，业务流量需绕行M-LAG设备之间的Peer-link链路。因此，在该场景下M-LAG设备之间必须配置静态Bypass VXLAN隧道，将绕行的业务流量引导至Peer-link链路上。

下面以Leaf1和Leaf2配置为例，Spine1、Spine2、Leaf3、Leaf4的配置与之类似，这里不再赘述，具体配置请参考配置脚本。

配置Leaf1。

```
[~Leaf1] vlan 100  //本VLAN不能划分给其他业务使用，本例中以100举例
[*Leaf1-vlan100] m-lag peer-link reserved  //仅允许peer-link加入到该VLAN
[*Leaf1-vlan100] quit
[*Leaf1] interface vlanif 100
[*Leaf1-Vlanif100] reserved for vxlan bypass  //指定peer-link接口上VLANIF的IPv4地址只给Bypass VXLAN隧道使用
[*Leaf1-Vlanif100] ip address 10.2.2.1 30  //配置静态Bypass VXLAN隧道的源端IPv4地址
[*Leaf1-Vlanif100] quit
[*Leaf1] ip route-static 13.13.13.13 32 10.2.2.2 preference 1  //配置静态路由，打通Bypass VXLAN隧道
[*Leaf1] interface nve 1
[*Leaf1-Nve1] pip-source 12.12.12.12 peer 13.13.13.13 bypass  //创建静态Bypass VXLAN隧道，指定源端地址和对端地址
[*Leaf1-Nve1] quit
[*Leaf1] commit
```

配置Leaf2。

```
[~Leaf2] vlan 100
[*Leaf2-vlan100] m-lag peer-link reserved
[*Leaf2-vlan100] quit
[*Leaf2] interface vlanif 100
[*Leaf2-Vlanif100] reserved for vxlan bypass
[*Leaf2-Vlanif100] ip address 10.2.2.2 30
[*Leaf2-Vlanif100] quit
[*Leaf2] ip route-static 12.12.12.12 32 10.2.2.1 preference 1
[*Leaf2] interface nve 1
[*Leaf2-Nve1] pip-source 13.13.13.13 peer 12.12.12.12 bypass
[*Leaf2-Nve1] quit
[*Leaf2] commit
```

步骤6 在Spine1、Spine2上配置静态路由，实现南、北向流量互通。

配置Spine1。Spine2的配置与Spine1类似，这里不再赘述，具体配置请参考配置脚本。

```
[~Spine1] interface 100ge 1/0/5
[~Spine1-100GE1/0/5] undo portswitch
[*Spine1-100GE1/0/5] ip address 10.1.10.1 24
[*Spine1-100GE1/0/5] ipv6 enable
[*Spine1-100GE1/0/5] ipv6 address fc00:10::1 64
[*Spine1-100GE1/0/5] quit
[~Spine1] interface 100ge 1/0/6
[~Spine1-100GE1/0/6] undo portswitch
[~Spine1-100GE1/0/6] ip address 10.1.20.1 24
[~Spine1-100GE1/0/6] ipv6 enable
[*Spine1-100GE1/0/6] ipv6 address fc00:20::1 64
[*Spine1-100GE1/0/6] quit
[*Spine1] ip route-static 0.0.0.0 0.0.0.0 10.1.10.2  //至公网PE的IPv4静态路由
[*Spine1] ip route-static 0.0.0.0 0.0.0.0 10.1.20.2
[*Spine1] ip route-static 10.1.1.0 24 vpn-instance vpn1  //至服务器网段的IPv4静态路由，下一跳为VPN实例
[*Spine1] ip route-static 10.1.2.0 24 vpn-instance vpn1
[*Spine1] ip route-static 10.1.3.0 24 vpn-instance vpn1
[*Spine1] ip route-static vpn-instance vpn1 0.0.0.0 0.0.0.0 public  //VPN实例的IPv4静态路由，下一跳为公网实例
[*Spine1] ipv6 route-static :: 0 fc00:10::2  //至公网PE的IPv6静态路由
[*Spine1] ipv6 route-static :: 0 fc00:20::2
[*Spine1] ipv6 route-static fc00:1:: 64 vpn-instance vpn1  //至服务器网段的IPv6静态路由，下一跳为VPN实例
[*Spine1] ipv6 route-static fc00:2:: 64 vpn-instance vpn1
[*Spine1] ipv6 route-static fc00:3:: 64 vpn-instance vpn1
[*Spine1] ipv6 route-static vpn-instance vpn1 :: 0 public  //VPN实例的IPv6静态路由，下一跳为公网实例
[*Spine1] commit
```

----结束

检查配置结果

上述配置成功后，执行**display vxlan tunnel**命令可查看到VXLAN隧道的信息。以Spine1显示为例。

```
[~Spine1] display vxlan tunnel
Number of vxlan tunnel : 2
Tunnel ID   Source            Destination       State  Type     Uptime
--------------------------------------------------------------------------------
4026531841  1.1.1.1           2.2.2.2           up     dynamic  0035h21m
4026531842  1.1.1.1           3.3.3.3           up     dynamic  0036h21m
```

配置完成后，服务器之间可以相互通信。

配置脚本

- Spine1的配置脚本

```
#
sysname Spine1
#
dfs-group 1
 priority 150
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 4.4.4.4 peer 5.5.5.5
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 4.4.4.4:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 4.4.4.4:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
interface Vlanif100
 ip address 10.1.1.1 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.2.1 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.3.1 255.255.255.0
 ospf network-type p2p
#
```

```
interface 100GE1/0/4
 undo portswitch
 ip address 192.168.4.1 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/5
 undo portswitch
 ipv6 enable
 ip address 10.1.10.1 255.255.255.0
 ipv6 address FC00:10::1/64
#
interface 100GE1/0/6
 undo portswitch
 ipv6 enable
 ip address 10.1.20.1 255.255.255.0
 ipv6 address FC00:20::1/64
#
interface 100GE1/0/7
 eth-trunk 1
#
interface 100GE1/0/8
 eth-trunk 1
#
interface LoopBack0
 ip address 4.4.4.4 255.255.255.255
#
interface LoopBack1
 ip address 1.1.1.1 255.255.255.255
#
interface LoopBack2
 ip address 10.10.10.10 255.255.255.255
#
interface Nve1
 source 1.1.1.1
 pip-source 10.10.10.10 peer 11.11.11.11 bypass
 mac-address 00e0-fc00-0101
#
bgp 100
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack0
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 peer 9.9.9.9 as-number 100
 peer 9.9.9.9 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 6.6.6.6 enable
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
  peer 9.9.9.9 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route static
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 advertise irb
  peer 6.6.6.6 advertise irbv6
  peer 6.6.6.6 reflect-client
  peer 7.7.7.7 enable
  peer 7.7.7.7 advertise irb
```

```
 peer 7.7.7.7 advertise irbv6
 peer 7.7.7.7 reflect-client
 peer 8.8.8.8 enable
 peer 8.8.8.8 advertise irb
 peer 8.8.8.8 advertise irbv6
 peer 8.8.8.8 reflect-client
 peer 9.9.9.9 enable
 peer 9.9.9.9 advertise irb
 peer 9.9.9.9 advertise irbv6
 peer 9.9.9.9 reflect-client
#
ospf 1
 area 0.0.0.0
  network 1.1.1.1 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.2.0 0.0.0.255
  network 192.168.3.0 0.0.0.255
  network 192.168.4.0 0.0.0.255
#
ip route-static 0.0.0.0 0.0.0.0 10.1.10.2
ip route-static 0.0.0.0 0.0.0.0 10.1.20.2
ip route-static 10.1.1.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.1.2.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.1.3.0 255.255.255.0 vpn-instance vpn1
ip route-static 11.11.11.11 32 10.1.1.2 preference 1
ip route-static vpn-instance vpn1 0.0.0.0 0.0.0.0 public
#
ipv6 route-static :: 0 FC00:10::2
ipv6 route-static :: 0 FC00:20::2
ipv6 route-static FC00:1:: 64 vpn-instance vpn1
ipv6 route-static FC00:2:: 64 vpn-instance vpn1
ipv6 route-static FC00:3:: 64 vpn-instance vpn1
ipv6 route-static vpn-instance vpn1 :: 0 public
#
return
```

- Spine2的配置脚本

```
#
sysname Spine2
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 5.5.5.5 peer 4.4.4.4
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 5.5.5.5:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 5.5.5.5:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
interface Vlanif100
 ip address 10.1.1.2 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
```

```
 mode lacp-static
 peer-link 1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.5.1 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.6.1 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.7.1 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/4
 undo portswitch
 ip address 192.168.8.1 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/5
 undo portswitch
 ipv6 enable
 ip address 10.1.30.1 255.255.255.0
 ipv6 address FC00:30::1/64
#
interface 100GE1/0/6
 undo portswitch
 ipv6 enable
 ip address 10.1.40.1 255.255.255.0
 ipv6 address FC00:40::1/64
#
interface 100GE1/0/7
 eth-trunk 1
#
interface 100GE1/0/8
 eth-trunk 1
#
interface LoopBack0
 ip address 5.5.5.5 255.255.255.255
#
interface LoopBack1
 ip address 1.1.1.1 255.255.255.255
#
interface LoopBack2
 ip address 11.11.11.11 255.255.255.255
#
interface Nve1
 source 1.1.1.1
 pip-source 11.11.11.11 peer 10.10.10.10 bypass
 mac-address 00e0-fc00-0101
#
bgp 100
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack0
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 peer 9.9.9.9 as-number 100
 peer 9.9.9.9 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 6.6.6.6 enable
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
```

```
 peer 9.9.9.9 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route static
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 advertise irb
  peer 6.6.6.6 advertise irbv6
  peer 6.6.6.6 reflect-client
  peer 7.7.7.7 enable
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 reflect-client
  peer 8.8.8.8 enable
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 reflect-client
  peer 9.9.9.9 enable
  peer 9.9.9.9 advertise irb
  peer 9.9.9.9 advertise irbv6
  peer 9.9.9.9 reflect-client
#
ospf 1
 area 0.0.0.0
  network 1.1.1.1 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.5.0 0.0.0.255
  network 192.168.6.0 0.0.0.255
  network 192.168.7.0 0.0.0.255
  network 192.168.8.0 0.0.0.255
#
ip route-static 0.0.0.0 0.0.0.0 10.1.30.2
ip route-static 0.0.0.0 0.0.0.0 10.1.40.2
ip route-static 10.1.1.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.1.2.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.1.3.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.10.10.10 32 10.1.1.1 preference 1
ip route-static vpn-instance vpn1 0.0.0.0 0.0.0.0 public
#
ipv6 route-static :: 0 FC00:30::2
ipv6 route-static :: 0 FC00:40::2
ipv6 route-static FC00:1:: 64 vpn-instance vpn1
ipv6 route-static FC00:2:: 64 vpn-instance vpn1
ipv6 route-static FC00:3:: 64 vpn-instance vpn1
ipv6 route-static vpn-instance vpn1 :: 0 public
#
return
```

- Leaf1的配置脚本

```
#
sysname Leaf1
#
dfs-group 1
 priority 150
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
 dual-active detection source ip 6.6.6.6 peer 7.7.7.7
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
```



```
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 6.6.6.6:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 6.6.6.6:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 6.6.6.6:10
  vpn-target 0:10 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:10 import-extcommunity
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 6.6.6.6:20
  vpn-target 0:20 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:20 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0105
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.2.1 255.255.255.0
 ipv6 address FC00:2::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0106
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.2.2.1 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk2
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2.10 mode l2
 encapsulation dot1q vid 10
```

```
 bridge-domain 10
#
interface Eth-Trunk3
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk3.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.5.2 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/3
 eth-trunk 1
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 2
#
interface 100GE1/0/6
 eth-trunk 3
#
interface LoopBack0
 ip address 6.6.6.6 255.255.255.255
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
#
interface LoopBack2
 ip address 12.12.12.12 255.255.255.255
#
interface Nve1
 source 2.2.2.2
 pip-source 12.12.12.12 peer 13.13.13.13 bypass
 vni 10 head-end peer-list protocol bgp
 vni 20 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0102
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk2 downlink 1
 port Eth-Trunk3 downlink 2
#
bgp 100
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack0
 peer 5.5.5.5 as-number 100
 peer 5.5.5.5 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 4.4.4.4 enable
  peer 5.5.5.5 enable
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
```

```
 ipv6-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  policy vpn-target
  peer 4.4.4.4 enable
  peer 4.4.4.4 advertise irb
  peer 4.4.4.4 advertise irbv6
  peer 5.5.5.5 enable
  peer 5.5.5.5 advertise irb
  peer 5.5.5.5 advertise irbv6
 #
ospf 1
 area 0.0.0.0
  network 2.2.2.2 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.5.0 0.0.0.255
#
ip route-static 13.13.13.13 32 10.2.2.2 preference 1
#
return
```

- Leaf2的配置脚本

```
#
sysname Leaf2
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 7.7.7.7 peer 6.6.6.6
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 7.7.7.7:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 7.7.7.7:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 7.7.7.7:10
  vpn-target 0:10 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:10 import-extcommunity
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 7.7.7.7:20
  vpn-target 0:20 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:20 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
```

```
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0105
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ipv6 address FC00:2::1/64
 ip address 10.1.2.1 255.255.255.0
 arp broadcast-detect enable
 mac-address 00e0-fc00-0106
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.2.2.2 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk2
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2.10 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface Eth-Trunk3
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk3.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.2.2 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.6.2 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/3
 eth-trunk 1
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 2
#
interface 100GE1/0/6
 eth-trunk 3
```

```
#
interface LoopBack0
 ip address 7.7.7.7 255.255.255.255
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
#
interface LoopBack2
 ip address 13.13.13.13 255.255.255.255
#
interface Nve1
 source 2.2.2.2
 pip-source 13.13.13.13 peer 12.12.12.12 bypass
 vni 10 head-end peer-list protocol bgp
 vni 20 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0102
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk2 downlink 1
 port Eth-Trunk3 downlink 2
#
bgp 100
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack0
 peer 5.5.5.5 as-number 100
 peer 5.5.5.5 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 4.4.4.4 enable
  peer 5.5.5.5 enable
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  policy vpn-target
  peer 4.4.4.4 enable
  peer 4.4.4.4 advertise irb
  peer 4.4.4.4 advertise irbv6
  peer 5.5.5.5 enable
  peer 5.5.5.5 advertise irb
  peer 5.5.5.5 advertise irbv6
#
ospf 1
 area 0.0.0.0
  network 2.2.2.2 0.0.0.0
  network 7.7.7.7 0.0.0.0
  network 192.168.2.0 0.0.0.255
  network 192.168.6.0 0.0.0.255
#
ip route-static 12.12.12.12 32 10.2.2.1 preference 1
return
```

- Leaf3的配置脚本

```
#
sysname Leaf3
#
dfs-group 1
 priority 150
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
 dual-active detection source ip 8.8.8.8 peer 9.9.9.9
#
```

```
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 8.8.8.8:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 8.8.8.8:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 8.8.8.8:20
  vpn-target 0:20 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:20 import-extcommunity
#
bridge-domain 30
 vxlan vni 30
 evpn
  route-distinguisher 8.8.8.8:30
  vpn-target 0:30 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:30 import-extcommunity
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ipv6 address FC00:2::1/64
 ip address 10.1.2.1 255.255.255.0
 arp broadcast-detect enable
 mac-address 00e0-fc00-0106
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.3.1 255.255.255.0
 ipv6 address FC00:3::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0107
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.3.3.1 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk2
 stp edged-port enable
```

```
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface Eth-Trunk3
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk3.30 mode l2
 encapsulation dot1q vid 30
 bridge-domain 30
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.3.2 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.7.2 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/3
 eth-trunk 1
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 2
#
interface 100GE1/0/6
 eth-trunk 3
#
interface LoopBack0
 ip address 8.8.8.8 255.255.255.255
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.255
#
interface LoopBack2
 ip address 14.14.14.14 255.255.255.255
#
interface Nve1
 source 3.3.3.3
 pip-source 14.14.14.14 peer 15.15.15.15 bypass
 vni 20 head-end peer-list protocol bgp
 vni 30 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0103
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk2 downlink 1
 port Eth-Trunk3 downlink 2
#
bgp 100
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack0
 peer 5.5.5.5 as-number 100
 peer 5.5.5.5 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 4.4.4.4 enable
  peer 5.5.5.5 enable
```



```
#
ipv6-family vpn-instance vpn1
 import-route direct
 advertise l2vpn evpn
#
ipv6-family vpn-instance vpn1
 import-route direct
 advertise l2vpn evpn
#
l2vpn-family evpn
 policy vpn-target
 peer 4.4.4.4 enable
 peer 4.4.4.4 advertise irb
 peer 4.4.4.4 advertise irbv6
 peer 5.5.5.5 enable
 peer 5.5.5.5 advertise irb
 peer 5.5.5.5 advertise irbv6
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 8.8.8.8 0.0.0.0
  network 192.168.3.0 0.0.0.255
  network 192.168.7.0 0.0.0.255
#
ip route-static 15.15.15.15 32 10.3.3.2 preference 1
return
```

- Leaf4的配置脚本

```
#
sysname Leaf4
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
 dual-active detection source ip 9.9.9.9 peer 8.8.8.8
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 9.9.9.9:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 9.9.9.9:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 9.9.9.9:20
  vpn-target 0:20 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:20 import-extcommunity
#
bridge-domain 30
 vxlan vni 30
 evpn
  route-distinguisher 9.9.9.9:30
  vpn-target 0:30 export-extcommunity
  vpn-target 0:1 export-extcommunity
```

```
 vpn-target 0:30 import-extcommunity
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.2.1 255.255.255.0
 ipv6 address FC00:2::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0106
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.3.1 255.255.255.0
 ipv6 address FC00:3::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0107
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.3.3.2 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk2
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface Eth-Trunk3
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk3.30 mode l2
 encapsulation dot1q vid 30
 bridge-domain 30
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.4.2 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.8.2 255.255.255.0
 ospf network-type p2p
#
interface 100GE1/0/3
 eth-trunk 1
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
```

```
 eth-trunk 2
#
interface 100GE1/0/6
 eth-trunk 3
#
interface LoopBack0
 ip address 9.9.9.9 255.255.255.255
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.255
#
interface LoopBack2
 ip address 15.15.15.15 255.255.255.255
#
interface Nve1
 source 3.3.3.3
 pip-source 15.15.15.15 peer 14.14.14.14 bypass
 vni 20 head-end peer-list protocol bgp
 vni 30 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0103
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk2 downlink 1
 port Eth-Trunk3 downlink 2
#
bgp 100
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack0
 peer 5.5.5.5 as-number 100
 peer 5.5.5.5 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 4.4.4.4 enable
  peer 5.5.5.5 enable
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  policy vpn-target
  peer 4.4.4.4 enable
  peer 4.4.4.4 advertise irb
  peer 4.4.4.4 advertise irbv6
  peer 5.5.5.5 enable
  peer 5.5.5.5 advertise irb
  peer 5.5.5.5 advertise irbv6
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 9.9.9.9 0.0.0.0
  network 192.168.4.0 0.0.0.255
  network 192.168.8.0 0.0.0.255
#
ip route-static 14.14.14.14 32 10.3.3.1 preference 1
return
```

1.4 配置分布式网关部署方式的 IPv6 VXLAN 示例

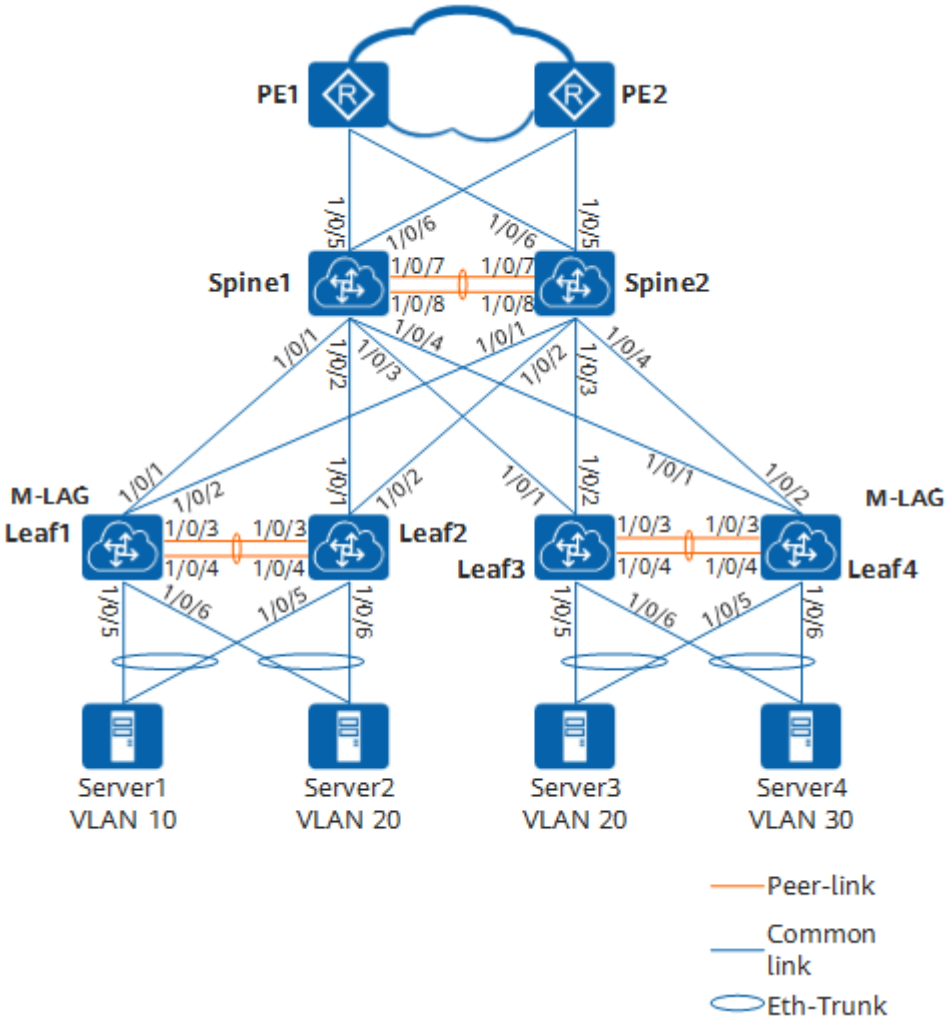
适用产品和版本

- CE16800（除X系列单板外）、CE8800、CE6800（除CE6820H、CE6820H-K、CE6820S外）系列产品V300R021C10或更高版本。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

如**图1-4**所示，某企业新建的数据中心网络采用分布式网关部署方式，其中Underlay基础网络为IPv6，Overlay网络为IPv4/IPv6。Leaf作为三层网关与服务器对接；Spine同时作为东西向流量的汇聚设备和网络出口网关。为了保证高可靠性，Spine、Leaf采用M-LAG部署方式。

图 1-4 分布式网关部署方式的 VXLAN 网络示意图



说明

上图中“1/0/1”为接口编号，接口速率为100GE，即“1/0/1”表示接口“100GE1/0/1”。其他接口类似。

表 1-4 接口地址表

设备名称	接口	IP地址	设备名称	接口	IP地址
Spine 1	100GE1/0/1	2001:db8:1:d301::1/64	Spine 2	100GE1/0/1	2001:db8:1:d305::1/64
	100GE1/0/2	2001:db8:1:d302::1/64		100GE1/0/2	2001:db8:1:d306::1/64
	100GE1/0/3	2001:db8:1:d303::1/64		100GE1/0/3	2001:db8:1:d307::1/64
	100GE1/0/4	2001:db8:1:d304::1/64		100GE1/0/4	2001:db8:1:d308::1/64
	100GE1/0/5	IPv4: 10.1.10.1/24 IPv6: fc00:10::1/64		100GE1/0/5	IPv4: 10.1.30.1/24 IPv6: fc00:30::1/64
	100GE1/0/6	IPv4: 10.1.20.1/24 IPv6: fc00:20::1/64		100GE1/0/6	IPv4: 10.1.40.1/24 IPv6: fc00:40::1/64
	Loopback0	4.4.4.4/32		Loopback0	5.5.5.5/32
	Loopback1	2001:db8:1:1301::1/128		Loopback1	2001:db8:1:1301::1/128
	Loopback2	2001:db8:1:4301::1/128		Loopback2	2001:db8:1:5301::1/128
	Loopback3	2001:db8:1:4302::1/128		Loopback3	2001:db8:1:5302::1/128
Leaf1	100GE1/0/1	2001:db8:1:d301::2/64	Leaf2	100GE1/0/1	2001:db8:1:d302::2/64
	100GE1/0/2	2001:db8:1:d305::2/64		100GE1/0/2	2001:db8:1:d306::2/64
	Loopback0	6.6.6.6/32		Loopback0	7.7.7.7/32
	Loopback1	2001:db8:1:2301::1/128		Loopback1	2001:db8:1:2301::1/128
	Loopback2	2001:db8:1:6301::1/128		Loopback2	2001:db8:1:7301::1/128

设备名称	接口	IP地址	设备名称	接口	IP地址
	Loopback3	2001:db8:1:6302::1/128		Loopback3	2001:db8:1:7302::1/128
Leaf3	100GE1/0/1	2001:db8:1:d303::2/64	Leaf4	100GE1/0/1	2001:db8:1:d304::2/64
	100GE1/0/2	2001:db8:1:d307::2/64		100GE1/0/2	2001:db8:1:d308::2/64
	Loopback0	8.8.8.8/32		Loopback0	9.9.9.9/32
	Loopback1	2001:db8:1:3301::1/128		Loopback1	2001:db8:1:3301::1/128
	Loopback2	2001:db8:1:8301::1/128		Loopback2	2001:db8:1:9301::1/128
	Loopback3	2001:db8:1:8302::1/128		Loopback3	2001:db8:1:9302::1/128

配置思路

采用如下思路配置分布式网关部署方式的VXLAN网络：

1. 配置路由协议，保证Underlay网络三层互通。
2. 配置M-LAG，实现服务器双活接入。
3. 配置BGP EVPN建立VXLAN隧道。

操作步骤

步骤1 配置路由协议，实现Underlay网络三层互通。

配置Leaf1。其他设备的配置与Leaf1类似，这里不再赘述，具体配置请参考**配置脚本**。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf1
[*HUAWEI] commit
[~Leaf1] ospfv3
[*Leaf1-ospfv3-1] router-id 6.6.6.6
[*Leaf1-ospfv3-1] quit
[*Leaf1] interface 100ge 1/0/1
[*Leaf1-100GE1/0/1] undo portswitch
[*Leaf1-100GE1/0/1] ipv6 enable
[*Leaf1-100GE1/0/1] ipv6 address 2001:db8:1:d301::2 64
[*Leaf1-100GE1/0/1] ospfv3 1 area 0
[*Leaf1-100GE1/0/1] ospfv3 network-type p2p
[*Leaf1-100GE1/0/1] quit
[*Leaf1] interface 100ge 1/0/2
[*Leaf1-100GE1/0/2] undo portswitch
[*Leaf1-100GE1/0/2] ipv6 enable
[*Leaf1-100GE1/0/2] ipv6 address 2001:db8:1:d305::2 64
[*Leaf1-100GE1/0/2] ospfv3 1 area 0
[*Leaf1-100GE1/0/2] ospfv3 network-type p2p
```

```
[*Leaf1-100GE1/0/2] quit
[*Leaf1] interface loopback 0
[*Leaf1-LoopBack0] ip address 6.6.6.6 32
[*Leaf1-LoopBack0] quit
[*Leaf1] interface loopback 1
[*Leaf1-LoopBack1] ipv6 enable
[*Leaf1-LoopBack1] ipv6 address 2001:db8:1:2301::1 128
[*Leaf1-LoopBack1] ospfv3 1 area 0
[*Leaf1-LoopBack1] quit
[*Leaf1] interface loopback 2
[*Leaf1-LoopBack2] ipv6 enable
[*Leaf1-LoopBack2] ipv6 address 2001:db8:1:6301::1 128
[*Leaf1-LoopBack2] ospfv3 1 area 0
[*Leaf1-LoopBack2] quit
[*Leaf1] interface loopback 3
[*Leaf1-LoopBack3] ipv6 enable
[*Leaf1-LoopBack3] ipv6 address 2001:db8:1:6302::1 128
[*Leaf1-LoopBack3] quit
[*Leaf1] commit
```

OSPFv3成功配置后，Leaf、Spine之间可通过OSPFv3协议发现对方的Loopback接口的IPv6地址，并能互相ping通。

步骤2 配置Leaf组成M-LAG系统。本示例中Leaf1、Leaf2组成M-LAG系统。Leaf3、Leaf4组成M-LAG系统，具体配置与此类似，不再赘述。

配置Leaf1。

```
[~Leaf1] stp mode rstp
[*Leaf1] stp v-stp enable
[*Leaf1] dfs-group 1
[*Leaf1-dfs-group-1] dual-active detection source ipv6 2001:DB8:1:6301::1 peer 2001:db8:1:7301::1
[*Leaf1-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Leaf1-dfs-group-1] priority 150
[*Leaf1-dfs-group-1] quit
[*Leaf1] interface eth-trunk 1
[*Leaf1-Eth-Trunk1] trunkport 100ge 1/0/3
[*Leaf1-Eth-Trunk1] trunkport 100ge 1/0/4
[*Leaf1-Eth-Trunk1] mode lacp-static
[*Leaf1-Eth-Trunk1] peer-link 1
[*Leaf1-Eth-Trunk1] quit
[*Leaf1] interface eth-trunk 2
[*Leaf1-Eth-Trunk2] trunkport 100ge 1/0/5
[*Leaf1-Eth-Trunk2] mode lacp-static
[*Leaf1-Eth-Trunk2] dfs-group 1 m-lag 1
[*Leaf1-Eth-Trunk2] stp edged-port enable
[*Leaf1-Eth-Trunk2] quit
[*Leaf1] interface eth-trunk 3
[*Leaf1-Eth-Trunk3] trunkport 100ge 1/0/6
[*Leaf1-Eth-Trunk3] mode lacp-static
[*Leaf1-Eth-Trunk3] dfs-group 1 m-lag 2
[*Leaf1-Eth-Trunk3] stp edged-port enable
[*Leaf1-Eth-Trunk3] quit
[*Leaf1] commit
[~Leaf1] monitor-link group 1
[*Leaf1-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf1-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf1-mtlk-group1] port eth-trunk 2 downlink 1
[*Leaf1-mtlk-group1] port eth-trunk 3 downlink 2
[*Leaf1-mtlk-group1] quit
[*Leaf1] commit
```

#配置Leaf2。

```
[~Leaf2] stp mode rstp
[*Leaf2] stp v-stp enable
[*Leaf2] dfs-group 1
[*Leaf2-dfs-group-1] dual-active detection source ipv6 2001:DB8:1:6301::1 peer 2001:db8:1:7301::1
[*Leaf2-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
```



```
[*Leaf2-dfs-group-1] quit
[*Leaf2] interface eth-trunk 1
[*Leaf2-Eth-Trunk1] trunkport 100ge 1/0/3
[*Leaf2-Eth-Trunk1] trunkport 100ge 1/0/4
[*Leaf2-Eth-Trunk1] mode lacp-static
[*Leaf2-Eth-Trunk1] peer-link 1
[*Leaf2-Eth-Trunk1] quit
[*Leaf2] interface eth-trunk 2
[*Leaf2-Eth-Trunk2] mode lacp-static
[*Leaf2-Eth-Trunk2] trunkport 100ge 1/0/5
[*Leaf2-Eth-Trunk2] dfs-group 1 m-lag 1
[*Leaf2-Eth-Trunk2] stp edged-port enable
[*Leaf2-Eth-Trunk2] quit
[*Leaf2] interface eth-trunk 3
[*Leaf2-Eth-Trunk3] mode lacp-static
[*Leaf2-Eth-Trunk3] trunkport 100ge 1/0/6
[*Leaf2-Eth-Trunk3] dfs-group 1 m-lag 2
[*Leaf2-Eth-Trunk3] stp edged-port enable
[*Leaf2-Eth-Trunk3] quit
[*Leaf2] commit
[~Leaf2] monitor-link group 1
[*Leaf2-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf2-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf2-mtlk-group1] port eth-trunk 2 downlink 1
[*Leaf2-mtlk-group1] port eth-trunk 3 downlink 2
[*Leaf2-mtlk-group1] quit
[*Leaf2] commit
```

步骤3 配置Spine1、Spine2组成M-LAG系统。

配置Spine1。

```
[~Spine1] stp mode rstp
[*Spine1] stp v-stp enable
[*Spine1] dfs-group 1
[*Spine1-dfs-group-1] dual-active detection source ipv6 2001:db8:1:4301::1 peer 2001:db8:1:5301::1
[*Spine1-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Spine1-dfs-group-1] priority 150
[*Spine1-dfs-group-1] quit
[*Spine1] interface eth-trunk 1
[*Spine1-Eth-Trunk1] trunkport 100ge 1/0/7
[*Spine1-Eth-Trunk1] trunkport 100ge 1/0/8
[*Spine1-Eth-Trunk1] mode lacp-static
[*Spine1-Eth-Trunk1] peer-link 1
[*Spine1-Eth-Trunk1] quit
[*Spine1] commit
```

#配置Spine2。

```
[~Spine2] stp mode rstp
[*Spine2] stp v-stp enable
[*Spine2] dfs-group 1
[*Spine2-dfs-group-1] dual-active detection source ipv6 2001:DB8:1:5301::1 peer 2001:db8:1:4301::1
[*Spine2-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Spine2-dfs-group-1] quit
[*Spine2] interface eth-trunk 1
[*Spine2-Eth-Trunk1] trunkport 100ge 1/0/7
[*Spine2-Eth-Trunk1] trunkport 100ge 1/0/8
[*Spine2-Eth-Trunk1] mode lacp-static
[*Spine2-Eth-Trunk1] peer-link 1
[*Spine2-Eth-Trunk1] quit
[*Spine2] commit
```

步骤4 配置BGP EVPN，建立VXLAN隧道。

1. 配置业务接入点。

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述。

   ```
   [~Leaf1] bridge-domain 10
   [*Leaf1-bd10] quit
   ```

```
[*Leaf1] bridge-domain 20
[*Leaf1-bd20] quit
[*Leaf1] interface eth-trunk 2.10 mode l2
[*Leaf1-Eth-Trunk2.1] encapsulation dot1q vid 10
[*Leaf1-Eth-Trunk2.1] bridge-domain 10
[*Leaf1-Eth-Trunk2.1] quit
[*Leaf1] interface eth-trunk 3.20 mode l2
[*Leaf1-Eth-Trunk3.1] encapsulation dot1q vid 20
[*Leaf1-Eth-Trunk3.1] bridge-domain 20
[*Leaf1-Eth-Trunk3.1] quit
[*Leaf1] commit
```

2. 配置BGP EVPN对等体关系。Spine1、Spine2作为路由反射器。

   # 配置Spine1。Spine2的配置与Spine1类似，这里不再赘述，具体配置请参考**配置脚本**。

```
[~Spine1] evpn-overlay enable
[*Spine1] bgp 100
[*Spine1-bgp] peer 2001:db8:1:6301::1 as-number 100
[*Spine1-bgp] peer 2001:db8:1:6301::1 connect-interface LoopBack2
[*Spine1-bgp] peer 2001:db8:1:7301::1 as-number 100
[*Spine1-bgp] peer 2001:db8:1:7301::1 connect-interface LoopBack2
[*Spine1-bgp] peer 2001:db8:1:8301::1 as-number 100
[*Spine1-bgp] peer 2001:db8:1:8301::1 connect-interface LoopBack2
[*Spine1-bgp] peer 2001:db8:1:9301::1 as-number 100
[*Spine1-bgp] peer 2001:db8:1:9301::1 connect-interface LoopBack2
[*Spine1-bgp] l2vpn-family evpn
[*Spine1-bgp-af-evpn] peer 2001:db8:1:6301::1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 2001:db8:1:6301::1 advertise irb
[*Spine1-bgp-af-evpn] peer 2001:db8:1:6301::1 advertise irbv6
[*Spine1-bgp-af-evpn] peer 2001:db8:1:6301::1 reflect-client
[*Spine1-bgp-af-evpn] peer 2001:db8:1:7301::1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 2001:db8:1:7301::1 advertise irb
[*Spine1-bgp-af-evpn] peer 2001:db8:1:7301::1 advertise irbv6
[*Spine1-bgp-af-evpn] peer 2001:db8:1:7301::1 reflect-client
[*Spine1-bgp-af-evpn] peer 2001:db8:1:8301::1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 2001:db8:1:8301::1 advertise irb
[*Spine1-bgp-af-evpn] peer 2001:db8:1:8301::1 advertise irbv6
[*Spine1-bgp-af-evpn] peer 2001:db8:1:8301::1 reflect-client
[*Spine1-bgp-af-evpn] peer 2001:db8:1:9301::1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 2001:db8:1:9301::1 advertise irb
[*Spine1-bgp-af-evpn] peer 2001:db8:1:9301::1 advertise irbv6
[*Spine1-bgp-af-evpn] peer 2001:db8:1:9301::1 reflect-client
[*Spine1-bgp-af-evpn] undo policy vpn-target
[*Spine1-bgp-af-evpn] quit
[*Spine1-bgp] quit
[*Spine1] commit
```

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述，具体配置请参考**配置脚本**。

```
[~Leaf1] evpn-overlay enable
[*Leaf1] bgp 100
[*Leaf1-bgp] peer 2001:db8:1:4301::1 as-number 100
[*Leaf1-bgp] peer 2001:db8:1:4301::1 connect-interface LoopBack2
[*Leaf1-bgp] peer 2001:db8:1:5301::1 as-number 100
[*Leaf1-bgp] peer 2001:db8:1:5301::1 connect-interface LoopBack2
[*Leaf1-bgp] l2vpn-family evpn
[*Leaf1-bgp-af-evpn] peer 2001:db8:1:4301::1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf1-bgp-af-evpn] peer 2001:db8:1:4301::1 advertise irb
[*Leaf1-bgp-af-evpn] peer 2001:db8:1:4301::1 advertise irbv6
[*Leaf1-bgp-af-evpn] peer 2001:db8:1:5301::1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf1-bgp-af-evpn] peer 2001:db8:1:5301::1 advertise irb
[*Leaf1-bgp-af-evpn] peer 2001:db8:1:5301::1 advertise irbv6
[*Leaf1-bgp-af-evpn] quit
```

```
[*Leaf1-bgp] quit
[*Leaf1] commit
```

3. 配置VPN实例及EVPN实例。

   # 配置Spine1。Spine2的配置与Spine1类似，这里不再赘述，具体配置请参考**配置脚本**。

```
[~Spine1] ip vpn-instance vpn1
[*Spine1-vpn-instance-vpn1] vxlan vni 5000
[*Spine1-vpn-instance-vpn1] ipv4-family
[*Spine1-vpn-instance-vpn1-af-ipv4] route-distinguisher 4.4.4.4:1
[*Spine1-vpn-instance-vpn1-af-ipv4] vpn-target 0:1 evpn
[*Spine1-vpn-instance-vpn1-af-ipv4] quit
[*Spine1-vpn-instance-vpn1] ipv6-family
[*Spine1-vpn-instance-vpn1-af-ipv6] route-distinguisher 4.4.4.4:1
[*Spine1-vpn-instance-vpn1-af-ipv6] vpn-target 0:1 evpn
[*Spine1-vpn-instance-vpn1-af-ipv6] quit
[*Spine1-vpn-instance-vpn1] quit
[*Spine1] bgp 100
[*Spine1-bgp] ipv4-family vpn-instance vpn1
[*Spine1-bgp-vpn1] import-route static
[*Spine1-bgp-vpn1] advertise l2vpn evpn
[*Spine1-bgp-vpn1] quit
[*Spine1-bgp] ipv6-family vpn-instance vpn1
[*Spine1-bgp-6-vpn1] import-route static
[*Spine1-bgp-6-vpn1] advertise l2vpn evpn
[*Spine1-bgp-6-vpn1] quit
[*Spine1-bgp] quit
[*Spine1] interface nve 1   //配置NVE
[*Spine1-Nve1] source 2001:db8:1:1301::1   //Spine1和Spine2作为M-LAG双活系统，这两台设备上配置的NVE接口的IP地址和MAC地址需要相同
[*Spine1-Nve1] mac-address 0000-5e00-0101
[*Spine1-Nve1] quit
[*Spine1] commit
```

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述，具体配置请参考**配置脚本**。

```
[~Leaf1] ip vpn-instance vpn1
[*Leaf1-vpn-instance-vpn1] vxlan vni 5000
[*Leaf1-vpn-instance-vpn1] ipv4-family
[*Leaf1-vpn-instance-vpn1-af-ipv4] route-distinguisher 6.6.6.6:1
[*Leaf1-vpn-instance-vpn1-af-ipv4] vpn-target 0:1 evpn
[*Leaf1-vpn-instance-vpn1-af-ipv4] quit
[*Leaf1-vpn-instance-vpn1] ipv6-family
[*Leaf1-vpn-instance-vpn1-af-ipv6] route-distinguisher 6.6.6.6:1
[*Leaf1-vpn-instance-vpn1-af-ipv6] vpn-target 0:1 evpn
[*Leaf1-vpn-instance-vpn1-af-ipv6] quit
[*Leaf1-vpn-instance-vpn1] quit
[*Leaf1] bridge-domain 10
[*Leaf1-bd10] vxlan vni 10
[*Leaf1-bd10] evpn
[*Leaf1-bd10-evpn] route-distinguisher 6.6.6.6:10
[*Leaf1-bd10-evpn] vpn-target 0:10
[*Leaf1-bd10-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd10-evpn] quit
[*Leaf1-bd10] quit
[*Leaf1] bridge-domain 20
[*Leaf1-bd20] vxlan vni 20
[*Leaf1-bd20] evpn
[*Leaf1-bd20-evpn] route-distinguisher 6.6.6.6:20
[*Leaf1-bd20-evpn] vpn-target 0:20
[*Leaf1-bd20-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd20-evpn] quit
[*Leaf1-bd20] quit
[*Leaf1] bgp 100
[*Leaf1-bgp] ipv4-family vpn-instance vpn1
[*Leaf1-bgp-vpn1] import-route direct
[*Leaf1-bgp-vpn1] advertise l2vpn evpn
[*Leaf1-bgp-vpn1] quit
[*Leaf1-bgp] ipv6-family vpn-instance vpn1
```

```
[*Leaf1-bgp-6-vpn1] import-route direct
[*Leaf1-bgp-6-vpn1] advertise l2vpn evpn
[*Leaf1-bgp-6-vpn1] quit
[*Leaf1-bgp] quit
[*Leaf1] interface nve 1      //配置NVE
[*Leaf1-Nve1] source 2001:db8:1:2301::1   //Leaf1和Leaf2作为M-LAG双活系统，这两台设备上配置的NVE接口的IP地址和MAC地址需要相同
[*Leaf1-Nve1] mac-address 0000-5e00-0102
[*Leaf1-Nve1] vni 10 head-end peer-list protocol bgp
[*Leaf1-Nve1] vni 20 head-end peer-list protocol bgp
[*Leaf1-Nve1] quit
[*Leaf1] commit
```

4. 在Leaf1、Leaf2、Leaf3、Leaf4上配置三层网关。

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述，具体配置请参考**配置脚本**。

```
[~Leaf1] interface vbdif 10
[*Leaf1-Vbdif10] ip binding vpn-instance vpn1
[*Leaf1-Vbdif10] ip address 10.1.1.1 24   //Leaf1和Leaf2作为M-LAG双活系统，这两台设备上配置的VBDIF接口的IP地址和MAC地址需要相同
[*Leaf1-Vbdif10] ipv6 enable
[*Leaf1-Vbdif10] ipv6 address fc00:1::1 64
[*Leaf1-Vbdif10] mac-address 0000-5e00-0105
[*Leaf1-Vbdif10] vxlan anycast-gateway enable
[*Leaf1-Vbdif10] arp collect host enable
[*Leaf1-Vbdif10] ipv6 nd collect host enable
[*Leaf1-Vbdif10] quit
[*Leaf1] interface vbdif 20
[*Leaf1-Vbdif20] ip binding vpn-instance vpn1
[*Leaf1-Vbdif20] ip address 10.1.2.1 24
[*Leaf1-Vbdif20] ipv6 enable
[*Leaf1-Vbdif20] ipv6 address fc00:2::1 64
[*Leaf1-Vbdif20] mac-address 0000-5e00-0106
[*Leaf1-Vbdif20] vxlan anycast-gateway enable
[*Leaf1-Vbdif20] arp collect host enable
[*Leaf1-Vbdif20] ipv6 nd collect host enable
[*Leaf1-Vbdif20] quit
[*Leaf1] commit
```

步骤5 在M-LAG设备中配置静态Bypass VXLAN隧道。

在M-LAG双归接入VXLAN的场景中，当下行一条链路发生故障时，业务流量需绕行M-LAG设备之间的Peer-link链路。因此，在该场景下M-LAG设备之间必须配置静态Bypass VXLAN隧道，将绕行的业务流量引导至Peer-link链路上。

下面以Leaf1和Leaf2配置为例，Spine1、Spine2、Leaf3、Leaf4的配置与之类似，这里不再赘述，具体配置请参考**配置脚本**。

配置Leaf1。

```
[~Leaf1] vlan 100   //本VLAN不能划分给其他业务使用，本例中以100举例
[*Leaf1-vlan100] m-lag peer-link reserved   //仅允许peer-link加入到该VLAN
[*Leaf1-vlan100] quit
[*Leaf1] interface vlanif 100
[*Leaf1-Vlanif100] reserved for vxlan bypass   //指定peer-link接口上VLANIF的IPv6地址只给Bypass VXLAN隧道使用
[*Leaf1-Vlanif100] ipv6 enable
[*Leaf1-Vlanif100] ipv6 address 2001:db8:2::1 64   //配置静态Bypass VXLAN隧道的源端IPv6地址
[*Leaf1-Vlanif100] ospfv3 1 area 0
[*Leaf1-Vlanif100] quit
[*Leaf1] ipv6 route-static 2001:db8:1:7302::1 128 2001:db8:2::2 preference 1   //配置静态路由，打通Bypass VXLAN隧道
[*Leaf1] interface nve 1
[*Leaf1-Nve1] pip-source 2001:db8:1:6302::1 peer 2001:db8:1:7302::1 bypass   //创建静态Bypass VXLAN隧道，指定源端地址和对端地址
[*Leaf1-Nve1] quit
[*Leaf1] commit
```

配置Leaf2。

```
[~Leaf2] vlan 100
[*Leaf2-vlan100] m-lag peer-link reserved
[*Leaf2-vlan100] quit
[*Leaf2] interface vlanif 100
[*Leaf2-Vlanif100] reserved for vxlan bypass
[*Leaf2-Vlanif100] ipv6 address 2001:db8:2::2 64
[*Leaf2-Vlanif100] ospfv3 1 area 0
[*Leaf2-Vlanif100] quit
[*Leaf2] ip route-static 2001:db8:1:6302::1 128 2001:db8:2::1 preference 1
[*Leaf2] interface nve 1
[*Leaf2-Nve1] pip-source 2001:db8:1:7302::1 peer 2001:db8:1:6302::1 bypass
[*Leaf2-Nve1] quit
[*Leaf2] commit
```

步骤6 在Spine1、Spine2上配置静态路由，实现南、北向流量互通。

配置Spine1。Spine2的配置与Spine1类似，这里不再赘述，具体配置请参考**配置脚本**。

```
[~Spine1] interface 100ge 1/0/5
[~Spine1-100GE1/0/5] undo portswitch
[*Spine1-100GE1/0/5] ip address 10.1.10.1 24
[*Spine1-100GE1/0/5] ipv6 enable
[*Spine1-100GE1/0/5] ipv6 address fc00:10::1 64
[*Spine1-100GE1/0/5] quit
[~Spine1] interface 100ge 1/0/6
[~Spine1-100GE1/0/6] undo portswitch
[~Spine1-100GE1/0/6] ip address 10.1.20.1 24
[~Spine1-100GE1/0/6] ipv6 enable
[*Spine1-100GE1/0/6] ipv6 address fc00:20::1 64
[*Spine1-100GE1/0/6] quit
[*Spine1] ip route-static 0.0.0.0 0.0.0.0 10.1.10.2  //至公网PE的IPv4静态路由
[*Spine1] ip route-static 0.0.0.0 0.0.0.0 10.1.20.2
[*Spine1] ip route-static 10.1.1.0 24 vpn-instance vpn1  //至服务器网段的IPv4静态路由，下一跳为VPN实例
[*Spine1] ip route-static 10.1.2.0 24 vpn-instance vpn1
[*Spine1] ip route-static 10.1.3.0 24 vpn-instance vpn1
[*Spine1] ip route-static vpn-instance vpn1 0.0.0.0 0.0.0.0 public  //VPN实例的IPv4静态路由，下一跳为公网实例
[*Spine1] ipv6 route-static :: 0 fc00:10::2  //至公网PE的IPv6静态路由
[*Spine1] ipv6 route-static :: 0 fc00:20::2
[*Spine1] ipv6 route-static fc00:1:: 64 vpn-instance vpn1  //至服务器网段的IPv6静态路由，下一跳为VPN实例
[*Spine1] ipv6 route-static fc00:2:: 64 vpn-instance vpn1
[*Spine1] ipv6 route-static fc00:3:: 64 vpn-instance vpn1
[*Spine1] ipv6 route-static vpn-instance vpn1 :: 0 public  //VPN实例的IPv6静态路由，下一跳为公网实例
[*Spine1] commit
```

----结束

验证配置结果

上述配置成功后，执行**display vxlan tunnel**命令可查看到VXLAN隧道的信息。以Spine1显示为例。

```
[~Spine1] display vxlan tunnel
Number of vxlan tunnel : 2
Tunnel ID   Source                  Destination             State  Type     Uptime
--------------------------------------------------------------------------------
4026531841  2001:db8:1:1301::1      2001:db8:1:2301::1      up     dynamic  0035h21m
4026531842  2001:db8:1:1301::1      2001:db8:1:3301::1      up     dynamic  0036h21m
```

配置完成后，服务器之间可以相互通信。

配置脚本

- Spine1的配置脚本

```
#
sysname Spine1
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ipv6 2001:db8:1:4301::1 peer 2001:db8:1:5301::1
 priority 150
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 4.4.4.4:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 4.4.4.4:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
VLAN 100
 m-lag peer-link reserved
#
ospfv3 1
 router-id 4.4.4.4
 area 0.0.0.0
#
interface Vlanif100
 ipv6 enable
 ipv6 address 2001:db8:1::1/64
 ospfv3 1 area 0.0.0.0
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:db8:1:D301::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:db8:1:D302::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/3
 undo portswitch
 ipv6 enable
 ipv6 address 2001:db8:1:D303::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/4
 undo portswitch
 ipv6 enable
```

```
 ipv6 address 2001:db8:1:D304::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/5
 undo portswitch
 ipv6 enable
 ip address 10.1.10.1 255.255.255.0
 ipv6 address FC00:10::1/64
#
interface 100GE1/0/6
 undo portswitch
 ipv6 enable
 ip address 10.1.20.1 255.255.255.0
 ipv6 address FC00:20::1/64
#
interface 100GE1/0/7
 eth-trunk 1
#
interface 100GE1/0/8
 eth-trunk 1
#
interface LoopBack0
 ip address 4.4.4.4 255.255.255.255
#
interface LoopBack1
 ipv6 enable
 ipv6 address 2001:db8:1:1301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack2
 ipv6 enable
 ipv6 address 2001:db8:1:4301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack3
 ipv6 enable
 ipv6 address 2001:db8:1:4302::1/128
#
interface Nve1
 source 2001:DB8:1:1301::1
 pip-source 2001:db8:1:4302::1 peer 2001:db8:1:5302::1 bypass
 mac-address 0000-5e00-0101
#
bgp 100
 peer 2001:DB8:1:6301::1 as-number 100
 peer 2001:DB8:1:6301::1 connect-interface LoopBack2
 peer 2001:DB8:1:7301::1 as-number 100
 peer 2001:DB8:1:7301::1 connect-interface LoopBack2
 peer 2001:DB8:1:8301::1 as-number 100
 peer 2001:DB8:1:8301::1 connect-interface LoopBack2
 peer 2001:DB8:1:9301::1 as-number 100
 peer 2001:DB8:1:9301::1 connect-interface LoopBack2
 #
 ipv4-family unicast
 #
 ipv4-family vpn-instance vpn1
  import-route static
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route static
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 2001:DB8:1:6301::1 enable
  peer 2001:DB8:1:6301::1 advertise irb
  peer 2001:DB8:1:6301::1 advertise irbv6
```



```
 peer 2001:DB8:1:6301::1 reflect-client
 peer 2001:DB8:1:7301::1 enable
 peer 2001:DB8:1:7301::1 advertise irb
 peer 2001:DB8:1:7301::1 advertise irbv6
 peer 2001:DB8:1:7301::1 reflect-client
 peer 2001:DB8:1:8301::1 enable
 peer 2001:DB8:1:8301::1 advertise irb
 peer 2001:DB8:1:8301::1 advertise irbv6
 peer 2001:DB8:1:8301::1 reflect-client
 peer 2001:DB8:1:9301::1 enable
 peer 2001:DB8:1:9301::1 advertise irb
 peer 2001:DB8:1:9301::1 advertise irbv6
 peer 2001:DB8:1:9301::1 reflect-client
#
ip route-static 0.0.0.0 0.0.0.0 10.1.10.2
ip route-static 0.0.0.0 0.0.0.0 10.1.20.2
ip route-static 10.1.1.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.1.2.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.1.3.0 255.255.255.0 vpn-instance vpn1
ip route-static vpn-instance vpn1 0.0.0.0 0.0.0.0 public
#
ipv6 route-static 2001:db8:1:5302::1 128 2001:db8:1::2 preference 1
ipv6 route-static :: 0 FC00:10::2
ipv6 route-static :: 0 FC00:20::2
ipv6 route-static FC00:1:: 64 vpn-instance vpn1
ipv6 route-static FC00:2:: 64 vpn-instance vpn1
ipv6 route-static FC00:3:: 64 vpn-instance vpn1
ipv6 route-static vpn-instance vpn1 :: 0 public
#
return
```

- Spine2的配置文件

```
#
sysname Spine2
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ipv6 2001:DB8:1:5301::1 peer 2001:db8:1:4301::1
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 5.5.5.5:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 5.5.5.5:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
VLAN 100
 m-lag peer-link reserved
#
ospfv3 1
 router-id 5.5.5.5
 area 0.0.0.0
#
interface Vlanif100
 ipv6 enable
 ipv6 address 2001:db8:1::2/64
 ospfv3 1 area 0.0.0.0
 reserved for vxlan bypass
#
interface Eth-Trunk1
```

```
 mode lacp-static
 peer-link 1
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D305::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D306::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/3
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D307::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/4
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D308::1/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/5
 undo portswitch
 ipv6 enable
 ip address 10.1.30.1 255.255.255.0
 ipv6 address FC00:30::1/64
#
interface 100GE1/0/6
 undo portswitch
 ipv6 enable
 ip address 10.1.40.1 255.255.255.0
 ipv6 address FC00:40::1/64
#
interface 100GE1/0/7
 eth-trunk 1
#
interface 100GE1/0/8
 eth-trunk 1
#
interface LoopBack0
 ip address 5.5.5.5 255.255.255.255
#
interface LoopBack1
 ipv6 enable
 ipv6 address 2001:DB8:1:1301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack2
 ipv6 enable
 ipv6 address 2001:DB8:1:5301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack3
 ipv6 enable
 ipv6 address 2001:db8:1:5302::1/128
#
interface Nve1
 source 2001:DB8:1:1301::1
 pip-source 2001:db8:1:5302::1 peer 2001:db8:1:4302::1 bypass
```

```
 mac-address 0000-5e00-0101
#
bgp 100
 peer 2001:DB8:1:6301::1 as-number 100
 peer 2001:DB8:1:6301::1 connect-interface LoopBack2
 peer 2001:DB8:1:7301::1 as-number 100
 peer 2001:DB8:1:7301::1 connect-interface LoopBack2
 peer 2001:DB8:1:8301::1 as-number 100
 peer 2001:DB8:1:8301::1 connect-interface LoopBack2
 peer 2001:DB8:1:9301::1 as-number 100
 peer 2001:DB8:1:9301::1 connect-interface LoopBack2
 #
 ipv4-family unicast
 #
 ipv4-family vpn-instance vpn1
  import-route static
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route static
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 2001:DB8:1:6301::1 enable
  peer 2001:DB8:1:6301::1 advertise irb
  peer 2001:DB8:1:6301::1 advertise irbv6
  peer 2001:DB8:1:6301::1 reflect-client
  peer 2001:DB8:1:7301::1 enable
  peer 2001:DB8:1:7301::1 advertise irb
  peer 2001:DB8:1:7301::1 advertise irbv6
  peer 2001:DB8:1:7301::1 reflect-client
  peer 2001:DB8:1:8301::1 enable
  peer 2001:DB8:1:8301::1 advertise irb
  peer 2001:DB8:1:8301::1 advertise irbv6
  peer 2001:DB8:1:8301::1 reflect-client
  peer 2001:DB8:1:9301::1 enable
  peer 2001:DB8:1:9301::1 advertise irb
  peer 2001:DB8:1:9301::1 advertise irbv6
  peer 2001:DB8:1:9301::1 reflect-client
#
ip route-static 0.0.0.0 0.0.0.0 10.1.30.2
ip route-static 0.0.0.0 0.0.0.0 10.1.40.2
ip route-static 10.1.1.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.1.2.0 255.255.255.0 vpn-instance vpn1
ip route-static 10.1.3.0 255.255.255.0 vpn-instance vpn1
ip route-static vpn-instance vpn1 0.0.0.0 0.0.0.0 public
#
ipv6 route-static 2001:db8:1:4302::1 128 2001:db8:1::1 preference 1
ipv6 route-static :: 0 FC00:30::2
ipv6 route-static :: 0 FC00:40::2
ipv6 route-static FC00:1:: 64 vpn-instance vpn1
ipv6 route-static FC00:2:: 64 vpn-instance vpn1
ipv6 route-static FC00:3:: 64 vpn-instance vpn1
ipv6 route-static vpn-instance vpn1 :: 0 public
#
return
```

- Leaf1的配置脚本

```
#
sysname Leaf1
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ipv6 2001:DB8:1:6301::1 peer 2001:db8:1:7301::1
 priority 150
#
stp mode rstp
stp v-stp enable
```

```
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 6.6.6.6:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 6.6.6.6:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
VLAN 100
 m-lag peer-link reserved
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 6.6.6.6:10
  vpn-target 0:10 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:10 import-extcommunity
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 6.6.6.6:20
  vpn-target 0:20 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:20 import-extcommunity
#
ospfv3 1
 router-id 6.6.6.6
 area 0.0.0.0
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 mac-address 0000-5e00-0105
 ipv6 nd collect host enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.2.1 255.255.255.0
 ipv6 address FC00:2::1/64
 mac-address 0000-5e00-0106
 ipv6 nd collect host enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ipv6 enable
 ipv6 address 2001:db8:2::1/64
 ospfv3 1 area 0.0.0.0
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk2
 stp edged-port enable
```

```
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2.10 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface Eth-Trunk3
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk3.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D301::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D305::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/3
 eth-trunk 1
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 2
#
interface 100GE1/0/6
 eth-trunk 3
#
interface LoopBack0
 ip address 6.6.6.6 255.255.255.255
#
interface LoopBack1
 ipv6 enable
 ipv6 address 2001:DB8:1:2301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack2
 ipv6 enable
 ip address 6.6.6.6 255.255.255.255
 ipv6 address 2001:DB8:1:6301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack3
 ipv6 enable
 ipv6 address 2001:db8:1:6302::1/128
#
interface Nve1
 source 2001:DB8:1:2301::1
 pip-source 2001:db8:1:6302::1 peer 2001:db8:1:7302::1 bypass
 vni 10 head-end peer-list protocol bgp
 vni 20 head-end peer-list protocol bgp
 mac-address 0000-5e00-0102
#
monitor-link group 1
 port 100GE1/0/1 uplink
```

```
 port 100GE1/0/2 uplink
 port Eth-Trunk2 downlink 1
 port Eth-Trunk3 downlink 2
#
bgp 100
 peer 2001:DB8:1:4301::1 as-number 100
 peer 2001:DB8:1:4301::1 connect-interface LoopBack2
 peer 2001:DB8:1:5301::1 as-number 100
 peer 2001:DB8:1:5301::1 connect-interface LoopBack2
 #
 ipv4-family unicast
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  policy vpn-target
  peer 2001:DB8:1:4301::1 enable
  peer 2001:DB8:1:4301::1 advertise irb
  peer 2001:DB8:1:4301::1 advertise irbv6
  peer 2001:DB8:1:5301::1 enable
  peer 2001:DB8:1:5301::1 advertise irb
  peer 2001:DB8:1:5301::1 advertise irbv6
#
ipv6 route-static 2001:db8:1:7302::1 128 2001:db8:2::2 preference 1
#
return
```

- Leaf2的配置脚本

```
#
sysname Leaf2
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ipv6 2001:DB8:1:7301::1 peer 2001:db8:1:6301::1
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 7.7.7.7:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 7.7.7.7:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
VLAN 100
 m-lag peer-link reserved
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 7.7.7.7:10
  vpn-target 0:10 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:10 import-extcommunity
#
bridge-domain 20
```

```
 vxlan vni 20
 evpn
  route-distinguisher 7.7.7.7:20
  vpn-target 0:20 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:20 import-extcommunity
#
ospfv3 1
 router-id 7.7.7.7
 area 0.0.0.0
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 mac-address 0000-5e00-0105
 ipv6 nd collect host enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.2.1 255.255.255.0
 ipv6 address FC00:2::1/64
 mac-address 0000-5e00-0106
 ipv6 nd collect host enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ipv6 enable
 ipv6 address 2001:db8:2::2/64
 ospfv3 1 area 0.0.0.0
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk2
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2.10 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface Eth-Trunk3
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk3.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D302::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
```



```
 ipv6 address 2001:DB8:1:D306::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/3
 eth-trunk 1
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 2
#
interface 100GE1/0/6
 eth-trunk 3
#
interface LoopBack0
 ip address 7.7.7.7 255.255.255.255
#
interface LoopBack1
 ipv6 enable
 ipv6 address 2001:DB8:1:2301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack2
 ipv6 enable
 ipv6 address 2001:DB8:1:7301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack3
 ipv6 enable
 ipv6 address 2001:db8:1:7302::1/128
#
interface Nve1
 source 2001:DB8:1:2301::1
 pip-source 2001:db8:1:7302::1 peer 2001:db8:1:6302::1 bypass
 vni 10 head-end peer-list protocol bgp
 vni 20 head-end peer-list protocol bgp
 mac-address 0000-5e00-0102
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk2 downlink 1
 port Eth-Trunk3 downlink 2
#
bgp 100
 peer 2001:DB8:1:4301::1 as-number 100
 peer 2001:DB8:1:4301::1 connect-interface LoopBack2
 peer 2001:DB8:1:5301::1 as-number 100
 peer 2001:DB8:1:5301::1 connect-interface LoopBack2
 #
 ipv4-family unicast
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  policy vpn-target
  peer 2001:DB8:1:4301::1 enable
  peer 2001:DB8:1:4301::1 advertise irb
  peer 2001:DB8:1:4301::1 advertise irbv6
  peer 2001:DB8:1:5301::1 enable
  peer 2001:DB8:1:5301::1 advertise irb
```



```
 peer 2001:DB8:1:5301::1 advertise irbv6
#
ipv6 route-static 2001:db8:1:6302::1 128 2001:db8:2::1 preference 1
#
return
```

- Leaf3的配置脚本

```
#
sysname Leaf3
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ipv6 2001:DB8:1:8301::1 peer 2001:db8:1:9301::1
 priority 150
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 8.8.8.8:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 8.8.8.8:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
VLAN 100
 m-lag peer-link reserved
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 8.8.8.8:20
  vpn-target 0:20 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:20 import-extcommunity
#
bridge-domain 30
 vxlan vni 30
 evpn
  route-distinguisher 8.8.8.8:30
  vpn-target 0:30 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:30 import-extcommunity
#
ospfv3 1
 router-id 8.8.8.8
 area 0.0.0.0
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.2.1 255.255.255.0
 ipv6 address FC00:2::1/64
 mac-address 0000-5e00-0106
 ipv6 nd collect host enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.3.1 255.255.255.0
 ipv6 address FC00:3::1/64
```

```
 mac-address 0000-5e00-0107
 ipv6 nd collect host enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ipv6 enable
 ipv6 address 2001:db8:3::1/64
 ospfv3 1 area 0.0.0.0
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk2
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface Eth-Trunk3
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk3.30 mode l2
 encapsulation dot1q vid 30
 bridge-domain 30
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D303::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D307::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/3
 eth-trunk 1
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 2
#
interface 100GE1/0/6
 eth-trunk 3
#
interface LoopBack0
 ip address 8.8.8.8 255.255.255.255
#
interface LoopBack1
 ipv6 enable
 ipv6 address 2001:DB8:1:3301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack2
 ipv6 enable
```

```
 ipv6 address 2001:DB8:1:8301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack3
 ipv6 enable
 ipv6 address 2001:db8:1:8302::1/128
#
interface Nve1
 source 2001:DB8:1:3301::1
 pip-source 2001:db8:1:8302::1 peer 2001:db8:1:9302::1 bypass
 vni 20 head-end peer-list protocol bgp
 vni 30 head-end peer-list protocol bgp
 mac-address 0000-5e00-0103
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk2 downlink 1
 port Eth-Trunk3 downlink 2
#
bgp 100
 peer 2001:DB8:1:4301::1 as-number 100
 peer 2001:DB8:1:4301::1 connect-interface LoopBack2
 peer 2001:DB8:1:5301::1 as-number 100
 peer 2001:DB8:1:5301::1 connect-interface LoopBack2
 #
 ipv4-family unicast
 #
 ipv4-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 ipv6-family vpn-instance vpn1
  import-route direct
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  policy vpn-target
  peer 2001:DB8:1:4301::1 enable
  peer 2001:DB8:1:4301::1 advertise irb
  peer 2001:DB8:1:4301::1 advertise irbv6
  peer 2001:DB8:1:5301::1 enable
  peer 2001:DB8:1:5301::1 advertise irb
  peer 2001:DB8:1:5301::1 advertise irbv6
#
ipv6 route-static 2001:db8:1:9302::1 128 2001:db8:3::2 preference 1
#
return
```

- Leaf4的配置脚本

```
#
sysname Leaf4
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
 dual-active detection source ipv6 2001:DB8:1:9301::1 peer 2001:db8:1:8301::1
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 9.9.9.9:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 9.9.9.9:1
```

```
 vpn-target 0:1 export-extcommunity evpn
 vpn-target 0:1 import-extcommunity evpn
 vxlan vni 5000
#
VLAN 100
 m-lag peer-link reserved
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 9.9.9.9:20
  vpn-target 0:20 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:20 import-extcommunity
#
bridge-domain 30
 vxlan vni 30
 evpn
  route-distinguisher 9.9.9.9:30
  vpn-target 0:30 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:30 import-extcommunity
#
ospfv3 1
 router-id 9.9.9.9
 area 0.0.0.0
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.2.1 255.255.255.0
 ipv6 address FC00:2::1/64
 mac-address 0000-5e00-0106
 ipv6 nd collect host enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.3.1 255.255.255.0
 ipv6 address FC00:3::1/64
 mac-address 0000-5e00-0107
 ipv6 nd collect host enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ipv6 enable
 ipv6 address 2001:db8:3::2/64
 ospfv3 1 area 0.0.0.0
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk2
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface Eth-Trunk3
 stp edged-port enable
 mode lacp-static
```

```
 dfs-group 1 m-lag 2
#
interface Eth-Trunk3.30 mode l2
 encapsulation dot1q vid 30
 bridge-domain 30
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D304::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1:D308::2/64
 ospfv3 1 area 0.0.0.0
 ospfv3 network-type p2p
#
interface 100GE1/0/3
 eth-trunk 1
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 2
#
interface 100GE1/0/6
 eth-trunk 3
#
interface LoopBack0
 ip address 9.9.9.9 255.255.255.255
#
interface LoopBack1
 ipv6 enable
 ipv6 address 2001:DB8:1:3301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack2
 ipv6 enable
 ipv6 address 2001:DB8:1:9301::1/128
 ospfv3 1 area 0.0.0.0
#
interface LoopBack3
 ipv6 enable
 ipv6 address 2001:db8:1:9302::1/128
#
interface Nve1
 source 2001:DB8:1:3301::1
 pip-source 2001:db8:1:9302::1 peer 2001:db8:1:8302::1 bypass
 vni 20 head-end peer-list protocol bgp
 vni 30 head-end peer-list protocol bgp
 mac-address 0000-5e00-0103
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk2 downlink 1
 port Eth-Trunk3 downlink 2
#
bgp 100
 peer 2001:DB8:1:4301::1 as-number 100
 peer 2001:DB8:1:4301::1 connect-interface LoopBack2
 peer 2001:DB8:1:5301::1 as-number 100
 peer 2001:DB8:1:5301::1 connect-interface LoopBack2
 #
 ipv4-family unicast
```



```
#
ipv4-family vpn-instance vpn1
 import-route direct
 advertise l2vpn evpn
#
ipv6-family vpn-instance vpn1
 import-route direct
 advertise l2vpn evpn
#
l2vpn-family evpn
 policy vpn-target
 peer 2001:DB8:1:4301::1 enable
 peer 2001:DB8:1:4301::1 advertise irb
 peer 2001:DB8:1:4301::1 advertise irbv6
 peer 2001:DB8:1:5301::1 enable
 peer 2001:DB8:1:5301::1 advertise irb
 peer 2001:DB8:1:5301::1 advertise irbv6
#
ipv6 route-static 2001:db8:1:8302::1 128 2001:db8:3::1 preference 1
#
return
```

1.5 单 DC 分布式网关部署方式的 VXLAN 二层架构举例

适用产品和版本

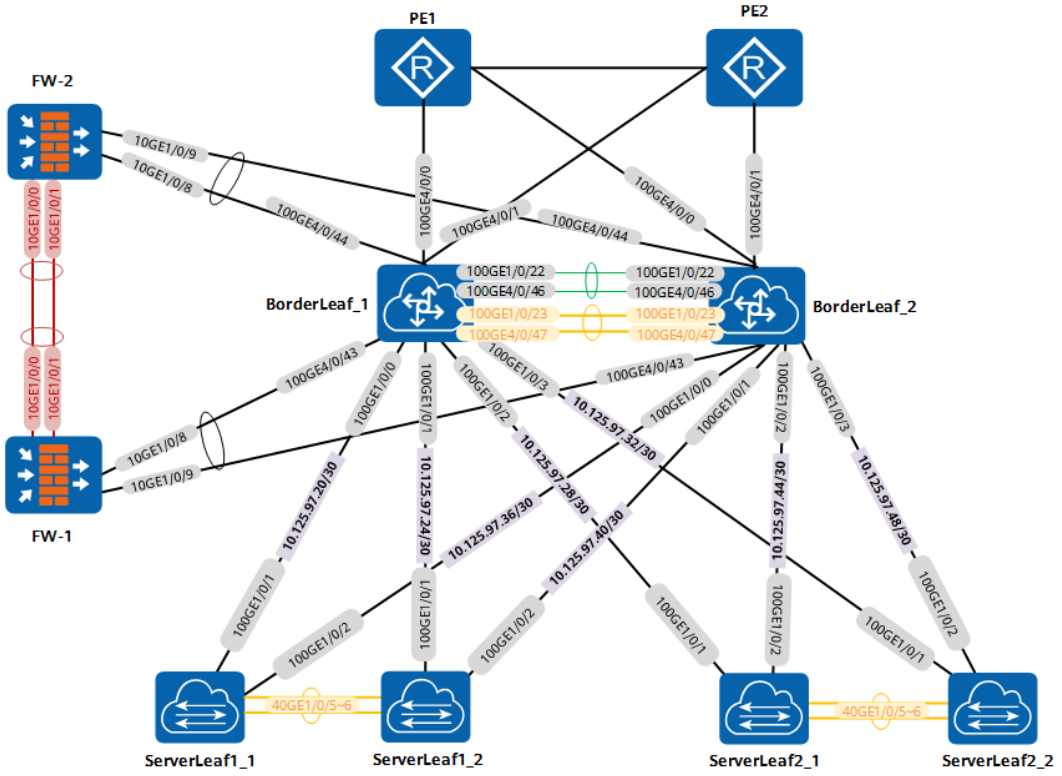
- CE16800（除X系列单板外）、CE8800、CE6800（除CE6820H外）系列产品V300R020C00或更高版本。

组网需求

如图1-5所示，二层架构中Spine、Border Leaf、Service Leaf三者融合部署，Server Leaf-Spine/Border Leaf/Service Leaf在物理拓扑上形成两个层次的架构，故属于“二层架构”。

- Border Leaf层：Border Leaf交换机作为分布式Overlay组网中的出口，南向与Server Leaf之间使用三层路由口互联，形成ECMP IP转发网络；北向与出口路由器PE互联。
- Server Leaf层：Server Leaf交换机部署M-LAG，北向与Spine设备通过三层路由口互联。

图 1-5 单 DC 分布式网关部署方式的 VXLAN 二层架构组网图



规划交换机的两类Loopback地址，建议如下所示。

- Loopback0：专门作为VTEP IP地址。对于双活设备组，组成员的VTEP IP必须保持一致。
- Loopback1：
  - 作为Router-ID地址
  - M-LAG的DFS-Group IP地址
  - 建立BGP EVPN对等体时发送BGP报文的源接口
- Loopback2：作为静态Bypass VXLAN隧道的源端IP地址。

每台交换机的Loopback地址的具体规划如表1-5所示。

表 1-5 数据准备表（Loopback 地址规划）

设备名称	Loopback0	Loopback1	Loopback2
BorderLeaf_1	10.125.99.1/32（虚MAC：00e0-fc00-0101）	10.125.98.1/32	10.135.98.1/32
BorderLeaf_2	10.125.99.1/32（虚MAC：00e0-fc00-0101）	10.125.98.2/32	10.135.98.2/32
ServerLeaf1_1	10.125.99.2/32	10.125.98.3/32	10.135.98.3/32

设备名称	Loopback0	Loopback1	Loopback2
ServerLeaf1_2	10.125.99.2/32	10.125.98.4/32	10.135.98.4/32
ServerLeaf2_1	10.125.99.3/32	10.125.98.5/32	10.135.98.5/32
ServerLeaf2_2	10.125.99.3/32	10.125.98.6/32	10.135.98.6/32

表 1-6 互联地址规划

设备名称	接口编号	IP地址	对接设备及接口编号	说明
BorderLeaf_1	Eth-Trunk20	10.125.97.17/30	BorderLeaf_2：Eth-Trunk20	出口逃生路径
	100GE4/0/0	10.125.97.1/30	PE1	-
	100GE4/0/1	10.125.97.5/30	PE2	-
	100GE1/0/0	10.125.97.21/30	ServerLeaf1_1：100GE1/0/1	-
	100GE1/0/1	10.125.97.25/30	ServerLeaf1_2：100GE1/0/1	-
	100GE1/0/2	10.125.97.29/30	ServerLeaf2_1：100GE1/0/1	-
	100GE1/0/3	10.125.97.33/30	ServerLeaf2_2：100GE1/0/1	-
	100GE4/0/43	-	FW-1	-
	100GE4/0/44	-	FW-2	-
	vlanif11	10.125.97.57/29	-	FW互联的管理链路接口地址
BorderLeaf_2	Eth-Trunk20	10.125.97.18/30	BorderLeaf_1：Eth-Trunk20	出口逃生路径
	100GE4/0/0	10.125.97.9/30	PE1	-
	100GE4/0/1	10.125.97.13/30	PE2	-
	100GE1/0/0	10.125.97.37/30	ServerLeaf1_1：100GE1/0/2	-

设备名称	接口编号	IP地址	对接设备及接口编号	说明
	100GE1/0/1	10.125.97.41/30	ServerLeaf1_2：100GE1/0/2	-
	100GE1/0/2	10.125.97.45/30	ServerLeaf2_1：100GE1/0/2	-
	100GE1/0/3	10.125.97.49/30	ServerLeaf2_2：100GE1/0/2	-
	100GE4/0/43	-	FW-1	-
	100GE4/0/44	-	FW-2	-
	vlanif11	10.125.97.57/29	-	FW互联的管理链路接口地址
ServerLeaf1_1	100GE1/0/1	10.125.97.22/30	BorderLeaf_1：100GE1/0/0	-
	100GE1/0/2	10.125.97.38/30	BorderLeaf_2：100GE1/0/0	-
ServerLeaf1_2	100GE1/0/1	10.125.97.26/30	BorderLeaf_1：100GE1/0/1	-
	100GE1/0/2	10.125.97.42/30	BorderLeaf_2：100GE1/0/1	-
ServerLeaf2_1	100GE1/0/1	10.125.97.30/30	BorderLeaf_1：100GE1/0/2	-
	100GE1/0/2	10.125.97.46/30	BorderLeaf_2：100GE1/0/2	-
ServerLeaf2_2	100GE1/0/1	10.125.97.34/30	BorderLeaf_1：100GE1/0/3	-
	100GE1/0/2	10.125.97.50/30	BorderLeaf_2：100GE1/0/3	-
FW-1	Eth-Trunk0（10GE 1/0/0 to 1/0/1）	10.125.97.73/30	FW2：Eth-Trunk0	防火墙心跳口
	Eth-Trunk11（10GE 1/0/8、1/0/9）	-	-	与BorderLeaf_1、BorderLeaf_2互联端口
	Vlanif3004	10.125.97.242/30	-	vsys_1
FW-2	Eth-Trunk0	10.125.97.74/30	FW1：Eth-Trunk0	防火墙心跳口

设备名称	接口编号	IP地址	对接设备及接口编号	说明
	Eth-Trunk11（10GE 1/0/8、1/0/9）	10.125.97.242/30	-	与BorderLeaf_1、BorderLeaf_2互联端口

表 1-7 外网 IP 地址

访问外网的IP地址
1.2.3.4/24

表 1-8 设备的 RD 值和 RT 值

设备名称	VLAN ID	广播域BD ID	VXLAN网络标识VNI ID	EVPN实例			VPN实例			
				RD值	RT值		VPN实例名称	VXLAN网络标识VNI ID	RD值	RT值
					ERT/IRT	ERT				ERT/IRT(EVPN)
ServerLeaf1_1	10	10	10	10:2	100:10	100:50110	vpn1	5010	20:2	100:5010
ServerLeaf1_2				10:4			vpn2		20:4	
ServerLeaf2_1	20	20	20	10:3	100:20		vpn3		20:3	
ServerLeaf2_2				10:5			vpn4		20:5	

配置思路

配置思路如下：

1. 配置VXLAN优化命令。
2. 配置Underlay网络。
   a. 配置Border Leaf。

i. 配置IP地址：配置与Server Leaf节点三层互联地址，与防火墙互联管理地址；配置Loopback0地址（作为VTEP地址）；配置Loopback1地址（作为Router-ID&dfs-group）；配置NVE接口VTEP IP地址。

ii. 配置M-LAG：配置M-LAG全局模式、DFS组、peer-link，并分别配置与防火墙互联M-LAG接口。

iii. 配置路由：配置OSPF路由，配置OSPF接口的网络类型为P2P，并发布Loopback地址及与防火墙管理地址；配置BGP EVPN作为VXLAN的控制平面。

b. 配置Server Leaf。

配置思路与Border Leaf一样。

c. 配置防火墙。

i. 配置防火墙基础信息。

ii. 关闭备份当前运行配置的功能，在主备防火墙上均需要配置。

iii. 配置防火墙与Border Leaf/Service Leaf互联端口。

iv. 配置两台防火墙之间的心跳接口。

v. 配置两台防火墙的主备镜像模式。

vi. 配置安全域及缺省安全策略。只需要在FW1中进行配置，FW2将自动同步。

vii. 使能防火墙的vsys功能。只需要在FW1中进行配置，FW2将自动同步。

3. 配置Overlay网络。

a. 配置Border Leaf。

b. 配置Server Leaf。

c. 配置防火墙。

操作步骤

步骤1 配置VXLAN优化命令。

在CE设备上进行VXLAN相关配置前，请先根据不同的设备款型，配置VXLAN优化命令、业务环回功能、三层口保留VLAN，以确保业务稳定运行。设备款型不同具体的配置命令行不一样。

配置BorderLeaf_1。其他设备的配置与BorderLeaf_1类似，这里不再赘述。

```
<HUAWEI> system-view
[~HUAWEI] sysname BorderLeaf_1
[*HUAWEI] commit
[*BorderLeaf_1] system resource large-route  //配置系统资源模式为大路由模式。该配置需要重启设备才能生效。仅CE16800（安装E系列单板、EK系列单板）、CE6863H、CE6863H-K、CE6881H、CE6881H-K支持。
[*BorderLeaf_1] vxlan tunnel-status track exact-route  //使能VXLAN隧道目的端精确路由状态订阅功能，优化网络收敛性能。
[*BorderLeaf_1] commit
```

步骤2 配置Underlay网络。

1. 配置Border Leaf。

a. 配置IP地址。

i. 配置Border Leaf与其他设备互联IP地址。

#配置BorderLeaf_1与Server Leaf的互联接口地址：

```
[~BorderLeaf_1] interface 100GE 1/0/0
[~BorderLeaf_1-100GE1/0/0] description to ServerLeaf1_1
```

```
[*BorderLeaf_1-100GE1/0/0] undo portswitch
[*BorderLeaf_1-100GE1/0/0] ip address 10.125.97.21 255.255.255.252
[*BorderLeaf_1-100GE1/0/0] ospf network-type p2p  //配置与Server Leaf互联OSPF接口的网络类型为P2P
[*BorderLeaf_1-100GE1/0/0] quit
[*BorderLeaf_1] interface 100GE 1/0/1
[*BorderLeaf_1-100GE1/0/1] description to ServerLeaf1_2
[*BorderLeaf_1-100GE1/0/1] undo portswitch
[*BorderLeaf_1-100GE1/0/1] ip address 10.125.97.25 255.255.255.252
[*BorderLeaf_1-100GE1/0/1] ospf network-type p2p
[*BorderLeaf_1-100GE1/0/1] quit
[*BorderLeaf_1] interface 100GE 1/0/2
[*BorderLeaf_1-100GE1/0/2] description to ServerLeaf2_1
[*BorderLeaf_1-100GE1/0/2] undo portswitch
[*BorderLeaf_1-100GE1/0/2] ip address 10.125.97.29 255.255.255.252
[*BorderLeaf_1-100GE1/0/2] ospf network-type p2p
[*BorderLeaf_1-100GE1/0/2] quit
[*BorderLeaf_1] interface 100GE 1/0/3
[*BorderLeaf_1-100GE1/0/3] description to ServerLeaf2_2
[*BorderLeaf_1-100GE1/0/3] undo portswitch
[*BorderLeaf_1-100GE1/0/3] ip address 10.125.97.33 255.255.255.252
[*BorderLeaf_1-100GE1/0/3] ospf network-type p2p
[*BorderLeaf_1-100GE1/0/3] quit
[*BorderLeaf_1-100GE1/0/3] commit
```

#配置BorderLeaf_2与Server Leaf的互联接口地址：

```
[~BorderLeaf_2] interface 100GE 1/0/0
[~BorderLeaf_2-100GE1/0/0] description to ServerLeaf1_1
[*BorderLeaf_2-100GE1/0/0] undo portswitch
[*BorderLeaf_2-100GE1/0/0] ip address 10.125.97.37 255.255.255.252
[*BorderLeaf_2-100GE1/0/0] ospf network-type p2p
[*BorderLeaf_2-100GE1/0/0] quit
[*BorderLeaf_2] interface 100GE 1/0/1
[*BorderLeaf_2-100GE1/0/1] description to ServerLeaf1_2
[*BorderLeaf_2-100GE1/0/1] undo portswitch
[*BorderLeaf_2-100GE1/0/1] ip address 10.125.97.41 255.255.255.252
[*BorderLeaf_2-100GE1/0/1] ospf network-type p2p
[*BorderLeaf_2-100GE1/0/1] quit
[*BorderLeaf_2] interface 100GE 1/0/2
[*BorderLeaf_2-100GE1/0/2] description to ServerLeaf2_1
[*BorderLeaf_2-100GE1/0/2] undo portswitch
[*BorderLeaf_2-100GE1/0/2] ip address 10.125.97.45 255.255.255.252
[*BorderLeaf_2-100GE1/0/2] ospf network-type p2p
[*BorderLeaf_2-100GE1/0/2] quit
[*BorderLeaf_2] interface 100GE 1/0/3
[*BorderLeaf_2-100GE1/0/3] description to ServerLeaf2_2
[*BorderLeaf_2-100GE1/0/3] undo portswitch
[*BorderLeaf_2-100GE1/0/3] ip address 10.125.97.49 255.255.255.252
[*BorderLeaf_2-100GE1/0/3] ospf network-type p2p
[*BorderLeaf_2-100GE1/0/3] quit
[*BorderLeaf_2-100GE1/0/3] commit
```

#配置BorderLeaf_1与PE的互联接口地址：

```
[~BorderLeaf_1] interface 100GE 4/0/0
[*BorderLeaf_1-100GE4/0/0] description to PE1
[*BorderLeaf_1-100GE4/0/0] undo portswitch
[*BorderLeaf_1-100GE4/0/0] ip address 10.125.97.1 255.255.255.252
[*BorderLeaf_1-100GE4/0/0] quit
[*BorderLeaf_1] interface 100GE 4/0/1
[*BorderLeaf_1-100GE4/0/1] description to PE2
[*BorderLeaf_1-100GE4/0/1] undo portswitch
[*BorderLeaf_1-100GE4/0/1] ip address 10.125.97.5 255.255.255.252
[*BorderLeaf_1-100GE4/0/1] quit
[*BorderLeaf_1-100GE4/0/1] commit
```

#配置BorderLeaf_2与PE的互联接口地址：

```
[~BorderLeaf_2] interface 100GE 4/0/0
[*BorderLeaf_2-100GE4/0/0] description to PE1
[*BorderLeaf_2-100GE4/0/0] undo portswitch
[*BorderLeaf_2-100GE4/0/0] ip address 10.125.97.9 255.255.255.252
```

```
[*BorderLeaf_2-100GE4/0/0] quit
[*BorderLeaf_2] interface 100GE 4/0/1
[*BorderLeaf_2-100GE4/0/1] description to PE2
[*BorderLeaf_2-100GE4/0/1] undo portswitch
[*BorderLeaf_2-100GE4/0/1] ip address 10.125.97.13 255.255.255.252
[*BorderLeaf_2-100GE4/0/1] quit
[*BorderLeaf_2-100GE4/0/1] commit
```

ii. 配置Border Leaf的Loopback接口地址。

#配置BorderLeaf_1的Loopback接口地址：

```
[~BorderLeaf_1] interface LoopBack 0
[*BorderLeaf_1-LoopBack0] description VTEP
[*BorderLeaf_1-LoopBack0] ipv6 enable    //当需要使用IPv6时，配置使能IPv6
[*BorderLeaf_1-LoopBack0] ip address 10.125.99.1 255.255.255.255
[*BorderLeaf_1-LoopBack0] quit
[*BorderLeaf_1] interface LoopBack 1
[*BorderLeaf_1-LoopBack1] description DFS-GROUP/ROUTER-ID
[*BorderLeaf_1-LoopBack1] ip address 10.125.98.1 255.255.255.255
[*BorderLeaf_1-LoopBack1] quit
[*BorderLeaf_1] interface LoopBack 2
[*BorderLeaf_1-LoopBack2] description Bypass VXLAN
[*BorderLeaf_1-LoopBack2] ip address 10.135.98.1 255.255.255.255
[*BorderLeaf_1-LoopBack2] quit
[*BorderLeaf_1] commit
```

#配置BorderLeaf_2的Loopback接口地址：

```
[~BorderLeaf_2] interface LoopBack 0
[*BorderLeaf_2-LoopBack0] description VTEP
[*BorderLeaf_2-LoopBack0] ipv6 enable    //当需要使用IPv6时，配置使能IPv6
[*BorderLeaf_2-LoopBack0] ip address 10.125.99.1 255.255.255.255
[*BorderLeaf_2-LoopBack0] quit
[*BorderLeaf_2] interface LoopBack 1
[*BorderLeaf_2-LoopBack1] description DFS-GROUP/ROUTER-ID
[*BorderLeaf_2-LoopBack1] ip address 10.125.98.2 255.255.255.255
[*BorderLeaf_2-LoopBack1] quit
[*BorderLeaf_2] interface LoopBack 2
[*BorderLeaf_2-LoopBack2] description Bypass VXLAN
[*BorderLeaf_2-LoopBack2] ip address 10.135.98.2 255.255.255.255
[*BorderLeaf_2-LoopBack2] quit
[*BorderLeaf_2] commit
```

iii. 配置NVE接口VTEP IP和虚拟MAC地址。

#配置BorderLeaf_1的NVE接口：

```
[~BorderLeaf_1] vlan 100  //本VLAN不能划分给其他业务使用，本例中以100举例
[*BorderLeaf_1-vlan100] m-lag peer-link reserved  //仅允许peer-link加入到该VLAN
[*BorderLeaf_1-vlan100] quit
[*BorderLeaf_1] interface vlanif 100
[*BorderLeaf_1-Vlanif100] reserved for vxlan bypass  //指定peer-link接口上VLANIF的IPv4地址只给Bypass VXLAN隧道使用
[*BorderLeaf_1-Vlanif100] ip address 10.125.96.1 30  //配置静态Bypass VXLAN隧道的源端IPv4地址
[*BorderLeaf_1-Vlanif100] quit
[*BorderLeaf_1] ip route-static 10.135.98.2 32 10.125.96.2 preference 1  //配置静态路由，打通Bypass VXLAN隧道
[~BorderLeaf_1] interface nve 1
[*BorderLeaf_1-Nve1] source 10.125.99.1
[*BorderLeaf_1-Nve1] mac-address 00e0-fc00-0101
[*BorderLeaf_1-Nve1] pip-source 10.135.98.1 peer 10.135.98.2 bypass  //创建静态Bypass VXLAN隧道，指定源端地址和对端地址
[*BorderLeaf_1-Nve1] commit
```

#配置BorderLeaf_2的NVE接口：

```
[~BorderLeaf_2] vlan 100
[*BorderLeaf_2-vlan100] m-lag peer-link reserved
[*BorderLeaf_2-vlan100] quit
[*BorderLeaf_2] interface vlanif 100
[*BorderLeaf_2-Vlanif100] reserved for vxlan bypass
[*BorderLeaf_2-Vlanif100] ip address 10.125.96.2 30
```

```
[*BorderLeaf_2-Vlanif100] quit
[*BorderLeaf_1] ip route-static 10.135.98.1 32 10.125.96.1 preference 1
[~BorderLeaf_2] interface nve 1
[*BorderLeaf_2-Nve1] source 10.125.99.1
[*BorderLeaf_2-Nve1] mac-address 00e0-fc00-0101
[*BorderLeaf_2-Nve1] pip-source 10.135.98.2 peer 10.135.98.1
[*BorderLeaf_2-Nve1] commit
```

b. 配置M-LAG。

i. 配置M-LAG模式。

#配置BorderLeaf_1的M-LAG模式：

```
[~BorderLeaf_1] stp mode rstp
[*BorderLeaf_1] stp v-stp enable  //配置V-STP方式的M-LAG
[*BorderLeaf_1] commit
```

#配置BorderLeaf_2的M-LAG模式：

```
[~BorderLeaf_2] stp mode rstp
[*BorderLeaf_2] stp v-stp enable  //配置V-STP方式的M-LAG
[*BorderLeaf_2] commit
```

ii. 配置M-LAG的DFS组。

#配置BorderLeaf_1的DFS组：

```
[~BorderLeaf_1] dfs-group 1
[*BorderLeaf_1-dfs-group-1] priority 150  //配置DFS优先级高于对端，默认是100
[*BorderLeaf_1-dfs-group-1] authentication-mode hmac-sha256 password
YsHsjx_202206
[*BorderLeaf_1-dfs-group-1] dual-actice detection source ip 10.125.98.1
[*BorderLeaf_1-dfs-group-1] consistency-check enable mode loose  //使能M-LAG配置一
致性检查，模式为松散模式
[*BorderLeaf_1-dfs-group-1] quit
[*BorderLeaf_1-dfs-group-1] commit
```

#配置BorderLeaf_2的DFS组：

```
[~BorderLeaf_2] dfs-group 1
[*BorderLeaf_2-dfs-group-1] authentication-mode hmac-sha256 password
YsHsjx_202206
[*BorderLeaf_2-dfs-group-1] dual-actice detection source ip 10.125.98.2
[*BorderLeaf_2-dfs-group-1] consistency-check enable mode loose
[*BorderLeaf_2-dfs-group-1] quit
[*BorderLeaf_2-dfs-group-1] commit
```

iii. 配置peer-link。

#配置BorderLeaf_1的peer-link：

```
[~BorderLeaf_1] interface Eth-Trunk 0
[*BorderLeaf_1-Eth-Trunk0] trunkport 100GE 4/0/47
[*BorderLeaf_1-Eth-Trunk0] trunkport 100GE 1/0/23
[*BorderLeaf_1-Eth-Trunk0] mode lacp-static
[*BorderLeaf_1-Eth-Trunk0] peer-link 1
[*BorderLeaf_1-Eth-Trunk0] commit
```

#配置BorderLeaf_2的peer-link：

```
[~BorderLeaf_2] interface Eth-Trunk 0
[*BorderLeaf_2-Eth-Trunk0] trunkport 100GE 4/0/47
[*BorderLeaf_2-Eth-Trunk0] trunkport 100GE 1/0/23
[*BorderLeaf_2-Eth-Trunk0] mode lacp-static
[*BorderLeaf_2-Eth-Trunk0] peer-link 1
[*BorderLeaf_2-Eth-Trunk0] commit
```

iv. 配置M-LAG接口。

配置与防火墙互联的业务链路。

#配置BorderLeaf_1与防火墙互联：

```
[~BorderLeaf_1] interface Eth-Trunk 11  //配置与FW主设备（FW-1）互联业务口
[*BorderLeaf_1-Eth-Trunk11] trunkport 100GE 4/0/43
[*BorderLeaf_1-Eth-Trunk11] port link-type trunk
```



```
[*BorderLeaf_1-Eth-Trunk11] undo port trunk allow-pass vlan 1
[*BorderLeaf_1-Eth-Trunk11] stp edged-port enable
[*BorderLeaf_1-Eth-Trunk11] mode lacp-static
[*BorderLeaf_1-Eth-Trunk11] dfs-group 1 m-lag 3
[*BorderLeaf_1-Eth-Trunk11] quit
[*BorderLeaf_1] interface Eth-Trunk12   //配置与FW备设备（FW-2）互联业务口
[*BorderLeaf_1-Eth-Trunk12] trunkport 100GE 4/0/44
[*BorderLeaf_1-Eth-Trunk12] port link-type trunk
[*BorderLeaf_1-Eth-Trunk12] undo port trunk allow-pass vlan 1
[*BorderLeaf_1-Eth-Trunk12] stp edged-port enable
[*BorderLeaf_1-Eth-Trunk12] mode lacp-static
[*BorderLeaf_1-Eth-Trunk12] dfs-group 1 m-lag 4
[*BorderLeaf_1-Eth-Trunk12] quit
[*BorderLeaf_1] commit
```

#配置BorderLeaf_2与防火墙互联：

```
[~BorderLeaf_2] interface Eth-Trunk 11   //配置与FW主设备（FW-1）互联业务口
[*BorderLeaf_2-Eth-Trunk11] trunkport 100GE 4/0/43
[*BorderLeaf_2-Eth-Trunk11] port link-type trunk
[*BorderLeaf_2-Eth-Trunk11] undo port trunk allow-pass vlan 1
[*BorderLeaf_2-Eth-Trunk11] stp edged-port enable
[*BorderLeaf_2-Eth-Trunk11] mode lacp-static
[*BorderLeaf_2-Eth-Trunk11] dfs-group 1 m-lag 3
[*BorderLeaf_2-Eth-Trunk11] quit
[*BorderLeaf_2] interface Eth-Trunk12   //配置与FW备设备（FW-2）互联业务口
[*BorderLeaf_2-Eth-Trunk12] trunkport 100GE 4/0/44
[*BorderLeaf_2-Eth-Trunk12] port link-type trunk
[*BorderLeaf_2-Eth-Trunk12] undo port trunk allow-pass vlan 1
[*BorderLeaf_2-Eth-Trunk12] stp edged-port enable
[*BorderLeaf_2-Eth-Trunk12] mode lacp-static
[*BorderLeaf_2-Eth-Trunk12] dfs-group 1 m-lag 4
[*BorderLeaf_2-Eth-Trunk12] quit
[*BorderLeaf_2-Eth-Trunk12] commit
```

c. 配置路由。

i. 配置OSPF路由打通VXLAN Underlay路由。

#配置BorderLeaf_1的OSPF路由：

```
[~BorderLeaf_1] bfd    //全局使能BFD功能
[*BorderLeaf_1-bfd] quit
[*BorderLeaf_1] ospf
[*BorderLeaf_1] ospf 1 router-id 10.125.98.1
[*BorderLeaf_1-ospf-1] bfd all-interfaces enable
[*BorderLeaf_1-ospf-1] bfd all-interfaces min-tx-interval 500 min-rx-interval 500
detect-multiplier 3
[*BorderLeaf_1-ospf-1] lsa-arrival-interval intelligent-timer 50 50 50     //设置OSPF LSA
接收的时间间隔，优化收敛时间
[*BorderLeaf_1-ospf-1] area 0.0.0.0
[*BorderLeaf_1-ospf-1-area-0.0.0.0] network 10.125.97.20 0.0.0.3      //分别建立与4台
Server Leaf设备的路由邻居
[*BorderLeaf_1-ospf-1-area-0.0.0.0] network 10.125.97.24 0.0.0.3
[*BorderLeaf_1-ospf-1-area-0.0.0.0] network 10.125.97.28 0.0.0.3
[*BorderLeaf_1-ospf-1-area-0.0.0.0] network 10.125.97.32 0.0.0.3
[*BorderLeaf_1-ospf-1-area-0.0.0.0] network 10.125.98.1 0.0.0.0      //发布Loopback地址
[*BorderLeaf_1-ospf-1-area-0.0.0.0] network 10.125.99.1 0.0.0.0
[*BorderLeaf_1-ospf-1-area-0.0.0.0] quit
[*BorderLeaf_1-ospf-1] quit
[*BorderLeaf_1-ospf-1] commit
```

#配置BorderLeaf_2的OSPF路由：

```
[~BorderLeaf_2] bfd    //全局使能BFD功能
[*BorderLeaf_2-bfd] quit
[*BorderLeaf_2] ospf
[*BorderLeaf_2] ospf 1 router-id 10.125.98.2
[*BorderLeaf_2-ospf-1] bfd all-interfaces enable
[*BorderLeaf_2-ospf-1] bfd all-interfaces min-tx-interval 500 min-rx-interval 500
detect-multiplier 3   //仅组网中全部为支持硬件BFD的款型时，配置500ms*3；其余保持默认
配置1000ms*3
[*BorderLeaf_2-ospf-1] lsa-arrival-interval intelligent-timer 50 50 50     //设置OSPF LSA
```

```
接收的时间间隔，优化收敛时间
[*BorderLeaf_2-ospf-1] area 0.0.0.0
[*BorderLeaf_2-ospf-1-area-0.0.0.0] network 10.125.97.36 0.0.0.3      //分别建立与4台Server Leaf设备的路由邻居
[*BorderLeaf_2-ospf-1-area-0.0.0.0] network 10.125.97.40 0.0.0.3
[*BorderLeaf_2-ospf-1-area-0.0.0.0] network 10.125.97.44 0.0.0.3
[*BorderLeaf_2-ospf-1-area-0.0.0.0] network 10.125.97.48 0.0.0.3
[*BorderLeaf_2-ospf-1-area-0.0.0.0] network 10.125.98.2 0.0.0.0      //发布Loopback地址
[*BorderLeaf_2-ospf-1-area-0.0.0.0] network 10.125.99.1 0.0.0.0
[*BorderLeaf_2-ospf-1-area-0.0.0.0] quit
[*BorderLeaf_2-ospf-1] quit
[*BorderLeaf_2-ospf-1] commit
```

ii. 配置OSPF网络故障收敛性能优化。

#配置BorderLeaf_1的OSPF网络故障收敛性能优化：

```
[~BorderLeaf_1] interface 100GE 1/0/0
[*BorderLeaf_1-100GE1/0/0] ospf peer hold-max-cost timer 300000   //所有Border Leaf和Server Leaf配置OSPF邻居建立后在本地设备的LSA中保持最大开销值的时间300s，源于240s的M-LAG延迟UP时间（同时overlay路由收敛）+ 60s的设备表项同步时间
[*BorderLeaf-1-100GE1/0/0] quit
[*BorderLeaf_1] interface 100GE 1/0/1
[*BorderLeaf_1-100GE1/0/1] ospf peer hold-max-cost timer 300000
[*BorderLeaf-1-100GE1/0/1] quit
[*BorderLeaf_1] interface 100GE 1/0/2
[*BorderLeaf_1-100GE1/0/2] ospf peer hold-max-cost timer 300000
[*BorderLeaf-1-100GE1/0/2] quit
[*BorderLeaf_1] interface 100GE 1/0/3
[*BorderLeaf_1-100GE1/0/3] ospf peer hold-max-cost timer 300000
[*BorderLeaf-1-100GE1/0/3] quit
[*BorderLeaf-1-100GE1/0/3] commit
```

#配置BorderLeaf_2的OSPF网络故障收敛性能优化，配置过程及数据与BorderLeaf_1一致，不再赘述。

iii. 配置BGP EVPN。

#配置BorderLeaf_1：

```
[~BorderLeaf_1] evpn-overlay enable     //使能EVPN作为VXLAN的控制平面
[*BorderLeaf_1] bgp 100
[*BorderLeaf_1-bgp] router-id 10.125.98.1
[*BorderLeaf_1-bgp] advertise lowest-priority all-address-family peer-up delay 360  //在邻居状态由Down到Up时将BGP路由的优先级调整为最低优先级；路由延时发布，解决回切场景丢包时间长问题
[*BorderLeaf_1-bgp] undo default ipv4-unicast       //关闭BGP IPv4单播邻居，降低设备负荷
[*BorderLeaf_1-bgp] group ServerLeaf internal       //配置Server Leaf的对等体组并加入相应对等体。
[*BorderLeaf_1-bgp] peer 10.125.98.3 group ServerLeaf
[*BorderLeaf_1-bgp] peer 10.125.98.4 group ServerLeaf
[*BorderLeaf_1-bgp] peer 10.125.98.5 group ServerLeaf
[*BorderLeaf_1-bgp] peer 10.125.98.6 group ServerLeaf
[*BorderLeaf_1-bgp] peer ServerLeaf connect-interface LoopBack1    //指定发送BGP报文的源接口
[*BorderLeaf_1-bgp] l2vpn-family evpn          //使能并进入BGP-EVPN地址族视图
[*BorderLeaf_1-bgp-af-evpn] undo policy vpn-target       //配置去使能对接收到的EVPN路由使能VPN-Target过滤功能
[*BorderLeaf_1-bgp-af-evpn] peer ServerLeaf enable
[*BorderLeaf_1-bgp-af-evpn] peer 10.125.98.3 group ServerLeaf
[*BorderLeaf_1-bgp-af-evpn] peer 10.125.98.4 group ServerLeaf
[*BorderLeaf_1-bgp-af-evpn] peer 10.125.98.5 group ServerLeaf
[*BorderLeaf_1-bgp-af-evpn] peer 10.125.98.6 group ServerLeaf
[*BorderLeaf_1-bgp-af-evpn] peer ServerLeaf advertise irb   //配置向BGP EVPN对等体组Server Leaf发布irb和irbv6路由
[*BorderLeaf_1-bgp-af-evpn] peer ServerLeaf advertise irbv6
[*BorderLeaf_1-bgp-af-evpn] peer ServerLeaf reflect-client   //配置路由反射器功能
[*BorderLeaf_1-bgp-af-evpn] quit
[*BorderLeaf_1-bgp] quit
[*BorderLeaf_1-bgp] commit
```

#配置BorderLeaf_2：

```
[~BorderLeaf_2] evpn-overlay enable
[*BorderLeaf_2] bgp 100
[*BorderLeaf_2-bgp] router-id 10.125.98.2
[*BorderLeaf_2-bgp] advertise lowest-priority all-address-family peer-up delay 360  //在邻居状态由Down到Up时将BGP路由的优先级调整为最低优先级；路由延时发布，解决回切场景丢包时间长问题
[*BorderLeaf_2-bgp] undo default ipv4-unicast      //关闭BGP IPv4单播邻居，降低设备负荷
[*BorderLeaf_2-bgp] group ServerLeaf internal      //配置Server Leaf的对等体组并加入相应对等体。
[*BorderLeaf_2-bgp] peer 10.125.98.3 group ServerLeaf
[*BorderLeaf_2-bgp] peer 10.125.98.4 group ServerLeaf
[*BorderLeaf_2-bgp] peer 10.125.98.5 group ServerLeaf
[*BorderLeaf_2-bgp] peer 10.125.98.6 group ServerLeaf
[*BorderLeaf_2-bgp] peer ServerLeaf connect-interface LoopBack1    //指定发送BGP报文的源接口
[*BorderLeaf_2-bgp] l2vpn-family evpn          //使能并进入BGP-EVPN地址族视图
[*BorderLeaf_2-bgp-af-evpn] undo policy vpn-target      //配置去使能对接收到的EVPN路由使能VPN-Target过滤功能
[*BorderLeaf_2-bgp-af-evpn] peer ServerLeaf enable
[*BorderLeaf_2-bgp-af-evpn] peer 10.125.98.3 group ServerLeaf
[*BorderLeaf_2-bgp-af-evpn] peer 10.125.98.4 group ServerLeaf
[*BorderLeaf_2-bgp-af-evpn] peer 10.125.98.5 group ServerLeaf
[*BorderLeaf_2-bgp-af-evpn] peer 10.125.98.6 group ServerLeaf
[*BorderLeaf_2-bgp-af-evpn] peer ServerLeaf advertise irb
[*BorderLeaf_2-bgp-af-evpn] peer ServerLeaf advertise irbv6
[*BorderLeaf_2-bgp-af-evpn] peer ServerLeaf reflect-client
[*BorderLeaf_2-bgp-af-evpn] quit
[*BorderLeaf_2-bgp] quit
[*BorderLeaf_2-bgp] commit
```

2. 配置接入Server Leaf组。
   a. 配置IP地址。
      i. 配置Server Leaf与Border Leaf互联IP地址。

      #配置ServerLeaf1_1与Border Leaf的互联接口地址：

```
[~ServerLeaf1_1] interface 100GE 1/0/1
[*ServerLeaf1_1-100GE1/0/1] description to BorderLeaf_1
[*ServerLeaf1_1-100GE1/0/1] undo portswitch
[*ServerLeaf1_1-100GE1/0/1] ip address 10.125.97.22 255.255.255.252
[*ServerLeaf1_1-100GE1/0/1] ospf network-type p2p  //配置与Border Leaf互联OSPF接口的网络类型为P2P
[*ServerLeaf1_1-100GE1/0/1] quit
[*ServerLeaf1_1] interface 100GE 1/0/2
[*ServerLeaf1_1-100GE1/0/2] description to BorderLeaf_2
[*ServerLeaf1_1-100GE1/0/2] undo portswitch
[*ServerLeaf1_1-100GE1/0/2] ip address 10.125.97.38 255.255.255.252
[*ServerLeaf1_1-100GE1/0/2] ospf network-type p2p
[*ServerLeaf1_1-100GE1/0/2] quit
[*ServerLeaf1_1-100GE1/0/2] commit
```

      #配置ServerLeaf1_2与Border Leaf的互联接口地址：

```
[~ServerLeaf1_2] interface 100GE 1/0/1
[*ServerLeaf1_2-100GE1/0/1] description to BorderLeaf_1
[*ServerLeaf1_2-100GE1/0/1] undo portswitch
[*ServerLeaf1_2-100GE1/0/1] ip address 10.125.97.26 255.255.255.252
[*ServerLeaf1_2-100GE1/0/1] ospf network-type p2p
[*ServerLeaf1_2-100GE1/0/1] quit
[*ServerLeaf1_2] interface 100GE 1/0/2
[*ServerLeaf1_2-100GE1/0/2] description to BorderLeaf_2
[*ServerLeaf1_2-100GE1/0/2] undo portswitch
[*ServerLeaf1_2-100GE1/0/2] ip address 10.125.97.42 255.255.255.252
[*ServerLeaf1_2-100GE1/0/2] ospf network-type p2p
[*ServerLeaf1_2-100GE1/0/2] quit
[*ServerLeaf1_2-100GE1/0/2] commit
```

      #配置ServerLeaf2_1与Border Leaf的互联接口地址：

```
[~ServerLeaf2_1] interface 100GE 1/0/1
[*ServerLeaf2_1-100GE1/0/1] description to BorderLeaf_1
```

```
[*ServerLeaf2_1-100GE1/0/1] undo portswitch
[*ServerLeaf2_1-100GE1/0/1] ip address 10.125.97.30 255.255.255.252
[*ServerLeaf2_1-100GE1/0/1] ospf network-type p2p
[*ServerLeaf2_1-100GE1/0/1] quit
[*ServerLeaf2_1] interface 100GE 1/0/2
[*ServerLeaf2_1-100GE1/0/2] description to BorderLeaf_2
[*ServerLeaf2_1-100GE1/0/2] undo portswitch
[*ServerLeaf2_1-100GE1/0/2] ip address 10.125.97.46 255.255.255.252
[*ServerLeaf2_1-100GE1/0/2] ospf network-type p2p
[*ServerLeaf2_1-100GE1/0/2] quit
[*ServerLeaf2_1-100GE1/0/2] commit
```

#配置ServerLeaf2_2与Border Leaf的互联接口地址：

```
[~ServerLeaf2_2] interface 100GE 1/0/1
[*ServerLeaf2_2-100GE1/0/1] description to BorderLeaf_1
[*ServerLeaf2_2-100GE1/0/1] undo portswitch
[*ServerLeaf2_2-100GE1/0/1] ip address 10.125.97.34 255.255.255.252
[*ServerLeaf2_2-100GE1/0/1] ospf network-type p2p
[*ServerLeaf2_2-100GE1/0/1] quit
[*ServerLeaf2_2] interface 100GE 1/0/2
[*ServerLeaf2_2-100GE1/0/2] description to BorderLeaf_2
[*ServerLeaf2_2-100GE1/0/2] undo portswitch
[*ServerLeaf2_2-100GE1/0/2] ip address 10.125.97.50 255.255.255.252
[*ServerLeaf2_2-100GE1/0/2] ospf network-type p2p
[*ServerLeaf2_2-100GE1/0/2] quit
[*ServerLeaf2_2-100GE1/0/2] commit
```

ii. 配置Server Leaf的Loopback接口地址。

#配置ServerLeaf1_1的Loopback接口地址：

```
[~ServerLeaf1_1] interface LoopBack 0
[*ServerLeaf1_1-LoopBack0] description VTEP
[*ServerLeaf1_1-LoopBack0] ipv6 enable    //当需要使用IPv6时，配置使能IPv6
[*ServerLeaf1_1-LoopBack0] ip address 10.125.99.2 255.255.255.255
[*ServerLeaf1_1-LoopBack0] quit
[*ServerLeaf1_1] interface LoopBack 1
[*ServerLeaf1_1-LoopBack1] description DFS-GROUP/ROUTER-ID
[*ServerLeaf1_1-LoopBack1] ip address 10.125.98.3 255.255.255.255
[*ServerLeaf1_1-LoopBack1] quit
[*ServerLeaf1_1] interface LoopBack 2
[*ServerLeaf1_1-LoopBack2] description Bypass VXLAN
[*ServerLeaf1_1-LoopBack2] ip address 10.135.98.3 255.255.255.255
[*ServerLeaf1_1-LoopBack2] quit
[*ServerLeaf1_1] commit
```

#配置ServerLeaf1_2的Loopback接口地址：

```
[~ServerLeaf1_2] interface LoopBack 0
[*ServerLeaf1_2-LoopBack0] description VTEP
[*ServerLeaf1_2-LoopBack0] ipv6 enable    //当需要使用IPv6时，配置使能IPv6
[*ServerLeaf1_2-LoopBack0] ip address 10.125.99.2 255.255.255.255
[*ServerLeaf1_2-LoopBack0] quit
[*ServerLeaf1_2] interface LoopBack 1
[*ServerLeaf1_2-LoopBack1] description DFS-GROUP/ROUTER-ID
[*ServerLeaf1_2-LoopBack1] ip address 10.125.98.4 255.255.255.255
[*ServerLeaf1_2-LoopBack1] quit
[*ServerLeaf1_2] interface LoopBack 2
[*ServerLeaf1_2-LoopBack2] description Bypass VXLAN
[*ServerLeaf1_2-LoopBack2] ip address 10.135.98.4 255.255.255.255
[*ServerLeaf1_2-LoopBack2] quit
[*ServerLeaf1_2] commit
```

#配置ServerLeaf2_1的Loopback接口地址：

```
[~ServerLeaf2_1] interface LoopBack 0
[*ServerLeaf2_1-LoopBack0] description VTEP
[*ServerLeaf2_1-LoopBack0] ipv6 enable    //当需要使用IPv6时，配置使能IPv6
[*ServerLeaf2_1-LoopBack0] ip address 10.125.99.3 255.255.255.255
[*ServerLeaf2_1-LoopBack0] quit
[*ServerLeaf2_1] interface LoopBack 1
[*ServerLeaf2_1-LoopBack1] description DFS-GROUP/ROUTER-ID
[*ServerLeaf2_1-LoopBack1] ip address 10.125.98.5 255.255.255.255
```

```
[*ServerLeaf2_1-LoopBack1] quit
[*ServerLeaf2_1] interface LoopBack 2
[*ServerLeaf2_1-LoopBack2] description Bypass VXLAN
[*ServerLeaf2_1-LoopBack2] ip address 10.135.98.5 255.255.255.255
[*ServerLeaf2_1-LoopBack2] quit
[*ServerLeaf2_1] commit
```

#配置ServerLeaf2_2的Loopback接口地址：

```
[~ServerLeaf2_2] interface LoopBack 0
[*ServerLeaf2_2-LoopBack0] description VTEP
[*ServerLeaf2_2-LoopBack0] ipv6 enable    //当需要使用IPv6时，配置使能IPv6
[*ServerLeaf2_2-LoopBack0] ip address 10.125.99.3 255.255.255.255
[*ServerLeaf2_2-LoopBack0] quit
[*ServerLeaf2_2] interface LoopBack 1
[*ServerLeaf2_2-LoopBack1] description DFS-GROUP/ROUTER-ID
[*ServerLeaf2_2-LoopBack1] ip address 10.125.98.6 255.255.255.255
[*ServerLeaf2_2-LoopBack1] quit
[*ServerLeaf2_2] interface LoopBack 2
[*ServerLeaf2_2-LoopBack2] description Bypass VXLAN
[*ServerLeaf2_2-LoopBack2] ip address 10.135.98.6 255.255.255.255
[*ServerLeaf2_2-LoopBack2] quit
[*ServerLeaf2_2] commit
```

iii. 配置NVE接口VTEP IP和虚拟MAC地址。

#配置ServerLeaf1_1的NVE接口：

```
[~ServerLeaf1_1] vlan 100
[*ServerLeaf1_1-vlan100] m-lag peer-link reserved
[*ServerLeaf1_1-vlan100] quit
[*ServerLeaf1_1] interface vlanif 100
[*ServerLeaf1_1-Vlanif100] ip address 10.125.96.5 30
[*ServerLeaf1_1-Vlanif100] reserved for vxlan bypass
[*ServerLeaf1_1-Vlanif100] quit
[*ServerLeaf1_1-Vlanif100] ip route-static 10.135.98.4 32 10.125.96.6 preference 1
[*ServerLeaf1_1] interface nve 1
[*ServerLeaf1_1-Nve1] source 10.125.99.2
[*ServerLeaf1_1-Nve1] mac-address 00e0-fc00-0102
[*ServerLeaf1_1-Nve1] pip-source 10.135.98.3 peer 10.135.98.4 bypass
[*ServerLeaf1_1-Nve1] commit
```

#配置ServerLeaf1_2的NVE接口：

```
[~ServerLeaf1_2] vlan 100
[*ServerLeaf1_2-vlan100] quit
[*ServerLeaf1_2-vlan100] m-lag peer-link reserved
[*ServerLeaf1_2] interface vlanif 100
[*ServerLeaf1_2-Vlanif100] ip address 10.125.96.6 30
[*ServerLeaf1_2-Vlanif100] reserved for vxlan bypass
[*ServerLeaf1_2-Vlanif100] quit
[*ServerLeaf1_2-Vlanif100] ip route-static 10.135.98.3 32 10.125.96.5 preference 1
[*ServerLeaf1_2] interface nve 1
[*ServerLeaf1_2-Nve1] source 10.125.99.2
[*ServerLeaf1_2-Nve1] mac-address 00e0-fc00-0102
[*ServerLeaf1_2-Nve1] pip-source 10.135.98.4 peer 10.135.98.3 bypass
[*ServerLeaf1_2-Nve1] commit
```

#配置ServerLeaf2_1的NVE接口：

```
[~ServerLeaf2_1] vlan 100
[*ServerLeaf2_1-vlan100] quit
[*ServerLeaf2_1-vlan100] m-lag peer-link reserved
[*ServerLeaf2_1] interface vlanif 100
[*ServerLeaf2_1-Vlanif100] ip address 10.125.96.9 30
[*ServerLeaf2_1-Vlanif100] reserved for vxlan bypass
[*ServerLeaf2_1-Vlanif100] quit
[*ServerLeaf2_1-Vlanif100] ip route-static 10.135.98.6 32 10.125.96.10 preference 1
[*ServerLeaf2_1] interface nve 1
[*ServerLeaf2_1-Nve1] source 10.125.99.3
[*ServerLeaf2_1-Nve1] mac-address 00e0-fc00-0103
[*ServerLeaf2_1-Nve1] pip-source 10.135.98.5 peer 10.135.98.6 bypass
[*ServerLeaf2_1-Nve1] commit
```

#配置ServerLeaf2_2的NVE接口：

```
[~ServerLeaf2_2] vlan 100
[*ServerLeaf2_2-vlan100] quit
[*ServerLeaf2_2-vlan100] m-lag peer-link reserved
[*ServerLeaf2_2] interface vlanif 100
[*ServerLeaf2_2-Vlanif100] ip address 10.125.96.10 30
[*ServerLeaf2_2-Vlanif100] reserved for vxlan bypass
[*ServerLeaf2_2-Vlanif100] quit
[*ServerLeaf2_2-Vlanif100] ip route-static 10.135.98.5 32 10.125.96.9 preference 1
[*ServerLeaf2_2] interface nve 1
[*ServerLeaf2_2-Nve1] source 10.125.99.3
[*ServerLeaf2_2-Nve1] mac-address 00e0-fc00-0103
[*ServerLeaf2_2-Nve1] pip-source 10.125.98.6 peer 10.125.98.5 bypass
[*ServerLeaf2_2-Nve1] commit
```

b. 配置M-LAG。

i. 配置M-LAG模式。

#配置ServerLeaf1_1的M-LAG模式：

```
[~ServerLeaf1_1] stp mode rstp
[*ServerLeaf1_1] stp v-stp enable        //配置V-STP方式的M-LAG
[*ServerLeaf1_1] stp tc-protection       //使能设备对TC类型BPDU报文的保护功能
[*ServerLeaf1_1] stp bpdu-protection     //使能设备的BPDU保护功能
[*ServerLeaf1_1] commit
```

#配置ServerLeaf1_2、ServerLeaf2_1、ServerLeaf2_2的M-LAG模式。配置过程及数据与ServerLeaf1_1一致，不再赘述。

ii. 配置M-LAG的DFS组。

#配置ServerLeaf1_1的DFS组：

```
[~ServerLeaf1_1] dfs-group 1
[*ServerLeaf1_1-dfs-group-1] priority 150    //配置DFS优先级高于对端，默认是100
[*ServerLeaf1_1-dfs-group-1] authentication-mode hmac-sha256 password
YsHsjx_202206
[*ServerLeaf1_1-dfs-group-1] dual-actice detection source ip 10.125.98.3
[*ServerLeaf1_1-dfs-group-1] consistency-check enable mode loose    //使能M-LAG配置
一致性检查，模式为松散模式
[*ServerLeaf1_1-dfs-group-1] quit
[*ServerLeaf1_1-dfs-group-1] commit
```

#配置ServerLeaf1_2的DFS组：

```
[~ServerLeaf1_2] dfs-group 1
[*ServerLeaf1_2-dfs-group-1] authentication-mode hmac-sha256 password
YsHsjx_202206
[*ServerLeaf1_2-dfs-group-1] dual-actice detection source ip 10.125.98.4
[*ServerLeaf1_2-dfs-group-1] consistency-check enable mode loose    //使能M-LAG配置
一致性检查，模式为松散模式
[*ServerLeaf1_2-dfs-group-1] quit
[*ServerLeaf1_2-dfs-group-1] commit
```

#配置ServerLeaf2_1的DFS组：

```
[~ServerLeaf2_1] dfs-group 1
[*ServerLeaf2_1-dfs-group-1] priority 150    //配置DFS优先级高于对端，默认是100
[*ServerLeaf2_1-dfs-group-1] authentication-mode hmac-sha256 password
YsHsjx_202206
[*ServerLeaf2_1-dfs-group-1] dual-actice detection source ip 10.125.98.5
[*ServerLeaf2_1-dfs-group-1] consistency-check enable mode loose    //使能M-LAG配置
一致性检查，模式为松散模式
[*ServerLeaf2_1-dfs-group-1] quit
[*ServerLeaf2_1-dfs-group-1] commit
```

#配置ServerLeaf2_2的DFS组：

```
[~ServerLeaf2_2] dfs-group 1
[*ServerLeaf2_2-dfs-group-1] authentication-mode hmac-sha256 password
YsHsjx_202206
[*ServerLeaf2_2-dfs-group-1] dual-actice detection source ip 10.125.98.6
[*ServerLeaf2_2-dfs-group-1] consistency-check enable mode loose    //使能M-LAG配置
```

```
一致性检查，模式为松散模式
[*ServerLeaf2_2-dfs-group-1] quit
[*ServerLeaf2_2-dfs-group-1] commit
```

iii. 配置peer-link。

#配置ServerLeaf1_1的peer-link：

```
[~ServerLeaf1_1] interface Eth-Trunk 0
[*ServerLeaf1_1-Eth-Trunk0] trunkport 100GE 1/0/5 to 1/0/6
[*ServerLeaf1_1-Eth-Trunk0] mode lacp-static
[*ServerLeaf1_1-Eth-Trunk0] peer-link 1
[*ServerLeaf1_1-Eth-Trunk0] commit
```

#配置ServerLeaf1_2的peer-link：

```
[~ServerLeaf1_2] interface Eth-Trunk 0
[*ServerLeaf1_2-Eth-Trunk0] trunkport 100GE 1/0/5 to 1/0/6
[*ServerLeaf1_2-Eth-Trunk0] mode lacp-static
[*ServerLeaf1_2-Eth-Trunk0] peer-link 1
[*ServerLeaf1_2-Eth-Trunk0] commit
```

#配置ServerLeaf2_1的peer-link：

```
[~ServerLeaf2_1] interface Eth-Trunk 0
[*ServerLeaf2_1-Eth-Trunk0] trunkport 100GE 1/0/5 to 1/0/6
[*ServerLeaf2_1-Eth-Trunk0] mode lacp-static
[*ServerLeaf2_1-Eth-Trunk0] peer-link 1
[*ServerLeaf2_1-Eth-Trunk0] commit
```

#配置ServerLeaf2_2的peer-link：

```
[~ServerLeaf2_2] interface Eth-Trunk 0
[*ServerLeaf2_2-Eth-Trunk0] trunkport 100GE 1/0/5 to 1/0/6
[*ServerLeaf2_2-Eth-Trunk0] mode lacp-static
[*ServerLeaf2_2-Eth-Trunk0] peer-link 1
[*ServerLeaf2_2-Eth-Trunk0] commit
```

iv. 配置业务服务器以负载分担方式接入。

#配置ServerLeaf1_1与业务服务器对接：

```
[~ServerLeaf1_1] interface Eth-Trunk 10          //配置边缘端口
[*ServerLeaf1_1-Eth-Trunk10] port link-type trunk
[*ServerLeaf1_1-Eth-Trunk10] undo port trunk allow-pass vlan 1
[*ServerLeaf1_1-Eth-Trunk10] trunkport 100GE 1/0/1
[*ServerLeaf1_1-Eth-Trunk10] dfs-group 1 m-lag 10
[*ServerLeaf1_1-Eth-Trunk10] mode lacp-static
[*ServerLeaf1_1-Eth-Trunk10] stp edged-port enable
[*ServerLeaf1_1-Eth-Trunk10] quit
[*ServerLeaf1_1] interface 100GE 1/0/1           //服务器接入端口
[*ServerLeaf1_1-100GE1/0/1] storm suppression unknown-unicast 5   //配置未知单播抑制，经验值为100GE端口的5%带宽，建议业务端口都部署
[*ServerLeaf1_1-100GE1/0/1] storm suppression multicast packets 1000  //配置未知组播报文抑制，经验值为1000pps。
[*ServerLeaf1_1-100GE1/0/1] storm suppression broadcast packets 1000  //配置广播报文抑制，经验值为1000pps，建议业务端口都部署
[*ServerLeaf1_1-100GE1/0/1] commit
```

#配置ServerLeaf1_2与业务服务器对接：

```
[~ServerLeaf1_2] interface Eth-Trunk 10          //配置边缘端口
[*ServerLeaf1_2-Eth-Trunk10] port link-type trunk
[*ServerLeaf1_2-Eth-Trunk10] undo port trunk allow-pass vlan 1
[*ServerLeaf1_2-Eth-Trunk10] trunkport 100GE 1/0/1
[*ServerLeaf1_2-Eth-Trunk10] dfs-group 1 m-lag 10
[*ServerLeaf1_2-Eth-Trunk10] mode lacp-static
[*ServerLeaf1_2-Eth-Trunk10] stp edged-port enable
[*ServerLeaf1_2-Eth-Trunk10] quit
[*ServerLeaf1_2] interface 100GE 1/0/1           //服务器接入端口
[*ServerLeaf1_2-100GE1/0/1] storm suppression unknown-unicast 5   //配置未知单播抑制，经验值为100GE端口的5%带宽，建议业务端口都部署
[*ServerLeaf1_2-100GE1/0/1] storm suppression multicast packets 1000  //配置未知组播报文抑制，经验值为1000pps。
[*ServerLeaf1_2-100GE1/0/1] storm suppression broadcast packets 1000  //配置广播报文
```



```
抑制，经验值为1000pps，建议业务端口都部署
[*ServerLeaf1_2-100GE1/0/1] commit
```

#配置ServerLeaf2_1、ServerLeaf2_2与业务服务器对接。配置过程与上述配置类似，不再赘述。

v. 配置业务服务器以主备方式接入。

#配置ServerLeaf1_1与业务服务器对接：

```
[~ServerLeaf1_1] interface 100GE 1/0/2
[*ServerLeaf1_1-100GE1/0/2] port link-type trunk
[*ServerLeaf1_1-100GE1/0/2] undo port trunk allow-pass vlan 1  //不放通VLAN1，防止成环
[*ServerLeaf1_1-100GE1/0/2] storm suppression unknown-unicast 5  //配置未知单播抑制，经验值为100GE端口的5%带宽，建议业务端口都部署
[*ServerLeaf1_1-100GE1/0/2] storm suppression multicast packets 1000  //配置未知组播报文抑制，经验值为1000pps。
[*ServerLeaf1_1-100GE1/0/2] storm suppression broadcast packets 1000  //配置广播报文抑制，经验值为1000pps，建议业务端口都部署
[*ServerLeaf1_1-100GE1/0/2] stp edged-port enable
[*ServerLeaf1_1-100GE1/0/2] commit
```

#配置ServerLeaf1_2与业务服务器对接：

```
[~ServerLeaf1_2] interface 100GE 1/0/2
[*ServerLeaf1_2-100GE1/0/2] port link-type trunk
[*ServerLeaf1_2-100GE1/0/2] undo port trunk allow-pass vlan 1  //不放通VLAN1，防止成环
[*ServerLeaf1_2-100GE1/0/2] storm suppression unknown-unicast 5  //配置未知单播抑制，经验值为100GE端口的5%带宽，建议业务端口都部署
[*ServerLeaf1_2-100GE1/0/2] storm suppression multicast packets 1000  //配置未知组播报文抑制，经验值为1000pps。
[*ServerLeaf1_2-100GE1/0/2] storm suppression broadcast packets 1000  //配置广播报文抑制，经验值为1000pps，建议业务端口都部署
[*ServerLeaf1_2-100GE1/0/2] stp edged-port enable
[*ServerLeaf1_2-100GE1/0/2] commit
```

#配置ServerLeaf2_1、ServerLeaf2_2与业务服务器对接。配置过程与上述配置类似，不再赘述。

vi. 配置monitor-link关联上行接口和下行接口，避免单台设备的所有上行链路都故障时，本台设备用户侧流量无法转发。

Downlink只列出了1个端口做示例，实际部署时请根据规划补齐。

#配置ServerLeaf1_1的monitor-link：

```
[~ServerLeaf1_1] monitor-link group 1
[*ServerLeaf1_1-mtlk-group1] port 100GE1/0/1 uplink
[*ServerLeaf1_1-mtlk-group1] port 100GE1/0/2 uplink
[*ServerLeaf1_1-mtlk-group1] port Eth-Trunk10 downlink 1
[*ServerLeaf1_1-mtlk-group1] timer recover-time 60      //配置回切时间，防止上行故障回切丢包。
[*ServerLeaf1_1-mtlk-group1] commit
```

#配置ServerLeaf1_2、ServerLeaf2_1、ServerLeaf2_2的monitor-link。配置过程及数据与ServerLeaf1_1一致，不再赘述。

c. 配置路由。

i. 配置OSPF路由打通VXLAN Underlay路由。

#配置ServerLeaf1_1的OSPF路由：

```
[~ServerLeaf1_1] bfd    //全局使能BFD功能
[*ServerLeaf1_1-bfd] quit
[*ServerLeaf1_1] ospf 1 router-id 10.125.98.3
[*ServerLeaf1_1-ospf-1] bfd all-interfaces enable
[*ServerLeaf1_1-ospf-1] bfd all-interfaces min-tx-interval 500 min-rx-interval 500 detect-multiplier 3
[*ServerLeaf1_1-ospf-1] lsa-arrival-interval intelligent-timer 50 50 50    //设置OSPF LSA接收的时间间隔，优化收敛时间
[*ServerLeaf1_1-ospf-1] area 0.0.0.0
```

```
[*ServerLeaf1_1-ospf-1-area-0.0.0.0] network 10.125.97.20 0.0.0.3      //分别建立与2台Border Leaf设备的路由邻居
[*ServerLeaf1_1-ospf-1-area-0.0.0.0] network 10.125.97.36 0.0.0.3
[*ServerLeaf1_1-ospf-1-area-0.0.0.0] network 10.125.98.3 0.0.0.0      //发布Loopback地址
[*ServerLeaf1_1-ospf-1-area-0.0.0.0] network 10.125.99.2 0.0.0.0
[*ServerLeaf1_1-ospf-1-area-0.0.0.0] quit
[*ServerLeaf1_1-ospf-1] quit
[*ServerLeaf1_1-ospf-1] commit
```

#配置ServerLeaf1_2的OSPF路由：

```
[~ServerLeaf1_2] bfd
[*ServerLeaf1_2-bfd] quit
[*ServerLeaf1_2] ospf 1 router-id 10.125.98.4
[*ServerLeaf1_2-ospf-1] bfd all-interfaces enable
[*ServerLeaf1_2-ospf-1] bfd all-interfaces min-tx-interval 500 min-rx-interval 500 detect-multiplier 3
[*ServerLeaf1_2-ospf-1] lsa-arrival-interval intelligent-timer 50 50 50      //设置OSPF LSA接收的时间间隔，优化收敛时间
[*ServerLeaf1_2-ospf-1] area 0.0.0.0
[*ServerLeaf1_2-ospf-1-area-0.0.0.0] network 10.125.97.24 0.0.0.3      //分别建立与2台Border Leaf设备的路由邻居
[*ServerLeaf1_2-ospf-1-area-0.0.0.0] network 10.125.97.40 0.0.0.3
[*ServerLeaf1_2-ospf-1-area-0.0.0.0] network 10.125.98.4 0.0.0.0      //发布Loopback地址
[*ServerLeaf1_2-ospf-1-area-0.0.0.0] network 10.125.99.2 0.0.0.0
[*ServerLeaf1_2-ospf-1-area-0.0.0.0] quit
[*ServerLeaf1_2-ospf-1] quit
[*ServerLeaf1_2-ospf-1] commit
```

#配置ServerLeaf2_1的OSPF路由：

```
[~ServerLeaf2_1] bfd
[*ServerLeaf2_1-bfd] quit
[*ServerLeaf2_1] ospf 1 router-id 10.125.98.5
[*ServerLeaf2_1-ospf-1] bfd all-interfaces enable
[*ServerLeaf2_1-ospf-1] bfd all-interfaces min-tx-interval 500 min-rx-interval 500 detect-multiplier 3
[*ServerLeaf2_1-ospf-1] lsa-arrival-interval intelligent-timer 50 50 50      //设置OSPF LSA接收的时间间隔，优化收敛时间
[*ServerLeaf2_1-ospf-1] area 0.0.0.0
[*ServerLeaf2_1-ospf-1-area-0.0.0.0] network 10.125.97.28 0.0.0.3      //分别建立与2台Border Leaf设备的路由邻居
[*ServerLeaf2_1-ospf-1-area-0.0.0.0] network 10.125.97.44 0.0.0.3
[*ServerLeaf2_1-ospf-1-area-0.0.0.0] network 10.125.98.5 0.0.0.0      //发布Loopback地址
[*ServerLeaf2_1-ospf-1-area-0.0.0.0] network 10.125.99.3 0.0.0.0
[*ServerLeaf2_1-ospf-1-area-0.0.0.0] quit
[*ServerLeaf2_1-ospf-1] quit
[*ServerLeaf2_1-ospf-1] commit
```

#配置ServerLeaf2_2的OSPF路由：

```
[~ServerLeaf2_2] bfd
[*ServerLeaf2_2-bfd] quit
[*ServerLeaf2_2] ospf 1 router-id 10.125.98.6
[*ServerLeaf2_2-ospf-1] bfd all-interfaces enable
[*ServerLeaf2_2-ospf-1] bfd all-interfaces min-tx-interval 500 min-rx-interval 500 detect-multiplier 3
[*ServerLeaf2_2-ospf-1] lsa-arrival-interval intelligent-timer 50 50 50      //设置OSPF LSA接收的时间间隔，优化收敛时间
[*ServerLeaf2_2-ospf-1] area 0.0.0.0
[*ServerLeaf2_2-ospf-1-area-0.0.0.0] network 10.125.97.32 0.0.0.3      //分别建立与2台Border Leaf设备的路由邻居
[*ServerLeaf2_2-ospf-1-area-0.0.0.0] network 10.125.97.48 0.0.0.3
[*ServerLeaf2_2-ospf-1-area-0.0.0.0] network 10.125.98.6 0.0.0.0      //发布Loopback地址
[*ServerLeaf2_2-ospf-1-area-0.0.0.0] network 10.125.99.3 0.0.0.0
[*ServerLeaf2_2-ospf-1-area-0.0.0.0] quit
[*ServerLeaf2_2-ospf-1] quit
[*ServerLeaf2_2-ospf-1] commit
```

ii. 配置OSPF网络故障收敛性能优化。

#配置ServerLeaf1_1的OSPF网络故障收敛性能优化：

```
[~ServerLeaf1_1] interface 100GE 1/0/1
[*ServerLeaf1_1-100GE1/0/1] ospf peer hold-max-cost timer 300000  //所有Border Leaf和Server Leaf配置OSPF邻居建立后在本地设备的LSA中保持最大开销值的时间300s，源于240s的M-LAG延迟UP时间（同时overlay路由收敛）+ 60s的设备表项同步时间
[*ServerLeaf1_1-100GE1/0/1] quit
[*ServerLeaf1_1] interface 100GE 1/0/2
[*ServerLeaf1_1-100GE1/0/2] ospf peer hold-max-cost timer 300000
[*ServerLeaf1_1-100GE1/0/2] quit
[*ServerLeaf1_1-100GE1/0/2] commit
```

#配置ServerLeaf1_2、ServerLeaf2_1、ServerLeaf2_2的OSPF网络故障收敛性能优化，配置过程及数据与ServerLeaf1_1一致，不再赘述。

iii. 配置BGP EVPN。

#配置ServerLeaf1_1的BGP EVPN：

```
[~ServerLeaf1_1] evpn-overlay enable  //使能EVPN作为VXLAN的控制平面
[*ServerLeaf1_1] bgp 100
[*ServerLeaf1_1-bgp] router-id 10.125.98.3
[*ServerLeaf1_1-bgp] advertise lowest-priority all-address-family peer-up delay 360  //在邻居状态由Down到Up时将BGP路由的优先级调整为最低优先级；路由延时发布，解决回切场景丢包时间长问题
[*ServerLeaf1_1-bgp] undo default ipv4-unicast  //关闭BGP IPv4单播邻居，降低设备负荷
[*ServerLeaf1_1-bgp] group BorderLeaf internal  //配置名为BorderLeaf的对等体组并加入相应对等体。
[*ServerLeaf1_1-bgp] peer 10.125.98.1 group BorderLeaf
[*ServerLeaf1_1-bgp] peer 10.125.98.2 group BorderLeaf
[*ServerLeaf1_1-bgp] peer ServerLeaf connect-interface LoopBack1  //指定发送BGP报文的源接口
[*ServerLeaf1_1-bgp] l2vpn-family evpn  //使能并进入BGP-EVPN地址族视图
[*ServerLeaf1_1-bgp-af-evpn] undo policy vpn-target  //配置去使能对接收到的EVPN路由使能VPN-Target过滤功能
[*ServerLeaf1_1-bgp-af-evpn] peer BorderLeaf enable
[*ServerLeaf1_1-bgp-af-evpn] peer 10.125.98.1 group BorderLeaf
[*ServerLeaf1_1-bgp-af-evpn] peer 10.125.98.2 group BorderLeaf
[*ServerLeaf1_1-bgp-af-evpn] peer BorderLeaf advertise irb  //配置向BGP EVPN对等体组BorderLeaf发布irb和irbv6路由
[*ServerLeaf1_1-bgp-af-evpn] peer BorderLeaf advertise irbv6
[*ServerLeaf1_1-bgp-af-evpn] quit
[*ServerLeaf1_1-bgp] quit
[*ServerLeaf1_1-bgp] commit
```

#配置ServerLeaf1_2的BGP EVPN：

```
[~ServerLeaf1_2] evpn-overlay enable  //使能EVPN作为VXLAN的控制平面
[*ServerLeaf1_2] bgp 100
[*ServerLeaf1_2-bgp] router-id 10.125.98.4
[*ServerLeaf1_2-bgp] advertise lowest-priority all-address-family peer-up delay 360  //在邻居状态由Down到Up时将BGP路由的优先级调整为最低优先级；路由延时发布，解决回切场景丢包时间长问题
[*ServerLeaf1_2-bgp] undo default ipv4-unicast  //关闭BGP IPv4单播邻居，降低设备负荷
[*ServerLeaf1_2-bgp] group BorderLeaf internal  //配置名为BorderLeaf的对等体组并加入相应对等体。
[*ServerLeaf1_2-bgp] peer 10.125.98.1 group BorderLeaf
[*ServerLeaf1_2-bgp] peer 10.125.98.2 group BorderLeaf
[*ServerLeaf1_2-bgp] peer ServerLeaf connect-interface LoopBack1  //指定发送BGP报文的源接口
[*ServerLeaf1_2-bgp] l2vpn-family evpn  //使能并进入BGP-EVPN地址族视图
[*ServerLeaf1_2-bgp-af-evpn] undo policy vpn-target  //配置去使能对接收到的EVPN路由使能VPN-Target过滤功能
[*ServerLeaf1_2-bgp-af-evpn] peer BorderLeaf enable
[*ServerLeaf1_2-bgp-af-evpn] peer 10.125.98.1 group BorderLeaf
[*ServerLeaf1_2-bgp-af-evpn] peer 10.125.98.2 group BorderLeaf
[*ServerLeaf1_2-bgp-af-evpn] peer BorderLeaf advertise irb  //配置向BGP EVPN对等体组BorderLeaf发布irb和irbv6路由
[*ServerLeaf1_2-bgp-af-evpn] peer BorderLeaf advertise irbv6
[*ServerLeaf1_2-bgp-af-evpn] quit
[*ServerLeaf1_2-bgp] quit
[*ServerLeaf1_2-bgp] commit
```

#配置ServerLeaf2_1、ServerLeaf2_2的BGP EVPN。配置过程与上述配置类似，不再赘述。

3. 配置防火墙。
   a. 配置防火墙基础信息。
      i. 配置防火墙的设备名称。

      #配置防火墙FW-1的设备名称：

      ```
      <HUAWEI> system-view
      [~HUAWEI] sysname FW-1
      ```

      #配置防火墙FW-2的设备名称：

      ```
      <HUAWEI> system-view
      [~HUAWEI] sysname FW-2
      ```

      ii. 配置防火墙管理口IP。

      #配置防火墙FW-1的管理IP：

      ```
      [~FW-1] interface 10GE 0/0/0
      [*FW-1-10GE0/0/0] ip address 192.168.39.50 24
      [*FW-1-10GE0/0/0] service-manage http permit
      [*FW-1-10GE0/0/0] service-manage https permit
      [*FW-1-10GE0/0/0] service-manage ping permit
      [*FW-1-10GE0/0/0] quit
      ```

      #配置防火墙FW-2的管理IP：

      ```
      [~FW-2] interface 10GE 0/0/0
      [*FW-2-10GE0/0/0] ip address 192.168.39.51 24
      [*FW-2-10GE0/0/0] service-manage http permit
      [*FW-2-10GE0/0/0] service-manage https permit
      [*FW-2-10GE0/0/0] service-manage ping permit
      [*FW-2-10GE0/0/0] quit
      ```

   b. 关闭备份当前运行配置的功能，在主备防火墙上均需要配置。

      #在FW-1上，关闭备份当前运行配置的功能：

      ```
      [~FW-1] configuration backup local disable
      ```

      #在FW2上，关闭备份当前运行配置的功能：

      ```
      [~FW-2] configuration backup local disable
      ```

   c. 配置防火墙与Border Leaf互联端口。

      #配置FW-1上的业务端口：

      ```
      [~FW-1] interface Eth-Trunk11
      [*FW-1-Eth-Trunk11] portswitch
      [*FW-1-Eth-Trunk11] port link-type trunk
      [*FW-1-Eth-Trunk11] undo port trunk allow-pass vlan 1
      [*FW-1-Eth-Trunk11] trunkport 10GE 1/0/8 to 1/0/9
      [*FW-1-Eth-Trunk11] mode lacp-static
      [*FW-1-Eth-Trunk11] quit
      [*FW-1] interface 10GE1/0/8  //开启当前使用的接口
      [*FW-1-10GE1/0/8] undo shutdown
      [*FW-1-10GE1/0/8] quit
      [*FW-1] interface 10GE1/0/9
      [*FW-1-10GE1/0/9] undo shutdown
      [*FW-1-10GE1/0/9] quit
      ```

      #配置FW-2上的业务端口，配置过程及数据与FW-1一致，不再赘述。

   d. 配置两台防火墙之间的心跳接口。

      #配置FW-1上的心跳接口：

      ```
      [~FW-1] interface Eth-Trunk0
      [*FW-1-Eth-Trunk0] description HRP
      [*FW-1-Eth-Trunk0] ip address 10.125.97.73 255.255.255.252
      [*FW-1-Eth-Trunk0] trunkport 10GE 1/0/0 to 1/0/1
      [*FW-1-Eth-Trunk0] mode lacp-static
      ```



```
[*FW-1-Eth-Trunk0] quit
[*FW-1] interface 10GE1/0/0  //开启当前使用的接口
[*FW-1-10GE1/0/0] undo shutdown
[*FW-1-10GE1/0/0] quit
[*FW-1] interface 10GE1/0/1
[*FW-1-10GE1/0/1] undo shutdown
[*FW-1-10GE1/0/1] quit
```

#配置FW-2上的心跳接口：

```
[~FW-2] interface Eth-Trunk0
[*FW-2-Eth-Trunk0] description HRP
[*FW-2-Eth-Trunk0] ip address 10.125.97.74 255.255.255.252
[*FW-2-Eth-Trunk0] trunkport 10GE 1/0/0 to 1/0/1
[*FW-2-Eth-Trunk0] mode lacp-static
[*FW-2-Eth-Trunk0] quit
[*FW-2] interface 10GE1/0/0  //开启当前使用的接口
[*FW-2-10GE1/0/0] undo shutdown
[*FW-2-10GE1/0/0] quit
[*FW-2] interface 10GE1/0/1
[*FW-2-10GE1/0/1] undo shutdown
[*FW-2-10GE1/0/1] quit
```

e. 配置两台防火墙的主备镜像模式。

#配置FW-1：

```
[~FW-1] hrp interface Eth-Trunk0 remote 10.125.97.74    //指定心跳口
[*FW-1] hrp mirror config enable                        //配置镜像模式
[*FW-1] hrp enable                                      //启用双机热备功能
[*FW-1] hrp track interface Eth-Trunk11                 //配置VGMP组监控上下行业务接口
[*FW-1] undo hrp track trunk-member enable              //关闭hrp监控trunk成员接口状态
[*FW-1] hrp mgt-interface Eth-Trunk1                    //配置双机热备管理接口
[*FW-1] hrp mirror session enable                       //启用会话快速备份功能
[*FW-1] hrp standby config enable                       //开启备用设备的部分配置功能
[*FW-1] hrp base config enable                          //配置FW启动时以基础配置启动，其他配置从对端设备同步。
[*FW-1] undo hrp preempt                                //关闭防火墙镜像模式下的主备抢占
```

#配置FW-2：

```
[~FW-2] hrp interface Eth-Trunk0 remote 10.125.97.73
[*FW-2] hrp mirror config enable
[*FW-2] hrp enable          //备防火墙配置完镜像模式，启用双机热备功能后，后续配置可以从主防火墙同步
[*FW-2] hrp track interface Eth-Trunk11
[*FW-2] undo hrp track trunk-member enable
[*FW-2] hrp mgt-interface Eth-Trunk1
[*FW-2] hrp mirror session enable
[*FW-2] hrp standby config enable
[*FW-2] hrp base config enable
[*FW-2] undo hrp preempt
```

f. 配置安全域及缺省安全策略。如下只需要在FW-1中进行配置，FW2将自动同步。

#配置Virtual-if0、管理口和心跳口加入安全域：

```
[~FW-1] firewall zone untrust
[*FW-1-zone-untrust] add interface Virtual-if0
[*FW-1-zone-untrust] quit
[*FW-1] firewall zone dmz
[*FW-1-zone-dmz] add interface Eth-Trunk1
[*FW-1-zone-dmz] add interface Eth-Trunk0
[*FW-1-zone-dmz] quit
```

#配置缺省的安全策略为permit：

```
[~FW-1] security-policy
[*FW-1-policy-security] default action permit
[*FW-1-policy-security] quit
```

g. 使能防火墙的vsys功能。如下只需要在FW-1中进行配置，FW-2将自动同步。

```
[~FW-1] vsys enable   //使能防火墙的vsys功能
[*FW-1] interface Virtual-if api transform    //开启北向接口的Virtual-if名称转换功能
```

```
[*FW-1] firewall forward cross-vsys extended  //将FW配置为扩展模式，同一报文最多可以实现跨两次vsys转发。不同VPN之间通过EIP在出口vsys互通场景必须配置。
```

步骤3 配置Overlay网络。

1. 配置Server Leaf。

   a. 配置业务接入点。

   # 配置ServerLeaf1_1的业务接入点。

```
[~ServerLeaf1_1] bridge-domain 10
[*ServerLeaf1_1-bd10] quit
[*ServerLeaf1_1] interface eth-trunk 10.1 mode l2
[*ServerLeaf1_1-Eth-Trunk10.1] encapsulation dot1q vid 10
[*ServerLeaf1_1-Eth-Trunk10.1] bridge-domain 10
[*ServerLeaf1_1-Eth-Trunk10.1] quit
[*ServerLeaf1_1-Eth-Trunk10.1] commit
```

   # 配置ServerLeaf1_2的业务接入点。

```
[~ServerLeaf1_2] bridge-domain 10
[*ServerLeaf1_2-bd10] quit
[*ServerLeaf1_2] interface eth-trunk 10.1 mode l2
[*ServerLeaf1_2-Eth-Trunk10.1] encapsulation dot1q vid 10
[*ServerLeaf1_2-Eth-Trunk10.1] bridge-domain 10
[*ServerLeaf1_2-Eth-Trunk10.1] quit
[*ServerLeaf1_2-Eth-Trunk10.1] commit
```

   # 配置ServerLeaf2_1、ServerLeaf2_2的业务接入点，配置过程与上述配置类似，只是需要将bd10配置为bd20。

   b. 配置VPN实例和EVPN实例。

   # 配置ServerLeaf1_1。

```
[~ServerLeaf1_1] ip vpn-instance vpn1
[*ServerLeaf1_1-vpn-instance-vpn1] vxlan vni 5010
[*ServerLeaf1_1-vpn-instance-vpn1] ipv4-family
[*ServerLeaf1_1-vpn-instance-vpn1-af-ipv4] route-distinguisher 20:2
[*ServerLeaf1_1-vpn-instance-vpn1-af-ipv4] vpn-target 100:5010 evpn
[*ServerLeaf1_1-vpn-instance-vpn1-af-ipv4] quit
[*ServerLeaf1_1-vpn-instance-vpn1] quit
[*ServerLeaf1_1] bridge-domain 10
[*ServerLeaf1_1-bd10] vxlan vni 10
[*ServerLeaf1_1-bd10] evpn
[*ServerLeaf1_1-bd10-evpn] route-distinguisher 10:2
[*ServerLeaf1_1-bd10-evpn] vpn-target 100:10
[*ServerLeaf1_1-bd10-evpn] vpn-target 100:5010 export-extcommunity
[*ServerLeaf1_1-bd10-evpn] quit
[*ServerLeaf1_1-bd10] quit
[*ServerLeaf1_1] commit
```

   # 配置ServerLeaf1_2。

```
[~ServerLeaf1_2] ip vpn-instance vpn2
[*ServerLeaf1_2-vpn-instance-vpn1] vxlan vni 5010
[*ServerLeaf1_2-vpn-instance-vpn1] ipv4-family
[*ServerLeaf1_2-vpn-instance-vpn1-af-ipv4] route-distinguisher 20:4
[*ServerLeaf1_2-vpn-instance-vpn1-af-ipv4] vpn-target 100:5010 evpn
[*ServerLeaf1_2-vpn-instance-vpn1-af-ipv4] quit
[*ServerLeaf1_2-vpn-instance-vpn1] quit
[*ServerLeaf1_2] bridge-domain 10
[*ServerLeaf1_2-bd10] vxlan vni 10
[*ServerLeaf1_2-bd10] evpn
[*ServerLeaf1_2-bd10-evpn] route-distinguisher 10:4
[*ServerLeaf1_2-bd10-evpn] vpn-target 100:10
[*ServerLeaf1_2-bd10-evpn] vpn-target 100:5010 export-extcommunity
[*ServerLeaf1_2-bd10-evpn] quit
[*ServerLeaf1_2-bd10] quit
[*ServerLeaf1_2] commit
```

   # 配置ServerLeaf2_1、ServerLeaf2_2，配置过程与上述配置类似。

c. 配置VNI的头端复制列表。

#配置ServerLeaf1_1的头端复制列表：

```
[~ServerLeaf1_1] interface nve 1
[*ServerLeaf1_1-Nve1] vni 10 head-end peer-list protocol bgp
```

#配置ServerLeaf1_2的头端复制列表：

```
[~ServerLeaf1_2] interface nve 1
[*ServerLeaf1_2-Nve1] vni 10 head-end peer-list protocol bgp
```

ServerLeaf2_1、ServerLeaf2_2的配置与上述配置类似，这里不再赘述。

d. 在Server Leaf上配置VXLAN三层网关。

在ServerLeaf1_1上配置VXLAN三层网关。

```
[~ServerLeaf1_1] interface vbdif10
[*ServerLeaf1_1-Vbdif10] ip binding vpn-instance vpn1
[*ServerLeaf1_1-Vbdif10] ip address 10.1.1.1 255.255.255.0
[*ServerLeaf1_1-Vbdif10] mac-address 00e0-fc00-0102
[*ServerLeaf1_1-Vbdif10] vxlan anycast-gateway enable
[*ServerLeaf1_1-Vbdif10] arp collect host enable
[*ServerLeaf1_1-Vbdif10] arp broadcast-detect enable
[*ServerLeaf1_1-Vbdif10] quit
[*ServerLeaf1_1] commit
```

ServerLeaf1_2、ServerLeaf2_1、ServerLeaf2_2的配置与ServerLeaf1_1类似，这里不再赘述。要注意ServerLeaf1_1与ServerLeaf1_2要配置相同的Vbdif接口的IP地址和MAC地址。

e. 配置BGP对邻居发布IP前缀路由。

配置ServerLeaf1_1。ServerLeaf1_2、ServerLeaf2_1、ServerLeaf2_2的配置与ServerLeaf1_1类似，这里不再赘述。

```
[~ServerLeaf1_1] bgp 100
[~ServerLeaf1_1-bgp] ipv4-family vpn-instance vpn1
[*ServerLeaf1_1-bgp] import-route direct
[*ServerLeaf1_1-bgp] import-route static //对接防火墙的外部路由
[*ServerLeaf1_1-bgp] advertise l2vpn evpn
[*ServerLeaf1_1-bgp] quit
[*ServerLeaf1_1-bgp] quit
[*ServerLeaf1_1] commit
```

2. 配置Border Leaf。

配置通过Border Leaf访问外网的静态路由。

配置Border_Leaf1_1。

```
[~Border_Leaf1_1] ip route-static 1.2.3.4 255.255.255.0 10.125.97.242
```

配置Border_Leaf1_2。

```
[~Border_Leaf1_2] ip route-static 1.2.3.4 255.255.255.0 10.125.97.242
```

3. 配置FW。

关联外部网络的业务，经过防火墙转发，但源IP未改变。为了实现租户使用公网IP与Internet互通，需要部署SNAT实现源IP地址的公私网转换。

a. 配置vsys关键参数。

配置FW-1。

```
[~FW-1] vlan 3004
[~FW-1-vlan3004] quit
[~FW-1] interface Vlanif3004
[~FW-1-Vlanif3004] ip binding vpn-instance vsys_1
[~FW-1-Vlanif3004] ip address 10.125.97.242 255.255.255.252
[~FW-1-Vlanif3004] quit
[~FW-1] ip route-static 0.0.0.0 0.0.0.0 public
[~FW-1] ip route-static 10.132.1.0 255.255.255.0 10.125.97.241
```

配置FW-2。

```
[~FW-2] vlan 3004
[~FW-2-vlan3004] quit
[~FW-2] interface Vlanif3004
[~FW-2-Vlanif3004] ip binding vpn-instance vsys_1
[~FW-2-Vlanif3004] ip address 10.125.97.242 255.255.255.252
[~FW-2-Vlanif3004] quit
[~FW-2] ip route-static 0.0.0.0 0.0.0.0 public
[~FW-2] ip route-static 10.132.1.0 255.255.255.0 10.125.97.241
```

b. 配置public关键参数。

配置FW-1。

```
[~FW-1] vsys name vsys_1 1
[~FW-1-vsys_1] assign vlan 3004
[~FW-1-vsys_1] quit
[~FW-1] ip vpn-instance vsys_1
[~FW-1-vsys_1] ipv4-family
[~FW-1-vsys_1] quit
[~FW-1] vlan 3005
[~FW-1-vlan3005] quit
[~FW-1] interface Vlanif3005
[~FW-1-Vlanif3005] ip address 10.125.97.242 255.255.255.252
[~FW-1-Vlanif3005] quit
[~FW-1] interface Eth-Trunk11
[~FW-1-Eth-Trunk11] portswitch
[~FW-1-Eth-Trunk11] port link-type trunk
[~FW-1-Eth-Trunk11] undo port trunk allow-pass vlan 1
[~FW-1-Eth-Trunk11] port trunk allow-pass vlan 3004 to 3005
[~FW-1-Eth-Trunk11] quit
[~FW-1] ip route-static 0.0.0.0 0.0.0.0 10.125.97.241
[~FW-1] ip route-static 10.132.1.0 255.255.255.0 vpn-instance vsys_1
```

配置FW-2。

```
[~FW-2] vsys name vsys_1 1
[~FW-2-vsys_1] assign vlan 3004
[~FW-2-vsys_1] quit
[~FW-2] ip vpn-instance vsys_1
[~FW-2-vsys_1] ipv4-family
[~FW-2-vsys_1] quit
[~FW-2] vlan 3005
[~FW-2-vlan3005] quit
[~FW-2] interface Vlanif3005
[~FW-2-Vlanif3005] ip address 10.125.97.242 255.255.255.252
[~FW-2-Vlanif3005] quit
[~FW-2] interface Eth-Trunk11
[~FW-2-Eth-Trunk11] portswitch
[~FW-2-Eth-Trunk11] port link-type trunk
[~FW-2-Eth-Trunk11] undo port trunk allow-pass vlan 1
[~FW-2-Eth-Trunk11] port trunk allow-pass vlan 3004 to 3005
[~FW-2-Eth-Trunk11] quit
[~FW-2] ip route-static 0.0.0.0 0.0.0.0 10.125.97.241
[~FW-2] ip route-static 10.132.1.0 255.255.255.0 vpn-instance vsys_1
```

c. 配置SNAT。

配置FW_1。

```
[~FW-1] nat address-group addgrp 0
[~FW-1-address-group-addgrp] mode pat
[~FW-1-address-group-addgrp] section 0 1.2.3.4 1.2.3.4
[~FW-1-address-group-addgrp] quit
[~FW-1] security-policy
[~FW-1-policy-security] rule name rule1
[~FW-1-policy-security-rule-rule1] source-address 10.132.1.0 mask 255.255.255.0
[~FW-1-policy-security-rule-rule1] action permit
[~FW-1] nat-policy
[~FW-1-policy-nat] rule name rule1
[~FW-1-policy-nat-rule-rule1] description SNAT_01
[~FW-1-policy-nat-rule-rule1] source-zone trust
```

```
[~FW-1-policy-nat-rule-rule1] destination-zone untrust
[~FW-1-policy-nat-rule-rule1] source-address 10.132.1.0 mask 255.255.255.0
[~FW-1-policy-nat-rule-rule1] action source-nat address-group addgrp
[~FW-1-policy-nat-rule-rule1] quit
[~FW-1-policy-nat] quit
[~FW-1] firewall import-flow public 1.2.3.4 1.2.3.4 vpn-instance vsys_1
```

配置FW-2。配置过程与参数与FW_1一致，这边不再赘述。

----结束

检查配置结果

1. Underlay配置完成后，需按照下述步骤检查配置结果是否正常。
   a. 检查Underlay路由邻居状态，Loop接口地址能相互ping通，以BorderLeaf_1的显示为例。

      #Border Leaf分别和Server Leaf建立OSPF邻居：

      ```
      <BorderLeaf_1> display ospf peer brief
      OSPF Process 1 with Router ID 10.125.98.1
                      Peer Statistic Information

      Total number of peer(s): 4
       Peer(s) in full state: 4
      --------------------------------------------------------------------------------
       Area Id          Interface                Neighbor id      State
       0.0.0.0          100GE1/0/0               10.125.98.3        Full
       0.0.0.0          100GE1/0/1               10.125.98.4        Full
       0.0.0.0          100GE1/0/2               10.125.98.5        Full
       0.0.0.0          100GE1/0/3               10.125.98.6        Full
      --------------------------------------------------------------------------------
      ```

   b. 检查BGP EVPN邻居状态，以BorderLeaf_1为例。Border Leaf与Server Leaf分别建立BGP EVPN对等体关系：

      ```
      <BorderLeaf_1> display bgp evpn peer
       BGP local router ID        : 10.125.98.1
       Local AS number            : 100
       Total number of peers      : 4
       Peers in established state : 4

        Peer            V        AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
        10.125.98.3     4       100      646     2973     0 08:44:07 Established        3
        10.125.98.4     4       100      651     2983     0 08:43:53 Established        3
        10.125.98.5     4       100      605     2729     0 08:43:50 Established        0
        10.125.98.6     4       100      607     2733     0 08:44:21 Established        0
      ```

2. Overlay配置完成后，需按照下述步骤检查配置结果是否正常。
   a. 检查VXLAN隧道的信息，在ServerLeaf1_2、ServerLeaf2_1、ServerLeaf2_2上执行**display vxlan tunnel**命令可查看到VXLAN隧道的信息。以ServerLeaf1_1显示为例。

      ```
      [~ServerLeaf1_1] display vxlan tunnel
      Number of vxlan tunnel : 1
      Tunnel ID   Source              Destination         State  Type     Uptime
      --------------------------------------------------------------------------------
      4026531841  10.125.99.2         10.125.99.3         up     dynamic  0032h21m
      ```

      说明

      ServerLeaf1_2、ServerLeaf2_1、ServerLeaf2_2上的二层子接口需要有服务器接入后，才可以查看到隧道状态为up。在无服务器接入的情况下，会因为没有IRB类型路由的发布，导致无法查看到VXLAN隧道状态。

   b. 检查BGP EVPN邻居状态，以BorderLeaf_1为例。Border Leaf与Server Leaf分别建立BGP EVPN对等体关系：

      ```
      <BorderLeaf_1> display bgp evpn peer
       BGP local router ID        : 10.125.98.1
      ```

```
Local AS number           : 100
Total number of peers     : 4
Peers in established state : 4

 Peer            V        AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
 10.125.98.3     4       100     646     2973     0 08:44:07 Established        3
 10.125.98.4     4       100     651     2983     0 08:43:53 Established        3
 10.125.98.5     4       100     605     2729     0 08:43:50 Established        0
 10.125.98.6     4       100     607     2733     0 08:44:21 Established        0
```

配置脚本

- BorderLeaf_1的配置脚本

```
# --------BorderLeaf_1与Server Leaf互联接口地址
interface 100GE1/0/0
 description "to ServerLeaf1_1"
 undo portswitch
 ip address 10.125.97.21 255.255.255.252
#
interface 100GE1/0/1
 description "to ServerLeaf1_2"
 undo portswitch
 ip address 10.125.97.25 255.255.255.252
#
interface 100GE1/0/2
 description "to ServerLeaf2_1"
 undo portswitch
 ip address 10.125.97.29 255.255.255.252
#
interface 100GE1/0/3
 description "to ServerLeaf2_2"
 undo portswitch
 ip address 10.125.97.33 255.255.255.252
#

#--------BorderLeaf_1与FW互联的管理链路接口地址
vlan 11
interface Vlanif11
 description "to FW1-2"
 ip address 10.125.97.57 255.255.255.248
 mac-address 00e0-fc00-0101
#

#-------静态Bypass VXLAN隧道源IP地址
vlan 100
 m-lag peer-link reserved
#
interface Vlanif100
 ip address 10.125.96.1 255.255.255.252
 reserved for vxlan bypass
#
ip route-static 10.135.98.2 32 10.125.96.2 preference 1
#

#--------BorderLeaf_1的Loopback接口地址
interface LoopBack0
 description VTEP
 ipv6 enable    //当需要使用IPv6时，配置使能IPv6
 ip address 10.125.99.1 255.255.255.255
#
interface LoopBack1
 description AC-MGMT/DFS-GROUP/ROUTER-ID
 ip address 10.125.98.1 255.255.255.255
#
interface LoopBack2
 description Bypass VXLAN
 ip address 10.135.98.1 255.255.255.255
#
```

```
#--------BorderLeaf_1的NVE接口
interface Nve1
 source 10.125.99.1
 mac-address 00e0-fc00-0101
 pip-source 10.135.98.1 peer 10.135.98.2
#

#--------BorderLeaf_1的M-LAG模式
stp mode rstp
stp v-stp enable   //配置V-STP方式的M-LAG
#

#--------M-LAG的DFS组
dfs-group 1
 priority 150    //配置DFS优先级高于对端，默认是100
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
 dual-actice detection source ip 10.125.98.1
 consistency-check enable mode loose    //使能M-LAG配置一致性检查，模式为松散模式
#

#--------peer-link
interface Eth-Trunk0
 trunkport 100GE 4/0/47
 trunkport 100GE 1/0/23
 mode lacp-static
 peer-link 1
 lacp mixed-rate link enable   //使能不同速率的接口加入LACP模式的Eth-Trunk接口后可转发数据报文功
能
#
interface 100GE1/0/23
 distribute-weight 4          //配置100GE成员接口的负载分担权重值为4，100GE成员接口的权重值保持默
认值1
#

#--------M-LAG接口：BorderLeaf_1与防火墙互联
interface Eth-Trunk1    //配置与FW主设备互联管理口
 description FW_1_MGMT
 trunkport 100GE 4/0/41
 port default vlan 11
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2    //配置与FW备设备互联管理口
 description FW_2_MGMT
 trunkport 100GE 4/0/42
 port default vlan 11
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk11   //配置与FW主设备互联业务口
 trunkport 100GE 4/0/43
 port link-type trunk
 undo port trunk allow-pass vlan 1
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 3
#
interface Eth-Trunk12   //配置与FW备设备互联业务口
 trunkport 100GE 4/0/44
 port link-type trunk
 undo port trunk allow-pass vlan 1
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 4
#

#--------OSPF路由打通VXLAN Underlay路由
bfd    //全局使能BFD功能
```

```
#
ospf 1 router-id 10.125.98.1
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 500 min-rx-interval 500 detect-multiplier 3
 lsa-arrival-interval intelligent-timer 50 50 50     //设置OSPF LSA接收的时间间隔，优化收敛时间
 area 0.0.0.0
  network 10.125.97.20 0.0.0.3
  network 10.125.97.24 0.0.0.3
  network 10.125.97.28 0.0.0.3
  network 10.125.97.32 0.0.0.3   //分别建立与4台Server Leaf设备的路由邻居
  network 10.125.97.56 0.0.0.7   //发布防火墙带内管理地址，打通路由
  network 10.125.98.1 0.0.0.0
  network 10.125.99.1 0.0.0.0    //发布Loopback地址
#
interface 100GE1/0/0
 description "to ServerLeaf1_1"
 undo portswitch
 ip address 10.125.97.21 255.255.255.252
 ospf network-type p2p   //配置与Server Leaf互联OSPF接口的网络类型为P2P
#
interface 100GE1/0/1
 description "to ServerLeaf1_2"
 undo portswitch
 ip address 10.125.97.25 255.255.255.252
 ospf network-type p2p
#
interface 100GE1/0/2
 description "to ServerLeaf2_1"
 undo portswitch
 ip address 10.125.97.29 255.255.255.252
 ospf network-type p2p
#
interface 100GE1/0/3
 description "to ServerLeaf2_2"
 undo portswitch
 ip address 10.125.97.33 255.255.255.252
 ospf network-type p2p
#

#--------OSPF网络故障收敛性能优化
interface 100GE1/0/0
 ospf peer hold-max-cost timer 300000   //所有Spine和Leaf配置OSPF邻居建立后在本地设备的LSA中保持最大开销值的时间300s，源于240s的M-LAG延迟UP时间（同时overlay路由收敛）+ 60s的设备表项同步时间
#
interface 100GE1/0/1
 ospf peer hold-max-cost timer 300000
#
interface 100GE1/0/2
 ospf peer hold-max-cost timer 300000
#
interface 100GE1/0/3
 ospf peer hold-max-cost timer 300000
#

#--------BGP EVPN
evpn-overlay enable     //使能EVPN作为VXLAN的控制平面
#
bgp 100
 router-id 10.125.98.1
 advertise lowest-priority all-address-family peer-up delay 360  //在邻居状态由Down到Up时将BGP路由的优先级调整为最低优先级；路由延时发布，解决回切场景丢包时间长问题
 undo default ipv4-unicast       //关闭BGP IPv4单播邻居，降低设备负荷
 group ServerLeaf internal       //配置Server Leaf的对等体组并加入相应对等体。
 peer 10.125.98.3 group ServerLeaf
 peer 10.125.98.4 group ServerLeaf
 peer 10.125.98.5 group ServerLeaf
 peer 10.125.98.6 group ServerLeaf
 peer ServerLeaf connect-interface LoopBack1    //指定发送BGP报文的源接口
```

```
 #
 l2vpn-family evpn          //使能并进入BGP-EVPN地址族视图
  undo policy vpn-target       //配置去使能对接收到的EVPN路由使能VPN-Target过滤功能
  peer ServerLeaf enable
  peer 10.125.98.3 group ServerLeaf
  peer 10.125.98.4 group ServerLeaf
  peer 10.125.98.5 group ServerLeaf
  peer 10.125.98.6 group ServerLeaf
  peer ServerLeaf advertise irb   //配置向BGP EVPN对等体组Server Leaf发布irb和irbv6路由
  peer ServerLeaf advertise irbv6
  peer ServerLeaf reflect-client   //配置路由反射器功能
#

#
ip route-static 1.2.3.4 255.255.255.0 10.125.97.242
#
```

- BorderLeaf_2的配置脚本

```
# --------BorderLeaf_2与Server Leaf互联接口地址
interface 100GE1/0/0
 description "to ServerLeaf1_1"
 undo portswitch
 ip address 10.125.97.37 255.255.255.252
#
interface 100GE1/0/1
 description "to ServerLeaf1_2"
 undo portswitch
 ip address 10.125.97.41 255.255.255.252
#
interface 100GE1/0/2
 description "to ServerLeaf2_1"
 undo portswitch
 ip address 10.125.97.45 255.255.255.252
#
interface 100GE1/0/3
 description "to ServerLeaf2_2"
 undo portswitch
 ip address 10.125.97.49 255.255.255.252
#

#--------BorderLeaf_2与FW互联的管理链路接口地址
vlan 11
interface Vlanif11
 description "to FW1-2"
 ip address 10.125.97.57 255.255.255.248
 mac-address 00e0-fc00-0101
#

#-------静态Bypass VXLAN隧道源IP地址
vlan 100
 m-lag peer-link reserved
#
interface Vlanif100
 ip address 10.125.96.2 255.255.255.252
 reserved for vxlan bypass
#
ip route-static 10.135.98.1 32 10.125.96.1 preference 1
#

#--------BorderLeaf_2的Loopback接口地址
interface LoopBack0
 description VTEP
 ipv6 enable   //当需要使用IPv6时，配置使能IPv6
 ip address 10.125.99.1 255.255.255.255
#
interface LoopBack1
 description AC-MGMT/DFS-GROUP/ROUTER-ID
 ip address 10.125.98.2 255.255.255.255
#
interface LoopBack2
```



```
 description Bypass VXLAN
 ip address 10.135.98.2 255.255.255.255
#

#--------BorderLeaf_2的NVE接口
interface Nve1
 source 10.125.99.1
 mac-address 00e0-fc00-0101
 pip-source 10.135.98.2 peer 10.135.98.1
#

#--------BorderLeaf_2的M-LAG模式
stp mode rstp
stp v-stp enable
#

#--------M-LAG的DFS组
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-actice detection source ip 10.125.98.2
 consistency-check enable mode loose
#

#--------peer-link
interface Eth-Trunk0
 trunkport 100GE 4/0/47
 trunkport 100GE 1/0/23
 mode lacp-static
 peer-link 1
 lacp mixed-rate link enable
#
interface 100GE1/0/23
 distribute-weight 4
#

#--------M-LAG接口：BorderLeaf_2与防火墙互联
interface Eth-Trunk1
 description FW_1_MGMT
 trunkport 100GE 4/0/41
 port default vlan 11
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk2
 description FW_2_MGMT
 trunkport 100GE 4/0/42
 port default vlan 11
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk11
 trunkport 100GE 4/0/43
 port link-type trunk
 undo port trunk allow-pass vlan 1
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 3
#
interface Eth-Trunk12
 trunkport 100GE 4/0/44
 port link-type trunk
 undo port trunk allow-pass vlan 1
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 4
#

#--------OSPF路由打通VXLAN Underlay路由
```

```
bfd
#
ospf 1 router-id 10.125.98.2
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 500 min-rx-interval 500 detect-multiplier 3  //仅组网中全部为支持硬件BFD的款型时，配置500ms*3；其余保持默认配置1000ms*3
 lsa-arrival-interval intelligent-timer 50 50 50
 area 0.0.0.0
  network 10.125.97.36 0.0.0.3
  network 10.125.97.40 0.0.0.3
  network 10.125.97.44 0.0.0.3
  network 10.125.97.48 0.0.0.3
  network 10.125.97.56 0.0.0.7
  network 10.125.98.2 0.0.0.0
  network 10.125.99.1 0.0.0.0
#
interface 100GE1/0/0
 description "to ServerLeaf1_1"
 undo portswitch
 ip address 10.125.97.37 255.255.255.252
 ospf network-type p2p
#
interface 100GE1/0/1
 description "to ServerLeaf1_2"
 undo portswitch
 ip address 10.125.97.41 255.255.255.252
 ospf network-type p2p
#
interface 100GE1/0/2
 description "to ServerLeaf2_1"
 undo portswitch
 ip address 10.125.97.45 255.255.255.252
 ospf network-type p2p
#
interface 100GE1/0/3
 description "to ServerLeaf2_2"
 undo portswitch
 ip address 10.125.97.49 255.255.255.252
 ospf network-type p2p
#

#--------OSPF网络故障收敛性能优化
interface 100GE1/0/0
 ospf peer hold-max-cost timer 300000   //所有Spine和Leaf配置OSPF邻居建立后在本地设备的LSA中保持最大开销值的时间300s，源于240s的M-LAG延迟UP时间（同时overlay路由收敛）+ 60s的设备表项同步时间
#
interface 100GE1/0/1
 ospf peer hold-max-cost timer 300000
#
interface 100GE1/0/2
 ospf peer hold-max-cost timer 300000
#
interface 100GE1/0/3
 ospf peer hold-max-cost timer 300000
#

#--------BGP EVPN
evpn-overlay enable
#
bgp 100
 router-id 10.125.98.2
 advertise lowest-priority all-address-family peer-up delay 360
 undo default ipv4-unicast
 group ServerLeaf internal
 peer 10.125.98.3 group ServerLeaf
 peer 10.125.98.4 group ServerLeaf
 peer 10.125.98.5 group ServerLeaf
 peer 10.125.98.6 group ServerLeaf
```



```
 peer ServerLeaf connect-interface LoopBack1
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer ServerLeaf enable
  peer 10.125.98.3 group ServerLeaf
  peer 10.125.98.4 group ServerLeaf
  peer 10.125.98.5 group ServerLeaf
  peer 10.125.98.6 group ServerLeaf
  peer ServerLeaf advertise irb
  peer ServerLeaf advertise irbv6
  peer ServerLeaf reflect-client
#

#
ip route-static 1.2.3.4 255.255.255.0 10.125.97.242
#
```

- ServerLeaf1_1的配置脚本

```
#--------ServerLeaf1_1与Border Leaf的互联接口地址
interface 100GE1/0/1
 description "to BorderLeaf_1"
 undo portswitch
 ip address 10.125.97.22 255.255.255.252
 ospf network-type p2p
#
interface 100GE1/0/2
 description "to BorderLeaf_2"
 undo portswitch
 ip address 10.125.97.38 255.255.255.252
 ospf network-type p2p
#

#--------Loopback接口地址
interface LoopBack0
 description VTEP
 ipv6 enable     //当需要使用IPv6时，配置使能IPv6
 ip address 10.125.99.2 255.255.255.255
#
interface LoopBack1
 description AC-MGMT/DFS-GROUP/ROUTER-ID
 ip address 10.125.98.3 255.255.255.255
#
interface LoopBack2
 description Bypass VXLAN
 ip address 10.135.98.3 255.255.255.255
#

#-------静态Bypass VXLAN隧道源IP地址
vlan 100
 m-lag peer-link reserved
#
interface Vlanif100
 ip address 10.125.96.5 255.255.255.252
 reserved for vxlan bypass
#
ip route-static 10.135.98.4 32 10.125.96.6 preference 1
#

#--------NVE接口VTEP IP和虚拟MAC地址
interface Nve1
 source 10.125.99.2
 mac-address 00e0-fc00-0102
 pip-source 10.135.98.3 peer 10.135.98.4 bypass
#

#--------M-LAG模式
stp mode rstp
stp v-stp enable   //配置V-STP方式的M-LAG
stp tc-protection              //使能设备对TC类型BPDU报文的保护功能
```

```
stp bpdu-protection            //使能设备的BPDU保护功能
arp ip-conflict-detect enable  //使能设备的IP地址冲突检测的功能
#

#--------M-LAG的DFS组
dfs-group 1
 priority 150    //配置DFS优先级高于对端，默认是100
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-actice detection source ip 10.125.98.3
 consistency-check enable mode loose    //使能M-LAG配置一致性检查，模式为松散模式
#

#--------peer-link
interface Eth-Trunk0
 trunkport 100GE 1/0/5 to 1/0/6
 mode lacp-static
 peer-link 1
#

#--------业务服务器以负载分担方式接入
interface eth-trunk 10
 port link-type trunk
 undo port trunk allow-pass vlan 1
 trunkport 100GE 1/0/1
 dfs-group 1 m-lag 10
 mode lacp-static
 stp edged-port enable    //配置边缘端口
#
interface 100GE1/0/1   //服务器接入端口
 storm suppression unknown-unicast 5   //配置未知单播抑制，经验值为100GE端口的5%带宽，建议业务
端口都部署
 storm suppression multicast packets 1000  //配置组播报文抑制，经验值为1000pps。
 storm suppression broadcast packets 1000  //配置广播报文抑制，经验值为1000pps，建议业务端口都部
署
#

#--------服务器以主备方式接入
interface 100GE1/0/2
 port link-type trunk
 undo port trunk allow-pass vlan 1  //不放通VLAN1，防止成环
 storm suppression unknown-unicast 5  //配置未知单播抑制，经验值为100GE端口的5%带宽，建议业务
端口都部署
 storm suppression multicast packets 1000  //配置组播报文抑制，经验值为1000pps
 storm suppression broadcast packets 1000  //配置广播报文抑制，经验值为1000pps，建议业务端口都部
署
 stp edged-port enable
#

#--------monitor-link关联上行接口和下行接口
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 timer recover-time 60   //配置回切时间，防止上行故障回切丢包。
#

# --------OSPF路由
bfd            //全局使能BFD功能
#
ospf 1 router-id 10.125.98.3
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 500 min-rx-interval 500 detect-multiplier 3
 lsa-arrival-interval intelligent-timer 50 50 50   //设置OSPF LSA接收的时间间隔，优化收敛时间
 area 0.0.0.0
  network 10.125.97.20 0.0.0.3
  network 10.125.97.36 0.0.0.3   //分别建立与2台Border Leaf设备的路由邻居
  network 10.125.98.3 0.0.0.0
  network 10.125.99.2 0.0.0.0   //发布Loopback地址
```

```
#

#--------网络故障收敛性能优化
interface 100GE1/0/2
 ospf peer hold-max-cost timer 300000    //所有Spine和Leaf配置OSPF邻居建立后在本地设备的LSA中保持最大开销值的时间300s，源于240s的M-LAG延迟UP时间（同时overlay路由收敛）+ 60s的设备表项同步时间
#
interface 100GE1/0/3
 ospf peer hold-max-cost timer 300000
#

#--------BGP EVPN
evpn-overlay enable   //使能EVPN作为VXLAN的控制平面
#
bgp 100
 router-id 10.125.98.3
 undo default ipv4-unicast   //关闭BGP IPv4单播邻居，降低设备负荷
 group BorderLeaf internal    //配置BorderLeaf的对等体组并加入相应对等体
 peer 10.125.98.1 group BorderLeaf
 peer 10.125.98.2 group BorderLeaf
 peer Spine connect-interface LoopBack1   //指定发送BGP报文的源接口
 #
 l2vpn-family evpn
  policy vpn-target
  peer BorderLeaf enable
  peer 10.125.98.1 group BorderLeaf
  peer 10.125.98.2 group BorderLeaf
  peer Spine advertise irb
  peer Spine advertise irbv6
#

#--------Overlay配置
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:2
  vpn-target 100:5010 export-extcommunity evpn
  vpn-target 100:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:2
  vpn-target 100:10 export-extcommunity
  vpn-target 100:5010 export-extcommunity
  vpn-target 100:10 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ip address 10.1.1.1 255.255.255.0
 arp broadcast-detect enable
 mac-address 00e0-fc00-0102
 vxlan anycast-gateway enable
 arp collect host enable
#
```

- ServerLeaf1_2的配置脚本

```
#--------ServerLeaf1_2与Border Leaf的互联接口地址
interface 100GE1/0/1
 description "to BorderLeaf_1"
 undo portswitch
 ip address 10.125.97.26 255.255.255.252
 ospf network-type p2p
#
interface 100GE1/0/2
 description "to BorderLeaf_2"
 undo portswitch
 ip address 10.125.97.42 255.255.255.252
 ospf network-type p2p
```



```
#

#--------Loopback接口地址
interface LoopBack0
 description VTEP
 ipv6 enable    //当需要使用IPv6时，配置使能IPv6
 ip address 10.125.99.2 255.255.255.255
#
interface LoopBack1
 description AC-MGMT/DFS-GROUP/ROUTER-ID
 ip address 10.125.98.4 255.255.255.0
#
interface LoopBack2
 description Bypass VXLAN
 ip address 10.135.98.4 255.255.255.255
#

#-------静态Bypass VXLAN隧道源IP地址
vlan 100
 m-lag peer-link reserved
#
interface Vlanif100
 ip address 10.125.96.6 255.255.255.252
 reserved for vxlan bypass
#
ip route-static 10.135.98.3 32 10.125.96.5 preference 1
#

#--------NVE接口VTEP IP和虚拟MAC地址
interface Nve1
 source 10.125.99.2
 mac-address 00e0-fc00-0102
 pip-source 10.135.98.4 peer 10.135.98.3 bypass
#

#--------M-LAG模式
stp mode rstp
stp v-stp enable   //配置V-STP方式的M-LAG
stp tc-protection              //使能设备对TC类型BPDU报文的保护功能
stp bpdu-protection            //使能设备的BPDU保护功能
arp ip-conflict-detect enable  //使能设备的IP地址冲突检测的功能
#

#--------M-LAG的DFS组
dfs-group 1
 source ip 10.125.98.4
 consistency-check enable mode loose
#

#--------peer-link
interface Eth-Trunk0
 trunkport 100GE 1/0/5 to 1/0/6
 mode lacp-static
 peer-link 1
#

#--------业务服务器以负载分担方式接入
interface eth-trunk 10
  port link-type trunk
  undo port trunk allow-pass vlan 1
  trunkport 100GE 1/0/1
  dfs-group 1 m-lag 10
  mode lacp-static
  stp edged-port enable
#
interface 100GE1/0/1
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
```

```
#

#--------服务器以主备方式接入
interface 100GE1/0/2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 stp edged-port enable
#

#--------monitor-link关联上行接口和下行接口
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 timer recover-time 60
#

# --------OSPF路由
bfd
#
ospf 1 router-id 10.125.98.4
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 500 min-rx-interval 500 detect-multiplier 3   //仅组网中全部为支持硬件BFD的款型时，配置500ms*3；其余保持默认配置1000ms*3
 lsa-arrival-interval intelligent-timer 50 50 50   //优化三层架构，两台物理设备之间多路ECMP情况的OSPF收敛时间
 area 0.0.0.0
  network 10.125.97.24 0.0.0.3
  network 10.125.97.40 0.0.0.3
  network 10.125.98.4 0.0.0.0
  network 10.125.99.2 0.0.0.0
#

#--------网络故障收敛性能优化
interface 100GE1/0/2
 ospf peer hold-max-cost timer 300000    //所有Spine和Leaf配置OSPF邻居建立后在本地设备的LSA中保持最大开销值的时间300s，源于240s的M-LAG延迟UP时间（同时overlay路由收敛）+ 60s的设备表项同步时间
#
interface 100GE1/0/3
 ospf peer hold-max-cost timer 300000
#

#--------BGP EVPN
evpn-overlay enable
#
bgp 100
 router-id 10.125.98.4
 undo default ipv4-unicast
 group Spine internal
 peer 10.125.98.1 group Spine
 peer 10.125.98.2 group Spine
 peer Spine connect-interface LoopBack1
 #
 l2vpn-family evpn
  policy vpn-target
  peer Spine enable
  peer 10.125.98.1 group Spine
  peer 10.125.98.2 group Spine
  peer Spine advertise irb
  peer Spine advertise irbv6
#

#
ip vpn-instance vpn1
 ipv4-family
```

```
 route-distinguisher 20:4
 vpn-target 100:5010 export-extcommunity evpn
 vpn-target 100:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:4
  vpn-target 100:10 export-extcommunity
  vpn-target 100:5010 export-extcommunity
  vpn-target 100:10 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ip address 10.1.1.1 255.255.255.0
 arp broadcast-detect enable
 mac-address 00e0-fc00-0102
 vxlan anycast-gateway enable
 arp collect host enable
#
```

- ServerLeaf2_1、ServerLeaf2_2的配置脚本与ServerLeaf1_1、ServerLeaf1_2类似，不再赘述。
- FW-1的配置脚本

```
#--------vsys关键配置
interface Vlanif3004
 ip binding vpn-instance vsys_1
 ip address 10.125.97.242
255.255.255.252
#
ip route-static 0.0.0.0 0.0.0.0 public
ip route-static 10.132.1.0 255.255.255.0
10.125.97.241
#

#--------public关键配
置
vsys name vsys_1 1
 assign vlan 3004
#
ip vpn-instance vsys_1
 ipv4-family
#
interface Vlanif3005
 ip address 10.125.97.242
255.255.255.252
#
interface Eth-Trunk11
 portswitch
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3004 to
3005
#
ip route-static 0.0.0.0 0.0.0.0 10.125.97.241
ip route-static 10.132.1.0 255.255.255.0 vpn-instance
vsys_78b845f994647ad9
#

#--------SNAT的配
置
nat address-group addgrp 0
 mode pat
 section 0 1.2.3.4 1.2.3.4
#
security-policy
 rule name 20191228113827
  source-address 10.132.1.0 mask 255.255.255.0
```



```
 action permit
#
nat-policy
 rule name rule1
  description SNAT_01
  source-zone trust
  destination-zone untrust
  source-address 10.132.1.0 mask 255.255.255.0
  action source-nat address-group addgrp
#
 firewall import-flow public 1.2.3.4 1.2.3.4 vpn-instance vsys_1
#
```

- FW-2的配置脚本

```
#--------vsys关键配
置
interface Vlanif3004
 ip binding vpn-instance vsys_1
 ip address 10.125.97.242
255.255.255.252
#
ip route-static 0.0.0.0 0.0.0.0 public
ip route-static 10.132.1.0 255.255.255.0
10.125.97.241
#

#--------public关键配置
#
vsys name vsys_1 1
 assign vlan 3004
#
ip vpn-instance vsys_1
 ipv4-family
#
interface Vlanif3005
 ip address 10.125.97.242
255.255.255.252
#
interface Eth-Trunk11
 portswitch
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 3004 to
3005
#
ip route-static 0.0.0.0 0.0.0.0 10.125.97.241
ip route-static 10.132.1.0 255.255.255.0 vpn-instance vsys_1
#

# --------SNAT的配置
nat address-group addgrp 0
 mode pat
 section 0 1.2.3.4 1.2.3.4
#
security-policy
 rule name 20191228113827
  source-address 10.132.1.0 mask 255.255.255.0
  action permit
#
nat-policy
 rule name rule1
  description SNAT_01
  source-zone trust
  destination-zone untrust
  source-address 10.132.1.0 mask 255.255.255.0
  action source-nat address-group addgrp
#
 firewall import-flow public 1.2.3.4 1.2.3.4 vpn-instance vsys_1
#
```

1.6 配置 NFVI 分布式网关示例（非对称式）

适用产品和版本

- V300R021C10及之后版本：CE8851、CE6866可以部署为该组网中的Leaf设备，CE16800（除X系列单板外）、CE8851可以部署为该组网中的Spine设备。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

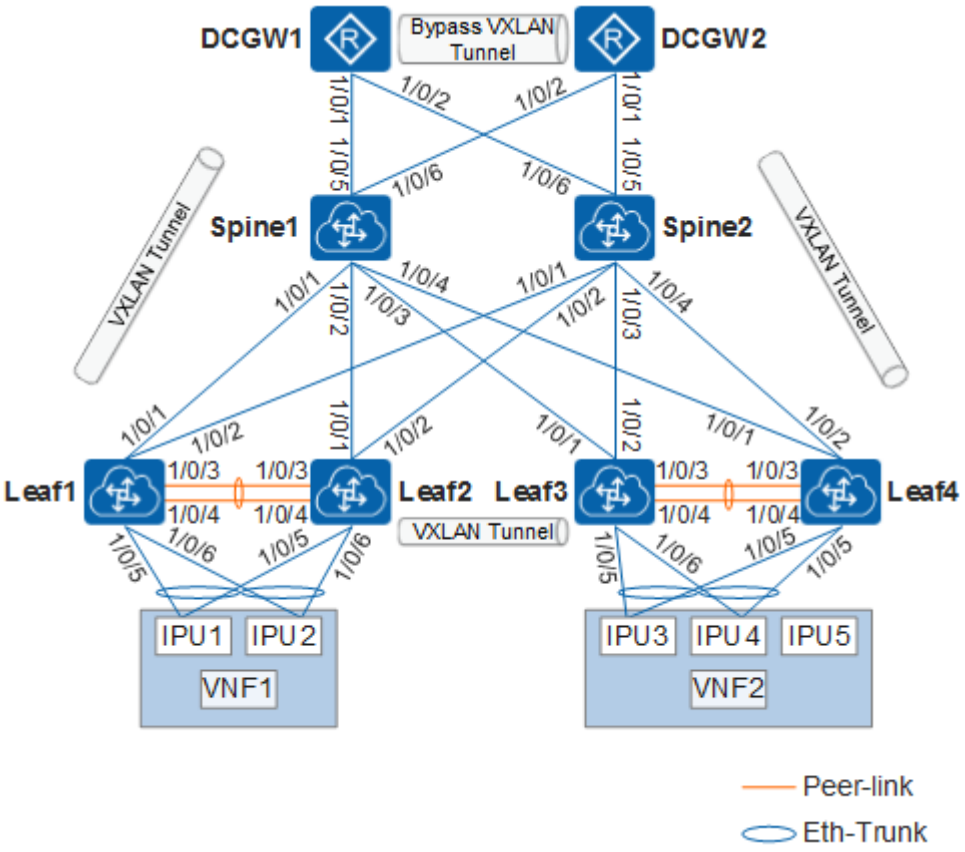
NFVI电信云解决方案是DCI（Data Center Interconnect）+DCN（Data Communication Network）的组网方案。其中大量手机业务流量会进入DCN网络并访问DCN网络内的vUGW与vMSE。经过vUGW与vMSE的处理后，这些手机业务流量再次从DCN网络转发出去，继续访问Internet中的目的设备。同样目的设备发往手机的回应流量亦要经历该过程。为了实现上述功能，并且确保手机业务流量在DCN网络内部可以实现负载均衡，则需要在DCN网络内部部署NFVI分布式网关功能。

如**配置IPv4 NFVI分布式网关组网图**所示，该组网为NFVI分布式网关的组网示意图。其中DCGW1和DCGW2为DCN网络的边界网关，可以和外部网络交换Internet路由。Leaf用于接入VNF（Virtualized Network Function）。VNF1和VNF2作为虚拟化网元可以分别部署并实现vUGW和vMSE的功能，并通过IPU（Interface Process Unit）与Leaf连接。

该组网可以看成分布式网关功能和VXLAN双活网关功能拼接组成：

- DCGW1和DCGW2上部署VXLAN双活网关功能，即DCGW1和DCGW2之间建立Bypass VXLAN隧道，同时DCGW1和DCGW2共同使用一个虚拟的Anycast VTEP地址分别与Leaf建立VXLAN隧道。
- Spine作为透传节点，连接DCGW和Leaf。
- Leaf上部署分布式网关功能并在两个M-LAG组间创建VXLAN隧道。

图 1-6 配置 NFVI 分布式网关组网图



说明

上图中“1/0/1”为接口编号，接口速率为100GE，即“1/0/1”表示接口“100GE1/0/1”。其他接口类似。

表 1-9 接口地址表

设备名称	接口	IP地址	设备名称	接口	IP地址
DCGW1	100GE1/0/1	192.168.9.2/24	DCGW2	100GE1/0/1	192.168.11.2/24
	100GE1/0/2	192.168.12.2/24		100GE1/0/2	192.168.10.2/24
	Loopback0	9.9.9.9/32		Loopback0	10.10.10.10/32
	Loopback1	11.11.11.11/32		Loopback1	11.11.11.11/32
Spine1	100GE1/0/1	192.168.1.1/24	Spine2	100GE1/0/1	192.168.5.1/24

设备名称	接口	IP地址	设备名称	接口	IP地址
	100GE1/0/2	192.168.2.1/24		100GE1/0/2	192.168.6.1/24
	100GE1/0/3	192.168.3.1/24		100GE1/0/3	192.168.7.1/24
	100GE1/0/4	192.168.4.1/24		100GE1/0/4	192.168.8.1/24
	100GE1/0/5	192.168.9.1/24		100GE1/0/5	192.168.11.1/24
	100GE1/0/6	192.168.10.1/24		100GE1/0/6	192.168.12.1/24
	Loopback0	7.7.7.7/32		Loopback0	8.8.8.8/32
Leaf1	100GE1/0/1	192.168.1.2/24	Leaf2	100GE1/0/1	192.168.2.2/24
	100GE1/0/2	192.168.5.2/24		100GE1/0/2	192.168.6.2/24
	Loopback0	1.1.1.1/32		Loopback0	2.2.2.2/32
	Loopback1	5.5.5.5/32		Loopback1	5.5.5.5/32
	Loopback2	12.12.12.12/32		Loopback2	13.13.13.13/32
Leaf3	100GE1/0/1	192.168.3.2/24	Leaf4	100GE1/0/1	192.168.4.2/24
	100GE1/0/2	192.168.7.2/24		100GE1/0/2	192.168.8.2/24
	Loopback0	3.3.3.3/32		Loopback0	4.4.4.4/32
	Loopback1	6.6.6.6/32		Loopback1	6.6.6.6/32
	Loopback2	14.14.14.14/32		Loopback2	15.15.15.15/32
IPU1	-	IPv4: 10.1.1.2/24 IPv6: fc00:1::2/64	IPU2	-	IPv4: 10.2.1.2/24 IPv6: fc00:2::2/64
IPU3	-	IPv4: 10.3.1.2/24 IPv6: fc00:3::2/64	IPU4	-	IPv4: 10.4.1.2/24 IPv6: fc00:4::2/64

设备名称	接口	IP地址	设备名称	接口	IP地址
VNF1	-	IPv4: 172.16.1.1/32 IPv6: 2001:db8:1::1/128	VNF2	-	IPv4: 172.16.2.1/32 IPv6: 2001:db8:2::1/128

配置思路

采用如下的思路配置：

1. 配置路由协议，保证网络三层互通。
2. 配置Leaf组建M-LAG。
3. 配置BGP EVPN，建立VXLAN隧道。
4. 配置Leaf至VNF的静态路由并通过BGP EVPN发布。
5. 配置路由负载分担。

操作步骤

步骤1 配置各接口的IP地址及Loopback接口的地址，并配置路由协议，保证网络三层互通。本示例采用了OSPF路由协议。

配置Leaf1。其他设备的配置与Leaf1类似，这里不再赘述。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf1
[*HUAWEI] commit
[~Leaf1] bfd  //全局使能BFD
[*Leaf1-bfd] quit
[*Leaf1] interface 100ge 1/0/1
[*Leaf1-100GE1/0/1] undo portswitch
[*Leaf1-100GE1/0/1] ip address 192.168.1.2 24
[*Leaf1-100GE1/0/1] ospf network-type p2p
[*Leaf1-100GE1/0/1] ospf peer hold-max-cost timer 800000
[*Leaf1-100GE1/0/1] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/1] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/1] qos phb marking dscp enable
[*Leaf1-100GE1/0/1] quit
[*Leaf1] interface 100ge 1/0/2
[*Leaf1-100GE1/0/2] undo portswitch
[*Leaf1-100GE1/0/2] ip address 192.168.5.2 24
[*Leaf1-100GE1/0/2] ospf network-type p2p
[*Leaf1-100GE1/0/2] ospf peer hold-max-cost timer 800000
[*Leaf1-100GE1/0/2] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/2] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/2] qos phb marking dscp enable
[*Leaf1-100GE1/0/2] quit
[*Leaf1] interface loopback 0
[*Leaf1-LoopBack0] ip address 1.1.1.1 32
[*Leaf1-LoopBack0] quit
[*Leaf1] interface loopback 1
[*Leaf1-LoopBack1] ip address 5.5.5.5 32
[*Leaf1-LoopBack1] quit
[*Leaf1] interface loopback 2
[*Leaf1-LoopBack2] ip address 12.12.12.12 32
[*Leaf1-LoopBack2] quit
```

```
[*Leaf1] ospf 1 router-id 1.1.1.1
[*Leaf1-ospf-1] spf-schedule-interval intelligent-timer 50 50 50
[*Leaf1-ospf-1] lsa-originate-interval intelligent-timer 500 50 100
[*Leaf1-ospf-1] lsa-arrival-interval intelligent-timer 50 50 50
[*Leaf1-ospf-1] bfd all-interfaces enable
[*Leaf1-ospf-1] bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
[*Leaf1-ospf-1] area 0
[*Leaf1-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[*Leaf1-ospf-1-area-0.0.0.0] network 192.168.5.0 0.0.0.255
[*Leaf1-ospf-1-area-0.0.0.0] network 1.1.1.1 0.0.0.0
[*Leaf1-ospf-1-area-0.0.0.0] network 5.5.5.5 0.0.0.0
[*Leaf1-ospf-1-area-0.0.0.0] quit
[*Leaf1-ospf-1] quit
[*Leaf1] commit
```

步骤2 配置Leaf组建M-LAG。

配置Leaf1和Leaf2组建M-LAG。Leaf3和Leaf4的配置与之类似，这里不再赘述。

```
[~Leaf1] stp mode rstp
[*Leaf1] stp v-stp enable
[*Leaf1] dfs-group 1
[*Leaf1-dfs-group-1] dual-active detection source ip 1.1.1.1 peer 2.2.2.2
[*Leaf1-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206  //在组建M-LAG的设备上均需配置本命令，请保持M-LAG组内的认证密码一致
[*Leaf1-dfs-group-1] priority 150
[*Leaf1-dfs-group-1] m-lag up-delay 240 auto-recovery interval 10  //配置M-LAG成员接口上报Up状态的延时时间，以及延迟自动恢复时间
[*Leaf1-dfs-group-1] dual-active detection delay 0  //配置peer-link故障时双主检测链路的检测时间为0，防止peer-link故障时BFD会话震荡
[*Leaf1-dfs-group-1] quit
[*Leaf1] interface eth-trunk 1  //配置peer-link
[*Leaf1-Eth-Trunk1] trunkport 100ge 1/0/3
[*Leaf1-Eth-Trunk1] trunkport 100ge 1/0/4
[*Leaf1-Eth-Trunk1] mode lacp-static
[*Leaf1-Eth-Trunk1] peer-link 1
[*Leaf1-Eth-Trunk1] port vlan exclude 1  //配置peer-link接口不允许通过VLAN1
[*Leaf1-Eth-Trunk1] quit
[*Leaf1] interface 100ge 1/0/3
[*Leaf1-100GE1/0/3] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/3] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/3] quit
[*Leaf1] interface 100ge 1/0/4
[*Leaf1-100GE1/0/4] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/4] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/4] quit
[*Leaf1] interface eth-trunk 10  //配置连接VNF的M-LAG成员接口
[*Leaf1-Eth-Trunk10] trunkport 100ge 1/0/5
[*Leaf1-Eth-Trunk10] mode lacp-static
[*Leaf1-Eth-Trunk10] lacp timeout fast
[*Leaf1-Eth-Trunk10] dfs-group 1 m-lag 1
[*Leaf1-Eth-Trunk10] arp anti-attack rate-limit 200  //指定ARP报文的限速值，每秒允许通过的ARP报文的个数为200
[*Leaf1-Eth-Trunk10] stp edged-port enable
[*Leaf1-Eth-Trunk10] quit
[*Leaf1] interface eth-trunk 11
[*Leaf1-Eth-Trunk11] trunkport 100ge 1/0/6
[*Leaf1-Eth-Trunk11] mode lacp-static
[*Leaf1-Eth-Trunk11] lacp timeout fast
[*Leaf1-Eth-Trunk11] dfs-group 1 m-lag 2
[*Leaf1-Eth-Trunk11] arp anti-attack rate-limit 200  //指定ARP报文的限速值，每秒允许通过的ARP报文的个数为200
[*Leaf1-Eth-Trunk11] stp edged-port enable
[*Leaf1-Eth-Trunk11] quit
[*Leaf1] commit
[~Leaf1] monitor-link group 1
[*Leaf1-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf1-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf1-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf1-mtlk-group1] port eth-trunk 11 downlink 2
```



```
[*Leaf1-mtlk-group1] timer recover-time 60
[*Leaf1-mtlk-group1] quit
[*Leaf1] interface 100ge 1/0/5
[*Leaf1-100GE1/0/5] storm suppression unknown-unicast 5  //配置接入交换机端口未知单播抑制功能
[*Leaf1-100GE1/0/5] storm suppression multicast packets 1000  //配置接入交换机端口未知组播抑制功能
[*Leaf1-100GE1/0/5] storm suppression broadcast packets 1000  //配置接入交换机端口广播抑制功能
[*Leaf1-100GE1/0/5] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/5] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/5] quit
[*Leaf1] interface 100ge 1/0/6
[*Leaf1-100GE1/0/6] storm suppression unknown-unicast 5
[*Leaf1-100GE1/0/6] storm suppression multicast packets 1000
[*Leaf1-100GE1/0/6] storm suppression broadcast packets 1000
[*Leaf1-100GE1/0/6] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/6] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/6] quit
[*Leaf1] commit
[~Leaf2] stp mode rstp
[*Leaf2] stp v-stp enable
[*Leaf2] dfs-group 1
[*Leaf2-dfs-group-1] dual-active detection source ip 2.2.2.2 peer 1.1.1.1
[*Leaf2-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Leaf2-dfs-group-1] m-lag up-delay 240 auto-recovery interval 10
[*Leaf2-dfs-group-1] dual-active detection delay 0
[*Leaf2-dfs-group-1] quit
[*Leaf2] interface eth-trunk 1
[*Leaf2-Eth-Trunk1] trunkport 100ge 1/0/3
[*Leaf2-Eth-Trunk1] trunkport 100ge 1/0/4
[*Leaf2-Eth-Trunk1] mode lacp-static
[*Leaf2-Eth-Trunk1] peer-link 1
[*Leaf2-Eth-Trunk1] port vlan exclude 1   //配置peer-link接口不允许通过VLAN1
[*Leaf2-Eth-Trunk1] quit
[*Leaf2] interface eth-trunk 10
[*Leaf2-Eth-Trunk10] trunkport 100ge 1/0/5
[*Leaf2-Eth-Trunk10] mode lacp-static
[*Leaf2-Eth-Trunk10] lacp timeout fast
[*Leaf2-Eth-Trunk10] dfs-group 1 m-lag 1
[*Leaf2-Eth-Trunk10] arp anti-attack rate-limit 200
[*Leaf2-Eth-Trunk10] stp edged-port enable
[*Leaf2-Eth-Trunk10] quit
[*Leaf2] interface eth-trunk 11
[*Leaf2-Eth-Trunk11] trunkport 100ge 1/0/6
[*Leaf2-Eth-Trunk11] mode lacp-static
[*Leaf2-Eth-Trunk11] lacp timeout fast
[*Leaf2-Eth-Trunk11] dfs-group 1 m-lag 2
[*Leaf2-Eth-Trunk11] arp anti-attack rate-limit 200
[*Leaf2-Eth-Trunk11] stp edged-port enable
[*Leaf2-Eth-Trunk11] quit
[*Leaf2] commit
[~Leaf2] monitor-link group 1
[*Leaf2-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf2-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf2-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf2-mtlk-group1] port eth-trunk 11 downlink 2
[*Leaf2-mtlk-group1] timer recover-time 60
[*Leaf2-mtlk-group1] quit
[*Leaf2] interface 100ge 1/0/5
[*Leaf2-100GE1/0/5] storm suppression unknown-unicast 5
[*Leaf2-100GE1/0/5] storm suppression multicast packets 1000
[*Leaf2-100GE1/0/5] storm suppression broadcast packets 1000
[*Leaf2-100GE1/0/5] quit
[*Leaf2] interface 100ge 1/0/6
[*Leaf2-100GE1/0/6] storm suppression unknown-unicast 5
[*Leaf2-100GE1/0/6] storm suppression multicast packets 1000
[*Leaf2-100GE1/0/6] storm suppression broadcast packets 1000
[*Leaf2-100GE1/0/6] quit
[*Leaf2] commit
```

步骤3 在M-LAG设备中配置静态Bypass VXLAN隧道。

在M-LAG双归接入VXLAN的场景中，当下行一条链路发生故障时，业务流量需绕行M-LAG设备之间的Peer-link链路。因此，在该场景下M-LAG设备之间必须配置静态Bypass VXLAN隧道，将绕行的业务流量引导至Peer-link链路上。

下面以Leaf1和Leaf2配置为例，Leaf3、Leaf4的配置与之类似，这里不再赘述，具体配置请参考配置脚本。

配置Leaf1和Leaf2。

```
[~Leaf1] vlan 100  //本VLAN不能划分给其他业务使用，本例中以100举例
[*Leaf1-vlan100] m-lag peer-link reserved  //仅允许peer-link加入到该VLAN
[*Leaf1-vlan100] quit
[*Leaf1] interface vlanif 100
[*Leaf1-Vlanif100] reserved for vxlan bypass  //指定peer-link接口上VLANIF的IPv4地址只给Bypass VXLAN隧道使用
[*Leaf1-Vlanif100] ip address 10.10.10.1 30  //配置静态Bypass VXLAN隧道的源端IPv4地址
[*Leaf1-Vlanif100] quit
[*Leaf1] ip route-static 13.13.13.13 32 10.10.10.2 preference 1  //配置静态路由，打通Bypass VXLAN隧道
[*Leaf1] commit
[*Leaf1] interface nve 1
[*Leaf1-Nve1] pip-source 12.12.12.12 peer 13.13.13.13 bypass  //创建静态Bypass VXLAN隧道，指定源端地址和对端地址
[*Leaf1-Nve1] quit
[*Leaf1] commit
[~Leaf2] vlan 100
[*Leaf2-vlan100] m-lag peer-link reserved
[*Leaf2-vlan100] quit
[*Leaf2] interface vlanif 100
[*Leaf2-Vlanif100] reserved for vxlan bypass
[*Leaf2-Vlanif100] ip address 10.10.10.2 30
[*Leaf2-Vlanif100] quit
[*Leaf2] ip route-static 12.12.12.12 32 10.10.10.1 preference 1
[*Leaf2] commit
[*Leaf2] interface nve 1
[*Leaf2-Nve1] pip-source 13.13.13.13 peer 12.12.12.12 bypass
[*Leaf2-Nve1] quit
[*Leaf2] commit
```

步骤4 配置BGP EVPN，建立VXLAN隧道。

1. 配置BGP EVPN对等体关系。Spine1和Spine2作为leaf节点路由反射器。

   # 配置Spine1。Spine2的配置与Spine1类似，这里不再赘述。

   ```
   [~Spine1] evpn-overlay enable
   [*Spine1] bgp 100
   [*Spine1] router-id 7.7.7.7
   [*Spine1-bgp] peer 1.1.1.1 as-number 100
   [*Spine1-bgp] peer 1.1.1.1 connect-interface LoopBack0
   [*Spine1-bgp] peer 2.2.2.2 as-number 100
   [*Spine1-bgp] peer 2.2.2.2 connect-interface LoopBack0
   [*Spine1-bgp] peer 3.3.3.3 as-number 100
   [*Spine1-bgp] peer 3.3.3.3 connect-interface LoopBack0
   [*Spine1-bgp] peer 4.4.4.4 as-number 100
   [*Spine1-bgp] peer 4.4.4.4 connect-interface LoopBack0
   [*Spine1-bgp] peer 9.9.9.9 as-number 100
   [*Spine1-bgp] peer 9.9.9.9 connect-interface LoopBack0
   [*Spine1-bgp] peer 10.10.10.10 as-number 100
   [*Spine1-bgp] peer 10.10.10.10 connect-interface LoopBack0
   [*Spine1-bgp] l2vpn-family evpn
   [*Spine1-bgp-af-evpn] bestroute add-path path-number 64  //使能BGP ADD-PATH功能
   [*Spine1-bgp-af-evpn] peer 1.1.1.1 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Spine1-bgp-af-evpn] peer 1.1.1.1 reflect-client
   [*Spine1-bgp-af-evpn] peer 1.1.1.1 advertise irb
   [*Spine1-bgp-af-evpn] peer 1.1.1.1 advertise irbv6
   [*Spine1-bgp-af-evpn] peer 1.1.1.1 capability-advertise add-path both
   [*Spine1-bgp-af-evpn] peer 1.1.1.1 advertise add-path path-number 64
   [*Spine1-bgp-af-evpn] peer 2.2.2.2 enable
   ```

```
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 2.2.2.2 advertise irb
[*Spine1-bgp-af-evpn] peer 2.2.2.2 advertise irbv6
[*Spine1-bgp-af-evpn] peer 2.2.2.2 reflect-client
[*Spine1-bgp-af-evpn] peer 2.2.2.2 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 2.2.2.2 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 3.3.3.3 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 3.3.3.3 advertise irb
[*Spine1-bgp-af-evpn] peer 3.3.3.3 advertise irbv6
[*Spine1-bgp-af-evpn] peer 3.3.3.3 reflect-client
[*Spine1-bgp-af-evpn] peer 3.3.3.3 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 3.3.3.3 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 4.4.4.4 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 4.4.4.4 advertise irb
[*Spine1-bgp-af-evpn] peer 4.4.4.4 advertise irbv6
[*Spine1-bgp-af-evpn] peer 4.4.4.4 reflect-client
[*Spine1-bgp-af-evpn] peer 4.4.4.4 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 4.4.4.4 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 9.9.9.9 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 9.9.9.9 advertise irb
[*Spine1-bgp-af-evpn] peer 9.9.9.9 advertise irbv6
[*Spine1-bgp-af-evpn] peer 9.9.9.9 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 9.9.9.9 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 10.10.10.10 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 10.10.10.10 advertise irb
[*Spine1-bgp-af-evpn] peer 10.10.10.10 advertise irbv6
[*Spine1-bgp-af-evpn] peer 10.10.10.10 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 10.10.10.10 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] undo policy vpn-target
[*Spine1-bgp-af-evpn] quit
[*Spine1-bgp] quit
[*Spine1] commit
```

配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] evpn-overlay enable
[*Leaf1] bgp 100
[*Leaf1] router-id 1.1.1.1
[*Leaf1-bgp] peer 7.7.7.7 as-number 100
[*Leaf1-bgp] peer 7.7.7.7 connect-interface LoopBack0
[*Leaf1-bgp] peer 8.8.8.8 as-number 100
[*Leaf1-bgp] peer 8.8.8.8 connect-interface LoopBack0
[*Leaf1-bgp] l2vpn-family evpn
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 advertise irb
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 advertise irbv6
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 advertise irb
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 advertise irbv6
[*Leaf1-bgp-af-evpn] quit
[*Leaf1-bgp] quit
[*Leaf1] commit
```

2. 配置VPN实例及EVPN实例。

   # 配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] ip vpn-instance vpn1
[*Leaf1-vpn-instance-vpn1] vxlan vni 200
[*Leaf1-vpn-instance-vpn1] ipv4-family
[*Leaf1-vpn-instance-vpn1-af-ipv4] route-distinguisher 1.1.1.1:1
[*Leaf1-vpn-instance-vpn1-af-ipv4] vpn-target 0:1 evpn
[*Leaf1-vpn-instance-vpn1-af-ipv4] quit
[*Leaf1-vpn-instance-vpn1] ipv6-family
[*Leaf1-vpn-instance-vpn1-af-ipv6] route-distinguisher 1.1.1.1:1
[*Leaf1-vpn-instance-vpn1-af-ipv6] vpn-target 0:1 evpn
```

```
[*Leaf1-vpn-instance-vpn1-af-ipv6] quit
[*Leaf1-vpn-instance-vpn1] quit
[*Leaf1] bridge-domain 10
[*Leaf1-bd10] vxlan vni 110
[*Leaf1-bd10] evpn
[*Leaf1-bd10-evpn] route-distinguisher 1.1.1.1:110
[*Leaf1-bd10-evpn] vpn-target 0:110
[*Leaf1-bd10-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd10-evpn] quit
[*Leaf1-bd10] quit
[*Leaf1] bridge-domain 20
[*Leaf1-bd20] vxlan vni 120
[*Leaf1-bd20] evpn
[*Leaf1-bd20-evpn] route-distinguisher 1.1.1.1:120
[*Leaf1-bd20-evpn] vpn-target 0:120
[*Leaf1-bd20-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd20-evpn] quit
[*Leaf1-bd20] quit
[*Leaf1] bridge-domain 30
[*Leaf1-bd30] vxlan vni 130
[*Leaf1-bd30] evpn
[*Leaf1-bd30-evpn] route-distinguisher 1.1.1.1:130
[*Leaf1-bd30-evpn] vpn-target 0:130
[*Leaf1-bd30-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd30-evpn] quit
[*Leaf1-bd30] quit
[*Leaf1] bridge-domain 40
[*Leaf1-bd40] vxlan vni 140
[*Leaf1-bd40] evpn
[*Leaf1-bd40-evpn] route-distinguisher 1.1.1.1:140
[*Leaf1-bd40-evpn] vpn-target 0:140
[*Leaf1-bd40-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd40-evpn] quit
[*Leaf1-bd40] quit
[*Leaf1] interface nve 1
[*Leaf1-Nve1] source 5.5.5.5   //组建M-LAG的两台设备上配置的NVE接口的IP地址和MAC地址需要相同
[*Leaf1-Nve1] mac-address 00e0-fc00-0111
[*Leaf1-Nve1] vni 110 head-end peer-list protocol bgp
[*Leaf1-Nve1] vni 120 head-end peer-list protocol bgp
[*Leaf1-Nve1] vni 130 head-end peer-list protocol bgp
[*Leaf1-Nve1] vni 140 head-end peer-list protocol bgp
[*Leaf1-Nve1] quit
[*Leaf1] commit
```

3. 配置三层网关。

   # 配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] interface vbdif10   //Leaf和DCGW上同一VBDIF的IP地址和MAC地址需要相同
[*Leaf1-Vbdif10] ip binding vpn-instance vpn1
[*Leaf1-Vbdif10] ip address 10.1.1.1 24
[*Leaf1-Vbdif10] ipv6 enable
[*Leaf1-Vbdif10] ipv6 address fc00:1::1 64
[*Leaf1-Vbdif10] arp generate-rd-table enable
[*Leaf1-Vbdif10] ipv6 nd generate-rd-table enable
[*Leaf1-Vbdif10] arp broadcast-detect enable
[*Leaf1-Vbdif10] ipv6 nd na glean
[*Leaf1-Vbdif10] vxlan anycast-gateway enable
[*Leaf1-Vbdif10] arp collect host enable
[*Leaf1-Vbdif10] ipv6 nd collect host enable
[*Leaf1-Vbdif10] mac-address 00e0-fc00-0101
[*Leaf1-Vbdif10] quit
[*Leaf1] interface vbdif20
[*Leaf1-Vbdif20] ip binding vpn-instance vpn1
[*Leaf1-Vbdif20] ip address 10.2.1.1 24
[*Leaf1-Vbdif20] ipv6 enable
[*Leaf1-Vbdif20] ipv6 address fc00:2::1 64
[*Leaf1-Vbdif20] arp generate-rd-table enable
[*Leaf1-Vbdif20] ipv6 nd generate-rd-table enable
[*Leaf1-Vbdif20] arp broadcast-detect enable
[*Leaf1-Vbdif20] ipv6 nd na glean
```

```
[*Leaf1-Vbdif20] vxlan anycast-gateway enable
[*Leaf1-Vbdif20] arp collect host enable
[*Leaf1-Vbdif20] ipv6 nd collect host enable
[*Leaf1-Vbdif20] mac-address 00e0-fc00-0102
[*Leaf1-Vbdif20] quit
[*Leaf1] interface vbdif30
[*Leaf1-Vbdif30] ip binding vpn-instance vpn1
[*Leaf1-Vbdif30] ip address 10.3.1.1 24
[*Leaf1-Vbdif30] ipv6 enable
[*Leaf1-Vbdif30] ipv6 address fc00:3::1 64
[*Leaf1-Vbdif30] arp generate-rd-table enable
[*Leaf1-Vbdif30] ipv6 nd generate-rd-table enable
[*Leaf1-Vbdif30] arp broadcast-detect enable
[*Leaf1-Vbdif30] ipv6 nd na glean
[*Leaf1-Vbdif30] vxlan anycast-gateway enable
[*Leaf1-Vbdif30] arp collect host enable
[*Leaf1-Vbdif30] ipv6 nd collect host enable
[*Leaf1-Vbdif30] mac-address 00e0-fc00-0103
[*Leaf1-Vbdif30] quit
[*Leaf1] interface vbdif40
[*Leaf1-Vbdif40] ip binding vpn-instance vpn1
[*Leaf1-Vbdif40] ip address 10.4.1.1 24
[*Leaf1-Vbdif40] ipv6 enable
[*Leaf1-Vbdif40] ipv6 address fc00:4::1 64
[*Leaf1-Vbdif40] arp generate-rd-table enable
[*Leaf1-Vbdif40] ipv6 nd generate-rd-table enable
[*Leaf1-Vbdif40] arp broadcast-detect enable
[*Leaf1-Vbdif40] ipv6 nd na glean
[*Leaf1-Vbdif40] vxlan anycast-gateway enable
[*Leaf1-Vbdif40] arp collect host enable
[*Leaf1-Vbdif40] ipv6 nd collect host enable
[*Leaf1-Vbdif40] mac-address 00e0-fc00-0104
[*Leaf1-Vbdif40] quit
[*Leaf1] commit
```

4. 配置业务接入点

   # 配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] interface eth-trunk 10.10 mode l2
[*Leaf1-Eth-Trunk10.10] encapsulation dot1q vid 10
[*Leaf1-Eth-Trunk10.10] bridge-domain 10
[*Leaf1-Eth-Trunk10.10] quit
[*Leaf1] interface eth-trunk 11.20 mode l2
[*Leaf1-Eth-Trunk11.20] encapsulation dot1q vid 20
[*Leaf1-Eth-Trunk11.20] bridge-domain 20
[*Leaf1-Eth-Trunk11.20] quit
[*Leaf1] commit
```

步骤5 在Leaf上配置BFD，检测Leaf和VNF之间链路。

配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] bfd toipu1_v4 bind peer-ip 10.1.1.2 vpn-instance vpn1 interface vbdif10 source-ip 10.1.1.1 one-arm-echo
[*Leaf1-bfd-session-toipu1_v4] discriminator local 10
[*Leaf1-bfd-session-toipu1_v4] detect-multiplier 6
[*Leaf1-bfd-session-toipu1_v4] min-echo-rx-interval 300
[*Leaf1-bfd-session-toipu1_v4] quit
[*Leaf1] bfd toipu2_v4 bind peer-ip 10.2.1.2 vpn-instance vpn1 interface vbdif20 source-ip 10.2.1.1 one-arm-echo
[*Leaf1-bfd-session-toipu2_v4] discriminator local 30
[*Leaf1-bfd-session-toipu2_v4] detect-multiplier 6
[*Leaf1-bfd-session-toipu2_v4] min-echo-rx-interval 300
[*Leaf1-bfd-session-toipu2_v4] quit
[*Leaf1] bfd toipu1_v6 bind peer-ipv6 fc00:1::2 vpn-instance vpn1 interface vbdif10 source-ipv6 fc00:1::1 one-arm-echo
[*Leaf1-bfd-session-toipu1_v6] discriminator local 50
[*Leaf1-bfd-session-toipu1_v6] detect-multiplier 6
[*Leaf1-bfd-session-toipu1_v6] min-echo-rx-interval 300
[*Leaf1-bfd-session-toipu1_v6] quit
```

```
[*Leaf1] bfd toipu2_v6 bind peer-ipv6 fc00:2::2 vpn-instance vpn1 interface vbdif20 source-ipv6 fc00:2::1 one-arm-echo
[*Leaf1-bfd-session-toipu2_v6] discriminator local 70
[*Leaf1-bfd-session-toipu2_v6] detect-multiplier 6
[*Leaf1-bfd-session-toipu2_v6] min-echo-rx-interval 300
[*Leaf1-bfd-session-toipu2_v6] quit
[*Leaf1] bfd
[*Leaf1-bfd] bfd forwarding match remote-discriminator 20  //指定M-LAG场景下的BFD远端标识符,即对应Leaf2的discriminator local值
[*Leaf1-bfd] bfd forwarding match remote-discriminator 40
[*Leaf1-bfd] bfd forwarding match remote-discriminator 60
[*Leaf1-bfd] bfd forwarding match remote-discriminator 80
[*Leaf1-bfd] quit
[*Leaf1] commit
```

步骤6 配置Leaf通往VNF的私网静态路由，并配置BGP EVPN引入私网静态路由，然后配置L3VPN实例应用路由策略，使这些静态私网路由保持原有下一跳。

配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] route-policy import_static_policy permit node 100  //配置一个路由策略，在将静态路由引入BGP时增加团体属性
[*Leaf1-route-policy] if-match tag 82345
[*Leaf1-route-policy] apply community 82345 additive
[*Leaf1-route-policy] quit
[*Leaf1] route-policy import_static_policy permit node 999
[*Leaf1-route-policy] quit
[*Leaf1] ip community-filter basic apply_gwip_route index 10 permit 82345  //创建团体属性过滤器
[*Leaf1] ip community-filter basic suppress_route index 10 permit 82346
[*Leaf1] route-policy export_policy deny node 10  //配置一个路由策略，用于过滤/允许路由通过
[*Leaf1-route-policy] if-match community-filter suppress_route
[*Leaf1-route-policy] quit
[*Leaf1] route-policy export_policy permit node 20
[*Leaf1-route-policy] if-match community-filter apply_gwip_route
[*Leaf1-route-policy] apply gateway-ip origin-nexthop  //设置路由的下一跳地址作为网关IP地址。在L2GW/L3GW上配置L3VPN实例向EVPN发布可以到达VNF的私网静态路由时，需要先创建路由策略，该路由策略可以过滤出L3VPN实例中可以到达VNF的私网静态路由。
[*Leaf1-route-policy] apply ipv6 gateway-ip origin-nexthop
[*Leaf1-route-policy] quit
[*Leaf1] route-policy export_policy permit node 999
[*Leaf1-route-policy] quit
[*Leaf1] ip route-static vpn-instance vpn1 172.16.1.1 32 10.1.1.2 preference 255 tag 82345 track bfd-session toipu1_v4 inter-protocol-ecmp  //此处配置的Tag值需要与路由策略相匹配，以实现Leaf将路由发布到DCGW时路由的下一跳地址为网关IP地址，实现非对称转发
[*Leaf1] ip route-static vpn-instance vpn1 172.16.1.1 32 10.2.1.2 preference 255 tag 82345 track bfd-session toipu2_v4 inter-protocol-ecmp
[*Leaf1] ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 fc00:1::2 preference 255 tag 82345 track bfd-session toipu1_v6 inter-protocol-ecmp
[*Leaf1] ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 fc00:2::2 preference 255 tag 82345 track bfd-session toipu2_v6 inter-protocol-ecmp
[*Leaf1] bgp 100
[*Leaf1-bgp] ipv4-family vpn-instance vpn1
[*Leaf1-bgp-vpn1] import-route static route-policy import_static_policy
[*Leaf1-bgp-vpn1] advertise l2vpn evpn import-route-multipath  //发布所有目的地址相同的路由
[*Leaf1-bgp-vpn1] irb asymmetric  //使能IRB路由的非对称模式
[*Leaf1-bgp-vpn1] quit
[*Leaf1-bgp] ipv6-family vpn-instance vpn1
[*Leaf1-bgp-6-vpn1] import-route static route-policy import_static_policy
[*Leaf1-bgp-6-vpn1] advertise l2vpn evpn import-route-multipath
[*Leaf1-bgp-6-vpn1] irb asymmetric
[*Leaf1-bgp-6-vpn1] quit
[*Leaf1-bgp] quit
[*Leaf1] commit
```

步骤7 配置路由负载分担。

配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] bgp 100
[*Leaf1-bgp] ipv4-family vpn-instance vpn1
```



```
[*Leaf1-bgp-vpn1] maximum load-balancing 64
[*Leaf1-bgp-vpn1] quit
[*Leaf1-bgp] ipv6-family vpn-instance vpn1
[*Leaf1-bgp-6-vpn1] maximum load-balancing 64
[*Leaf1-bgp-6-vpn1] quit
[*Leaf1-bgp] l2vpn-family evpn
[*Leaf1-bgp-af-evpn] bestroute add-path path-number 64   //使能BGP ADD-PATH功能
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 capability-advertise add-path both
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 advertise add-path path-number 64
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 route-policy export_policy export  //对RR使用路由策略
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 capability-advertise add-path both
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 advertise add-path path-number 64
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 route-policy export_policy export
[*Leaf1-bgp-af-evpn] quit
[*Leaf1-bgp] quit
[*Leaf1] commit
```

----结束

检查配置结果

配置完成后，在Leaf上执行**display ip routing-table vpn-instance vpn1**命令可以看到从DCGW收到的路由，其下一跳为Anycast VTEP地址。以Leaf1为例：

```
[~Leaf1] display ip routing-table vpn-instance vpn1
Proto: Protocol        Pre: Preference
Route Flags: R - relay, D - download to fib, T - to vpn-instance, B - black hole route
------------------------------------------------------------------------------
Routing Table : vpn1
         Destinations : 16       Routes : 18

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        0.0.0.0/0   IBGP    255  0           RD  11.11.11.11     VXLAN
       10.1.1.0/24  Direct  0    0           D   10.1.1.1        Vbdif10
       10.1.1.1/32  Direct  0    0           D   127.0.0.1       Vbdif10
     10.1.1.255/32  Direct  0    0           D   127.0.0.1       Vbdif10
       10.2.1.0/24  Direct  0    0           D   10.2.1.1        Vbdif20
       10.2.1.1/32  Direct  0    0           D   127.0.0.1       Vbdif20
     10.2.1.255/32  Direct  0    0           D   127.0.0.1       Vbdif20
       10.3.1.0/24  Direct  0    0           D   10.3.1.1        Vbdif30
       10.3.1.1/32  Direct  0    0           D   127.0.0.1       Vbdif30
     10.3.1.255/32  Direct  0    0           D   127.0.0.1       Vbdif30
       10.4.1.0/24  Direct  0    0           D   10.4.1.1        Vbdif40
       10.4.1.1/32  Direct  0    0           D   127.0.0.1       Vbdif40
     10.4.1.255/32  Direct  0    0           D   127.0.0.1       Vbdif40
     172.16.1.1/32  Static  255  0           RD  10.1.1.2        Vbdif10
                    Static  255  0           RD  10.2.1.2        Vbdif20
     172.16.2.1/32  IBGP    255  0           RD  10.3.1.2        Vbdif30
                    IBGP    255  0           RD  10.4.1.2        Vbdif40
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

在DCGW上可以查看到IPU的ARP信息，略。

配置脚本

- Spine1的配置脚本

```
#
sysname Spine1
#
evpn-overlay enable
#
bfd
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
```

```
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.2.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.3.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 undo portswitch
 ip address 192.168.4.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 undo portswitch
 ip address 192.168.9.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
#
interface 100GE1/0/6
 undo portswitch
 ip address 192.168.10.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
#
interface LoopBack0
 ip address 7.7.7.7 255.255.255.255
#
bgp 100
 router-id 7.7.7.7
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack0
 peer 2.2.2.2 as-number 100
 peer 2.2.2.2 connect-interface LoopBack0
 peer 3.3.3.3 as-number 100
 peer 3.3.3.3 connect-interface LoopBack0
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack0
 peer 9.9.9.9 as-number 100
 peer 9.9.9.9 connect-interface LoopBack0
 peer 10.10.10.10 as-number 100
 peer 10.10.10.10 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 1.1.1.1 enable
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
  peer 4.4.4.4 enable
  peer 9.9.9.9 enable
```

```
 peer 10.10.10.10 enable
#
l2vpn-family evpn
 undo policy vpn-target
 bestroute add-path path-number 64
 peer 1.1.1.1 enable
 peer 1.1.1.1 advertise irb
 peer 1.1.1.1 advertise irbv6
 peer 1.1.1.1 reflect-client
 peer 1.1.1.1 capability-advertise add-path both
 peer 1.1.1.1 advertise add-path path-number 64
 peer 2.2.2.2 enable
 peer 2.2.2.2 advertise irb
 peer 2.2.2.2 advertise irbv6
 peer 2.2.2.2 reflect-client
 peer 2.2.2.2 capability-advertise add-path both
 peer 2.2.2.2 advertise add-path path-number 64
 peer 3.3.3.3 enable
 peer 3.3.3.3 advertise irb
 peer 3.3.3.3 advertise irbv6
 peer 3.3.3.3 reflect-client
 peer 3.3.3.3 capability-advertise add-path both
 peer 3.3.3.3 advertise add-path path-number 64
 peer 4.4.4.4 enable
 peer 4.4.4.4 advertise irb
 peer 4.4.4.4 advertise irbv6
 peer 4.4.4.4 reflect-client
 peer 4.4.4.4 capability-advertise add-path both
 peer 4.4.4.4 advertise add-path path-number 64
 peer 9.9.9.9 enable
 peer 9.9.9.9 advertise irb
 peer 9.9.9.9 advertise irbv6
 peer 9.9.9.9 capability-advertise add-path both
 peer 9.9.9.9 advertise add-path path-number 64
 peer 10.10.10.10 enable
 peer 10.10.10.10 advertise irb
 peer 10.10.10.10 advertise irbv6
 peer 10.10.10.10 capability-advertise add-path both
 peer 10.10.10.10 advertise add-path path-number 64
#
ospf 1 router-id 7.7.7.7
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
 area 0.0.0.0
 network 7.7.7.7 0.0.0.0
 network 192.168.1.0 0.0.0.255
 network 192.168.2.0 0.0.0.255
 network 192.168.3.0 0.0.0.255
 network 192.168.4.0 0.0.0.255
 network 192.168.9.0 0.0.0.255
 network 192.168.10.0 0.0.0.255
#
return
```

- Spine2的配置脚本

```
#
sysname Spine2
#
evpn-overlay enable
#
bfd
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.5.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
```

```
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.6.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.7.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 undo portswitch
 ip address 192.168.8.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 undo portswitch
 ip address 192.168.11.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
#
interface 100GE1/0/6
 undo portswitch
 ip address 192.168.12.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
#
interface LoopBack0
 ip address 8.8.8.8 255.255.255.255
#
bgp 100
 router-id 8.8.8.8
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack0
 peer 2.2.2.2 as-number 100
 peer 2.2.2.2 connect-interface LoopBack0
 peer 3.3.3.3 as-number 100
 peer 3.3.3.3 connect-interface LoopBack0
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack0
 peer 9.9.9.9 as-number 100
 peer 9.9.9.9 connect-interface LoopBack0
 peer 10.10.10.10 as-number 100
 peer 10.10.10.10 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 1.1.1.1 enable
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
  peer 4.4.4.4 enable
  peer 9.9.9.9 enable
  peer 10.10.10.10 enable
 #
```

```
 l2vpn-family evpn
  undo policy vpn-target
  bestroute add-path path-number 64
  peer 1.1.1.1 enable
  peer 1.1.1.1 advertise irb
  peer 1.1.1.1 advertise irbv6
  peer 1.1.1.1 reflect-client
  peer 1.1.1.1 capability-advertise add-path both
  peer 1.1.1.1 advertise add-path path-number 64
  peer 2.2.2.2 enable
  peer 2.2.2.2 advertise irb
  peer 2.2.2.2 advertise irbv6
  peer 2.2.2.2 reflect-client
  peer 2.2.2.2 capability-advertise add-path both
  peer 2.2.2.2 advertise add-path path-number 64
  peer 3.3.3.3 enable
  peer 3.3.3.3 advertise irb
  peer 3.3.3.3 advertise irbv6
  peer 3.3.3.3 reflect-client
  peer 3.3.3.3 capability-advertise add-path both
  peer 3.3.3.3 advertise add-path path-number 64
  peer 4.4.4.4 enable
  peer 4.4.4.4 advertise irb
  peer 4.4.4.4 advertise irbv6
  peer 4.4.4.4 reflect-client
  peer 4.4.4.4 capability-advertise add-path both
  peer 4.4.4.4 advertise add-path path-number 64
  peer 9.9.9.9 enable
  peer 9.9.9.9 advertise irb
  peer 9.9.9.9 advertise irbv6
  peer 9.9.9.9 capability-advertise add-path both
  peer 9.9.9.9 advertise add-path path-number 64
  peer 10.10.10.10 enable
  peer 10.10.10.10 advertise irb
  peer 10.10.10.10 advertise irbv6
  peer 10.10.10.10 capability-advertise add-path both
  peer 10.10.10.10 advertise add-path path-number 64
#
ospf 1 router-id 8.8.8.8
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
 area 0.0.0.0
  network 8.8.8.8 0.0.0.0
  network 192.168.5.0 0.0.0.255
  network 192.168.6.0 0.0.0.255
  network 192.168.7.0 0.0.0.255
  network 192.168.8.0 0.0.0.255
  network 192.168.11.0 0.0.0.255
  network 192.168.12.0 0.0.0.255
#
return
```

- Leaf1的配置脚本

```
#
sysname Leaf1
#
dfs-group 1
 priority 150
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
 dual-active detection delay 0
 dual-active detection source ip 1.1.1.1 peer 2.2.2.2
 m-lag up-delay 240 auto-recovery interval 10
#
vlan 100
 m-lag peer-link reserved
#
```

```
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 1.1.1.1:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 1.1.1.1:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 200
#
bfd
 bfd forwarding match remote-discriminator 20
 bfd forwarding match remote-discriminator 40
 bfd forwarding match remote-discriminator 60
 bfd forwarding match remote-discriminator 80
#
bridge-domain 10
 vxlan vni 110
 evpn
  route-distinguisher 1.1.1.1:110
  vpn-target 0:110 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:110 import-extcommunity
#
bridge-domain 20
 vxlan vni 120
 evpn
  route-distinguisher 1.1.1.1:120
  vpn-target 0:120 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:120 import-extcommunity
#
bridge-domain 30
 vxlan vni 130
 evpn
  route-distinguisher 1.1.1.1:130
  vpn-target 0:130 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:130 import-extcommunity
#
bridge-domain 40
 vxlan vni 140
 evpn
  route-distinguisher 1.1.1.1:140
  vpn-target 0:140 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:140 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0101
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
```

```
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.2.1.1 255.255.255.0
 ipv6 address FC00:2::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0102
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.3.1.1 255.255.255.0
 ipv6 address FC00:3::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0103
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif40
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.4.1.1 255.255.255.0
 ipv6 address FC00:4::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0104
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.10.10.1 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 1
 arp anti-attack rate-limit 200
#
interface Eth-Trunk10.10 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface Eth-Trunk11
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 2
 arp anti-attack rate-limit 200
#
interface Eth-Trunk11.20 mode l2
```



```
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.5.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 eth-trunk 10
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/6
 eth-trunk 11
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface LoopBack0
 ip address 1.1.1.1 255.255.255.255
#
interface LoopBack1
 ip address 5.5.5.5 255.255.255.255
#
interface LoopBack2
 ip address 12.12.12.12 255.255.255.255
#
interface Nve1
 source 5.5.5.5
 pip-source 12.12.12.12 peer 13.13.13.13 bypass
 vni 110 head-end peer-list protocol bgp
 vni 120 head-end peer-list protocol bgp
 vni 130 head-end peer-list protocol bgp
 vni 140 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0111
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
```

```
 port Eth-Trunk11 downlink 2
 timer recover-time 60
#
bfd toipu1_v4 bind peer-ip 10.1.1.2 vpn-instance vpn1 interface Vbdif10 source-ip 10.1.1.1 one-arm-
echo
 discriminator local 10
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu1_v6 bind peer-ipv6 FC00:1::2 vpn-instance vpn1 interface Vbdif10 source-ipv6 FC00:1::1 one-
arm-echo
 discriminator local 50
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu2_v4 bind peer-ip 10.2.1.2 vpn-instance vpn1 interface Vbdif20 source-ip 10.2.1.1 one-arm-
echo
 discriminator local 30
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu2_v6 bind peer-ipv6 FC00:2::2 vpn-instance vpn1 interface Vbdif20 source-ipv6 FC00:2::1 one-
arm-echo
 discriminator local 70
 detect-multiplier 6
 min-echo-rx-interval 300
#
bgp 100
 router-id 1.1.1.1
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing 64
  irb asymmetric
  advertise l2vpn evpn import-route-multipath
 #
 ipv6-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing 64
  irb asymmetric
  advertise l2vpn evpn import-route-multipath
 #
 l2vpn-family evpn
  policy vpn-target
  bestroute add-path path-number 64
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy export_policy export
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 capability-advertise add-path both
  peer 7.7.7.7 advertise add-path path-number 64
  peer 8.8.8.8 enable
  peer 8.8.8.8 route-policy export_policy export
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 capability-advertise add-path both
  peer 8.8.8.8 advertise add-path path-number 64
#
ospf 1 router-id 1.1.1.1
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
```

```
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
 area 0.0.0.0
  network 1.1.1.1 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.5.0 0.0.0.255
#
ip community-filter basic apply_gwip_route index 10 permit 82345
ip community-filter basic suppress_route index 10 permit 82346
#
route-policy export_policy deny node 10
 if-match community-filter suppress_route
#
route-policy export_policy permit node 20
 if-match community-filter apply_gwip_route
 apply gateway-ip origin-nexthop
 apply ipv6 gateway-ip origin-nexthop
#
route-policy export_policy permit node 999
#
route-policy import_static_policy permit node 100
 if-match tag 82345
 apply community 82345 additive
#
route-policy import_static_policy permit node 999
#
ip route-static 13.13.13.13 32 10.10.10.2 preference 1
ip route-static vpn-instance vpn1 172.16.1.1 255.255.255.255 10.1.1.2 preference 255 tag 82345 track
bfd-session toipu1_v4 inter-protocol-ecmp
ip route-static vpn-instance vpn1 172.16.1.1 255.255.255.255 10.2.1.2 preference 255 tag 82345 track
bfd-session toipu2_v4 inter-protocol-ecmp
#
ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 FC00:1::2 preference 255 tag 82345 track bfd-
session toipu1_v6 inter-protocol-ecmp
ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 FC00:2::2 preference 255 tag 82345 track bfd-
session toipu2_v6 inter-protocol-ecmp
#
return
```

- Leaf2的配置脚本

```
#
sysname Leaf2
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection delay 0
 dual-active detection source ip 2.2.2.2 peer 1.1.1.1
 m-lag up-delay 240 auto-recovery interval 10
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 2.2.2.2:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 2.2.2.2:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 200
```

```
#
bfd
 bfd forwarding match remote-discriminator 10
 bfd forwarding match remote-discriminator 30
 bfd forwarding match remote-discriminator 50
 bfd forwarding match remote-discriminator 70
#
bridge-domain 10
 vxlan vni 110
 evpn
  route-distinguisher 2.2.2.2:110
  vpn-target 0:110 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:110 import-extcommunity
#
bridge-domain 20
 vxlan vni 120
 evpn
  route-distinguisher 2.2.2.2:120
  vpn-target 0:120 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:120 import-extcommunity
#
bridge-domain 30
 vxlan vni 130
 evpn
  route-distinguisher 2.2.2.2:130
  vpn-target 0:130 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:130 import-extcommunity
#
bridge-domain 40
 vxlan vni 140
 evpn
  route-distinguisher 2.2.2.2:140
  vpn-target 0:140 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:140 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0101
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.2.1.1 255.255.255.0
 ipv6 address FC00:2::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0102
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif30
 ip binding vpn-instance vpn1
```

```
 ipv6 enable
 ip address 10.3.1.1 255.255.255.0
 ipv6 address FC00:3::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0103
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif40
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.4.1.1 255.255.255.0
 ipv6 address FC00:4::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0104
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.10.10.2 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 1
 arp anti-attack rate-limit 200
#
interface Eth-Trunk10.10 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface Eth-Trunk11
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 2
 arp anti-attack rate-limit 200
#
interface Eth-Trunk11.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.2.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.6.2 255.255.255.0
```

```
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 eth-trunk 10
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/6
 eth-trunk 11
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface LoopBack0
 ip address 2.2.2.2 255.255.255.255
#
interface LoopBack1
 ip address 5.5.5.5 255.255.255.255
#
interface LoopBack2
 ip address 13.13.13.13 255.255.255.255
#
interface Nve1
 source 5.5.5.5
 pip-source 13.13.13.13 peer 12.12.12.12 bypass
 vni 110 head-end peer-list protocol bgp
 vni 120 head-end peer-list protocol bgp
 vni 130 head-end peer-list protocol bgp
 vni 140 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0111
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk11 downlink 2
 timer recover-time 60
#
bfd toipu1_v4 bind peer-ip 10.1.1.2 vpn-instance vpn1 interface Vbdif10 source-ip 10.1.1.1 one-arm-
echo
 discriminator local 20
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu1_v6 bind peer-ipv6 FC00:1::2 vpn-instance vpn1 interface Vbdif10 source-ipv6 FC00:1::1 one-
arm-echo
 discriminator local 60
 detect-multiplier 6
 min-echo-rx-interval 300
#
```

```
bfd toipu2_v4 bind peer-ip 10.2.1.2 vpn-instance vpn1 interface Vbdif20 source-ip 10.2.1.1 one-arm-
echo
 discriminator local 40
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu2_v6 bind peer-ipv6 FC00:2::2 vpn-instance vpn1 interface Vbdif20 source-ipv6 FC00:2::1 one-
arm-echo
 discriminator local 80
 detect-multiplier 6
 min-echo-rx-interval 300
#
bgp 100
 router-id 2.2.2.2
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing 64
  irb asymmetric
  advertise l2vpn evpn import-route-multipath
 #
 ipv6-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing 64
  irb asymmetric
  advertise l2vpn evpn import-route-multipath
 #
 l2vpn-family evpn
  policy vpn-target
  bestroute add-path path-number 64
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy export_policy export
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 capability-advertise add-path both
  peer 7.7.7.7 advertise add-path path-number 64
  peer 8.8.8.8 enable
  peer 8.8.8.8 route-policy export_policy export
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 capability-advertise add-path both
  peer 8.8.8.8 advertise add-path path-number 64
#
ospf 1 router-id 2.2.2.2
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
 area 0.0.0.0
  network 2.2.2.2 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.2.0 0.0.0.255
  network 192.168.6.0 0.0.0.255
#
ip community-filter basic apply_gwip_route index 10 permit 82345
ip community-filter basic suppress_route index 10 permit 82346
#
route-policy export_policy deny node 10
 if-match community-filter suppress_route
#
```



```
route-policy export_policy permit node 20
 if-match community-filter apply_gwip_route
 apply gateway-ip origin-nexthop
 apply ipv6 gateway-ip origin-nexthop
#
route-policy export_policy permit node 999
#
route-policy import_static_policy permit node 100
 if-match tag 82345
 apply community 82345 additive
#
route-policy import_static_policy permit node 999
#
ip route-static 12.12.12.12 32 10.10.10.1 preference 1
ip route-static vpn-instance vpn1 172.16.1.1 255.255.255.255 10.1.1.2 preference 255 tag 82345 track
bfd-session toipu1_v4 inter-protocol-ecmp
ip route-static vpn-instance vpn1 172.16.1.1 255.255.255.255 10.2.1.2 preference 255 tag 82345 track
bfd-session toipu2_v4 inter-protocol-ecmp
#
ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 FC00:1::2 preference 255 tag 82345 track bfd-
session toipu1_v6 inter-protocol-ecmp
ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 FC00:2::2 preference 255 tag 82345 track bfd-
session toipu2_v6 inter-protocol-ecmp
#
return
```

- Leaf3的配置脚本

```
#
sysname Leaf3
#
dfs-group 1
 priority 150
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection delay 0
 dual-active detection source ip 3.3.3.3 peer 4.4.4.4
 m-lag up-delay 240 auto-recovery interval 10
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 3.3.3.3:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 1.1.1.1:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 200
#
bfd
 bfd forwarding match remote-discriminator 20
 bfd forwarding match remote-discriminator 40
 bfd forwarding match remote-discriminator 60
 bfd forwarding match remote-discriminator 80
#
bridge-domain 10
 vxlan vni 110
 evpn
  route-distinguisher 3.3.3.3:110
  vpn-target 0:110 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:110 import-extcommunity
```



```
#
bridge-domain 20
 vxlan vni 120
 evpn
  route-distinguisher 3.3.3.3:120
  vpn-target 0:120 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:120 import-extcommunity
#
bridge-domain 30
 vxlan vni 130
 evpn
  route-distinguisher 3.3.3.3:130
  vpn-target 0:130 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:130 import-extcommunity
#
bridge-domain 40
 vxlan vni 140
 evpn
  route-distinguisher 3.3.3.3:140
  vpn-target 0:140 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:140 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0101
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.2.1.1 255.255.255.0
 ipv6 address FC00:2::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0102
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.3.1.1 255.255.255.0
 ipv6 address FC00:3::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0103
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif40
 ip binding vpn-instance vpn1
```

```
 ipv6 enable
 ip address 10.4.1.1 255.255.255.0
 ipv6 address FC00:4::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0104
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.10.10.5 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 1
 arp anti-attack rate-limit 200
#
interface Eth-Trunk10.30 mode l2
 encapsulation dot1q vid 30
 bridge-domain 30
#
interface Eth-Trunk11
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 2
 arp anti-attack rate-limit 200
#
interface Eth-Trunk11.40 mode l2
 encapsulation dot1q vid 40
 bridge-domain 40
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.3.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.7.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 eth-trunk 1
 port crc-statistics trigger error-down
```



```
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 eth-trunk 10
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/6
 eth-trunk 11
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface LoopBack0
 ip address 3.3.3.3 255.255.255.255
#
interface LoopBack1
 ip address 6.6.6.6 255.255.255.255
#
interface LoopBack2
 ip address 14.14.14.14 255.255.255.255
#
interface Nve1
 source 6.6.6.6
 pip-source 14.14.14.14 peer 15.15.15.15 bypass
 vni 110 head-end peer-list protocol bgp
 vni 120 head-end peer-list protocol bgp
 vni 130 head-end peer-list protocol bgp
 vni 140 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0112
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk11 downlink 2
 timer recover-time 60
#
bfd toipu3_v4 bind peer-ip 10.3.1.2 vpn-instance vpn1 interface Vbdif30 source-ip 10.3.1.1 one-arm-
echo
 discriminator local 10
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu3_v6 bind peer-ipv6 FC00:3::2 vpn-instance vpn1 interface Vbdif30 source-ipv6 FC00:3::1 one-
arm-echo
 discriminator local 50
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu4_v4 bind peer-ip 10.4.1.2 vpn-instance vpn1 interface Vbdif40 source-ip 10.4.1.1 one-arm-
echo
 discriminator local 30
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu4_v6 bind peer-ipv6 FC00:4::2 vpn-instance vpn1 interface Vbdif40 source-ipv6 FC00:4::1 one-
arm-echo
 discriminator local 70
 detect-multiplier 6
 min-echo-rx-interval 300
#
bgp 100
 router-id 3.3.3.3
```

```
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing 64
  irb asymmetric
  advertise l2vpn evpn import-route-multipath
 #
 ipv6-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing 64
  irb asymmetric
  advertise l2vpn evpn import-route-multipath
 #
 l2vpn-family evpn
  policy vpn-target
  bestroute add-path path-number 64
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy export_policy export
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 capability-advertise add-path both
  peer 7.7.7.7 advertise add-path path-number 64
  peer 8.8.8.8 enable
  peer 8.8.8.8 route-policy export_policy export
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 capability-advertise add-path both
  peer 8.8.8.8 advertise add-path path-number 64
#
ospf 1 router-id 3.3.3.3
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
  area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.3.0 0.0.0.255
  network 192.168.7.0 0.0.0.255
#
ip community-filter basic apply_gwip_route index 10 permit 82345
ip community-filter basic suppress_route index 10 permit 82346
#
route-policy export_policy deny node 10
 if-match community-filter suppress_route
#
route-policy export_policy permit node 20
 if-match community-filter apply_gwip_route
 apply gateway-ip origin-nexthop
 apply ipv6 gateway-ip origin-nexthop
#
route-policy export_policy permit node 999
#
route-policy import_static_policy permit node 100
 if-match tag 82345
 apply community 82345 additive
#
route-policy import_static_policy permit node 999
#
ip route-static 15.15.15.15 32 10.10.10.6 preference 1
```

```
ip route-static vpn-instance vpn1 172.16.2.1 255.255.255.255 10.3.1.2 preference 255 tag 82345 track
bfd-session toipu3_v4 inter-protocol-ecmp
ip route-static vpn-instance vpn1 172.16.2.1 255.255.255.255 10.4.1.2 preference 255 tag 82345 track
bfd-session toipu4_v4 inter-protocol-ecmp
#
ipv6 route-static vpn-instance vpn1 2001:db8:2::1 128 FC00:3::2 preference 255 tag 82345 track bfd-
session toipu3_v6 inter-protocol-ecmp
ipv6 route-static vpn-instance vpn1 2001:db8:2::1 128 FC00:4::2 preference 255 tag 82345 track bfd-
session toipu4_v6 inter-protocol-ecmp
#
return
```

- Leaf4的配置脚本

```
#
sysname Leaf4
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection delay 0
 dual-active detection source ip 4.4.4.4 peer 3.3.3.3
 m-lag up-delay 240 auto-recovery interval 10
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 4.4.4.4:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 1.1.1.1:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 200
#
bfd
 bfd forwarding match remote-discriminator 10
 bfd forwarding match remote-discriminator 30
 bfd forwarding match remote-discriminator 50
 bfd forwarding match remote-discriminator 70
#
bridge-domain 10
 vxlan vni 110
 evpn
  route-distinguisher 4.4.4.4:110
  vpn-target 0:110 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:110 import-extcommunity
#
bridge-domain 20
 vxlan vni 120
 evpn
  route-distinguisher 4.4.4.4:120
  vpn-target 0:120 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:120 import-extcommunity
#
bridge-domain 30
 vxlan vni 130
 evpn
  route-distinguisher 4.4.4.4:130
  vpn-target 0:130 export-extcommunity
  vpn-target 0:1 export-extcommunity
```



```
 vpn-target 0:130 import-extcommunity
#
bridge-domain 40
 vxlan vni 140
 evpn
  route-distinguisher 4.4.4.4:140
  vpn-target 0:140 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:140 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0101
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.2.1.1 255.255.255.0
 ipv6 address FC00:2::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0102
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.3.1.1 255.255.255.0
 ipv6 address FC00:3::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0103
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif40
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.4.1.1 255.255.255.0
 ipv6 address FC00:4::1/64
 arp generate-rd-table enable
 arp broadcast-detect enable
 mac-address 00e0-fc00-0104
 ipv6 nd collect host enable
 ipv6 nd na glean
 ipv6 nd generate-rd-table enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.10.10.6 255.255.255.252
 reserved for vxlan bypass
```

```
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 1
 arp anti-attack rate-limit 200
#
interface Eth-Trunk10.30 mode l2
 encapsulation dot1q vid 30
 bridge-domain 30
#
interface Eth-Trunk11
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 2
 arp anti-attack rate-limit 200
#
interface Eth-Trunk11.40 mode l2
 encapsulation dot1q vid 40
 bridge-domain 40
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.4.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.8.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 eth-trunk 10
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/6
 eth-trunk 11
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
```

```
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface LoopBack0
 ip address 4.4.4.4 255.255.255.255
#
interface LoopBack1
 ip address 6.6.6.6 255.255.255.255
#
interface LoopBack2
 ip address 15.15.15.15 255.255.255.255
#
interface Nve1
 source 6.6.6.6
 pip-source 15.15.15.15 peer 14.14.14.14 bypass
 vni 110 head-end peer-list protocol bgp
 vni 120 head-end peer-list protocol bgp
 vni 130 head-end peer-list protocol bgp
 vni 140 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0112
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk11 downlink 2
 timer recover-time 60
#
bfd toipu3_v4 bind peer-ip 10.3.1.2 vpn-instance vpn1 interface Vbdif30 source-ip 10.3.1.1 one-arm-
echo
 discriminator local 20
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu3_v6 bind peer-ipv6 FC00:3::2 vpn-instance vpn1 interface Vbdif30 source-ipv6 FC00:3::1 one-
arm-echo
 discriminator local 60
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu4_v4 bind peer-ip 10.4.1.2 vpn-instance vpn1 interface Vbdif40 source-ip 10.4.1.1 one-arm-
echo
 discriminator local 40
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu4_v6 bind peer-ipv6 FC00:4::2 vpn-instance vpn1 interface Vbdif40 source-ipv6 FC00:4::1 one-
arm-echo
 discriminator local 80
 detect-multiplier 6
 min-echo-rx-interval 300
#
bgp 100
 router-id 4.4.4.4
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
 #
 ipv4-family vpn-instance vpn1
   import-route static route-policy import_static_policy
  maximum load-balancing 64
  irb asymmetric
  advertise l2vpn evpn import-route-multipath
 #
```

```
 ipv6-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing 64
  irb asymmetric
  advertise l2vpn evpn import-route-multipath
 #
 l2vpn-family evpn
  policy vpn-target
  bestroute add-path path-number 64
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy export_policy export
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 capability-advertise add-path both
  peer 7.7.7.7 advertise add-path path-number 64
  peer 8.8.8.8 enable
  peer 8.8.8.8 route-policy export_policy export
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 capability-advertise add-path both
  peer 8.8.8.8 advertise add-path path-number 64
#
ospf 1 router-id 4.4.4.4
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
  area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.4.0 0.0.0.255
  network 192.168.8.0 0.0.0.255
#
ip community-filter basic apply_gwip_route index 10 permit 82345
ip community-filter basic suppress_route index 10 permit 82346
#
route-policy export_policy deny node 10
 if-match community-filter suppress_route
#
route-policy export_policy permit node 20
 if-match community-filter apply_gwip_route
 apply gateway-ip origin-nexthop
 apply ipv6 gateway-ip origin-nexthop
#
route-policy export_policy permit node 999
#
route-policy import_static_policy permit node 100
 if-match tag 82345
 apply community 82345 additive
#
route-policy import_static_policy permit node 999
#
ip route-static 14.14.14.14 32 10.10.10.5 preference 1
ip route-static vpn-instance vpn1 172.16.2.1 255.255.255.255 10.3.1.2 preference 255 tag 82345 track
bfd-session toipu3_v4 inter-protocol-ecmp
ip route-static vpn-instance vpn1 172.16.2.1 255.255.255.255 10.4.1.2 preference 255 tag 82345 track
bfd-session toipu4_v4 inter-protocol-ecmp
#
ipv6 route-static vpn-instance vpn1 2001:db8:2::1 128 FC00:3::2 preference 255 tag 82345 track bfd-
session toipu3_v6 inter-protocol-ecmp
ipv6 route-static vpn-instance vpn1 2001:db8:2::1 128 FC00:4::2 preference 255 tag 82345 track bfd-
session toipu4_v6 inter-protocol-ecmp
#
return
```

1.7 配置 NFVI 分布式网关示例（对称式）

适用产品和版本

- V300R021C10及之后版本：CE8851、CE6866可以部署为该组网中的Leaf设备，CE16800（除X系列单板外）、CE8851可以部署为该组网中的Spine设备。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

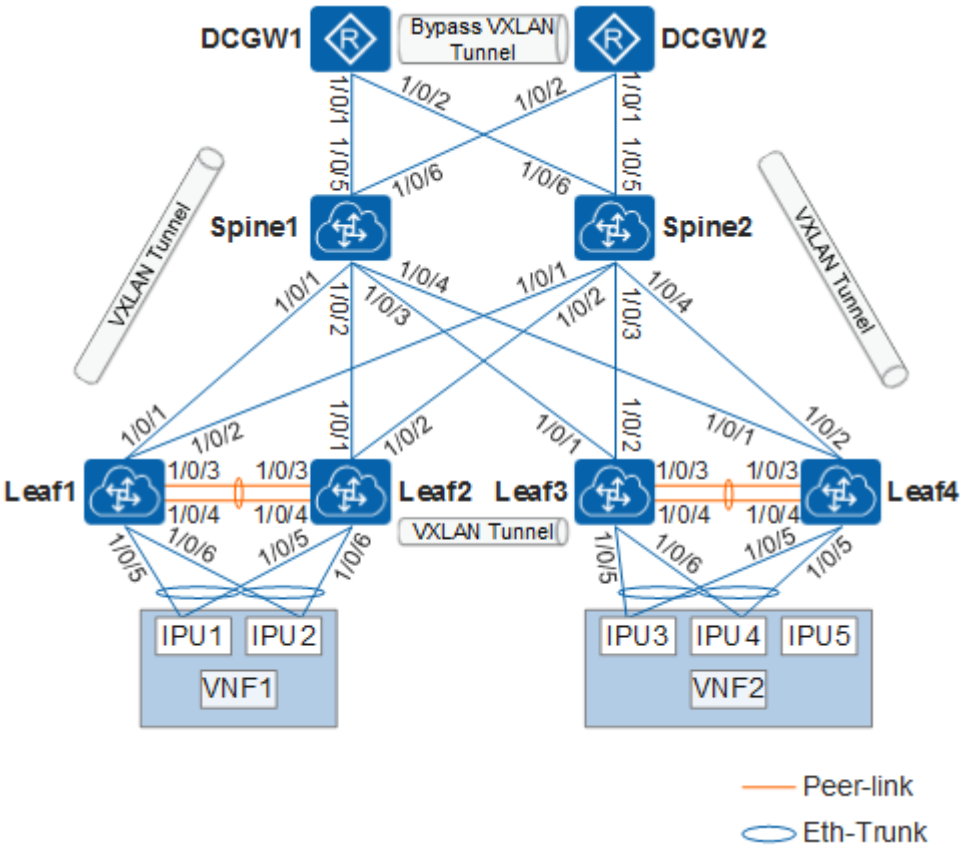
NFVI电信云解决方案是DCI（Data Center Interconnect）+DCN（Data Communication Network）的组网方案。其中大量手机业务流量会进入DCN网络并访问DCN网络内的vUGW与vMSE。经过vUGW与vMSE的处理后，这些手机业务流量再次从DCN网络转发出去，继续访问Internet中的目的设备。同样目的设备发往手机的回应流量亦要经历该过程。为了实现上述功能，并且确保手机业务流量在DCN网络内部可以实现负载均衡，则需要在DCN网络内部部署NFVI分布式网关功能。

如**配置IPv4 NFVI分布式网关组网图**所示，该组网为NFVI分布式网关的组网示意图。其中DCGW1和DCGW2为DCN网络的边界网关，可以和外部网络交换Internet路由。Leaf用于接入VNF（Virtualized Network Function）。VNF1和VNF2作为虚拟化网元可以分别部署并实现vUGW和vMSE的功能，并通过IPU（Interface Process Unit）与Leaf连接。

该组网可以看成分布式网关功能和VXLAN双活网关功能拼接组成：

- DCGW1和DCGW2上部署VXLAN双活网关功能，即DCGW1和DCGW2之间建立Bypass VXLAN隧道，同时DCGW1和DCGW2共同使用一个虚拟的Anycast VTEP地址分别与Leaf建立VXLAN隧道。
- Spine作为透传节点，连接DCGW和Leaf。
- Leaf上部署分布式网关功能并在两个M-LAG组间创建VXLAN隧道。

图 1-7 配置 NFVI 分布式网关组网图



说明

上图中“1/0/1”为接口编号，接口速率为100GE，即“1/0/1”表示接口“100GE1/0/1”。其他接口类似。

表 1-10 接口地址表

设备名称	接口	IP地址	设备名称	接口	IP地址
DCGW1	100GE1/0/1	192.168.9.2/24	DCGW2	100GE1/0/1	192.168.11.2/24
	100GE1/0/2	192.168.12.2/24		100GE1/0/2	192.168.10.2/24
	Loopback0	9.9.9.9/32		Loopback0	10.10.10.10/32
	Loopback1	11.11.11.11/32		Loopback1	11.11.11.11/32
Spine1	100GE1/0/1	192.168.1.1/24	Spine2	100GE1/0/1	192.168.5.1/24

设备名称	接口	IP地址	设备名称	接口	IP地址
	100GE1/0/2	192.168.2.1/24		100GE1/0/2	192.168.6.1/24
	100GE1/0/3	192.168.3.1/24		100GE1/0/3	192.168.7.1/24
	100GE1/0/4	192.168.4.1/24		100GE1/0/4	192.168.8.1/24
	100GE1/0/5	192.168.9.1/24		100GE1/0/5	192.168.11.1/24
	100GE1/0/6	192.168.10.1/24		100GE1/0/6	192.168.12.1/24
	Loopback0	7.7.7.7/32		Loopback0	8.8.8.8/32
Leaf1	100GE1/0/1	192.168.1.2/24	Leaf2	100GE1/0/1	192.168.2.2/24
	100GE1/0/2	192.168.5.2/24		100GE1/0/2	192.168.6.2/24
	Loopback0	1.1.1.1/32		Loopback0	2.2.2.2/32
	Loopback1	5.5.5.5/32		Loopback1	5.5.5.5/32
	Loopback2	12.12.12.12/32		Loopback2	13.13.13.13/32
Leaf3	100GE1/0/1	192.168.3.2/24	Leaf4	100GE1/0/1	192.168.4.2/24
	100GE1/0/2	192.168.7.2/24		100GE1/0/2	192.168.8.2/24
	Loopback0	3.3.3.3/32		Loopback0	4.4.4.4/32
	Loopback1	6.6.6.6/32		Loopback1	6.6.6.6/32
	Loopback2	14.14.14.14/32		Loopback2	15.15.15.15/32
IPU1	-	IPv4: 10.1.1.2/24 IPv6: fc00:1::2/64	IPU2	-	IPv4: 10.2.1.2/24 IPv6: fc00:2::2/64
IPU3	-	IPv4: 10.3.1.2/24 IPv6: fc00:3::2/64	IPU4	-	IPv4: 10.4.1.2/24 IPv6: fc00:4::2/64

设备名称	接口	IP地址	设备名称	接口	IP地址
VNF1	-	IPv4: 172.16.1.1/32 IPv6: 2001:db8:1::1/128	VNF2	-	IPv4: 172.16.2.1/32 IPv6: 2001:db8:2::1/128

配置思路

采用如下的思路配置：

1. 配置路由协议，保证网络三层互通。
2. 配置Leaf组建M-LAG。
3. 配置BGP EVPN，建立VXLAN隧道。
4. 配置Leaf至VNF的静态路由并通过BGP EVPN发布。
5. 配置路由负载分担。

操作步骤

步骤1 配置各接口的IP地址及Loopback接口的地址，并配置路由协议，保证网络三层互通。本示例采用了OSPF路由协议。

配置Leaf1。其他设备的配置与Leaf1类似，这里不再赘述。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf1
[*HUAWEI] commit
[~Leaf1] bfd  //全局使能BFD
[*Leaf1-bfd] quit
[*Leaf1] interface 100ge 1/0/1
[*Leaf1-100GE1/0/1] undo portswitch
[*Leaf1-100GE1/0/1] ip address 192.168.1.2 24
[*Leaf1-100GE1/0/1] ospf network-type p2p
[*Leaf1-100GE1/0/1] ospf peer hold-max-cost timer 800000
[*Leaf1-100GE1/0/1] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/1] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/1] qos phb marking dscp enable
[*Leaf1-100GE1/0/1] quit
[*Leaf1] interface 100ge 1/0/2
[*Leaf1-100GE1/0/2] undo portswitch
[*Leaf1-100GE1/0/2] ip address 192.168.5.2 24
[*Leaf1-100GE1/0/2] ospf network-type p2p
[*Leaf1-100GE1/0/2] ospf peer hold-max-cost timer 800000
[*Leaf1-100GE1/0/2] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/2] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/2] qos phb marking dscp enable
[*Leaf1-100GE1/0/2] quit
[*Leaf1] interface loopback 0
[*Leaf1-LoopBack0] ip address 1.1.1.1 32
[*Leaf1-LoopBack0] quit
[*Leaf1] interface loopback 1
[*Leaf1-LoopBack1] ip address 5.5.5.5 32
[*Leaf1-LoopBack1] quit
[*Leaf1] interface loopback 2
[*Leaf1-LoopBack2] ip address 12.12.12.12 32
[*Leaf1-LoopBack2] quit
```

```
[*Leaf1] ospf 1 router-id 1.1.1.1
[*Leaf1-ospf-1] spf-schedule-interval intelligent-timer 50 50 50
[*Leaf1-ospf-1] lsa-originate-interval intelligent-timer 500 50 100
[*Leaf1-ospf-1] lsa-arrival-interval intelligent-timer 50 50 50
[*Leaf1-ospf-1] bfd all-interfaces enable
[*Leaf1-ospf-1] bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
[*Leaf1-ospf-1] area 0
[*Leaf1-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[*Leaf1-ospf-1-area-0.0.0.0] network 192.168.5.0 0.0.0.255
[*Leaf1-ospf-1-area-0.0.0.0] network 1.1.1.1 0.0.0.0
[*Leaf1-ospf-1-area-0.0.0.0] network 5.5.5.5 0.0.0.0
[*Leaf1-ospf-1-area-0.0.0.0] quit
[*Leaf1-ospf-1] quit
[*Leaf1] commit
```

步骤2 配置Leaf组建M-LAG。

配置Leaf1和Leaf2组建M-LAG。Leaf3和Leaf4的配置与之类似，这里不再赘述。

```
[~Leaf1] stp mode rstp
[*Leaf1] stp v-stp enable
[*Leaf1] dfs-group 1
[*Leaf1-dfs-group-1] dual-active detection source ip 1.1.1.1 peer 2.2.2.2
[*Leaf1-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206  //在组建M-LAG的设备上均需配置本命令，请保持M-LAG组内的认证密码一致
[*Leaf1-dfs-group-1] priority 150
[*Leaf1-dfs-group-1] m-lag up-delay 240 auto-recovery interval 10  //配置M-LAG成员接口上报Up状态的延时时间，以及延迟自动恢复时间
[*Leaf1-dfs-group-1] dual-active detection delay 0  //配置peer-link故障时双主检测链路的检测时间为0，防止peer-link故障时BFD会话震荡
[*Leaf1-dfs-group-1] quit
[*Leaf1] interface eth-trunk 1  //配置peer-link
[*Leaf1-Eth-Trunk1] trunkport 100ge 1/0/3
[*Leaf1-Eth-Trunk1] trunkport 100ge 1/0/4
[*Leaf1-Eth-Trunk1] mode lacp-static
[*Leaf1-Eth-Trunk1] peer-link 1
[*Leaf1-Eth-Trunk1] port vlan exclude 1  //配置peer-link接口不允许通过VLAN1
[*Leaf1-Eth-Trunk1] quit
[*Leaf1] interface 100ge 1/0/3
[*Leaf1-100GE1/0/3] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/3] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/3] quit
[*Leaf1] interface 100ge 1/0/4
[*Leaf1-100GE1/0/4] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/4] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/4] quit
[*Leaf1] interface eth-trunk 10  //配置连接VNF的M-LAG成员接口
[*Leaf1-Eth-Trunk10] trunkport 100ge 1/0/5
[*Leaf1-Eth-Trunk10] mode lacp-static
[*Leaf1-Eth-Trunk10] lacp timeout fast
[*Leaf1-Eth-Trunk10] dfs-group 1 m-lag 1
[*Leaf1-Eth-Trunk10] arp anti-attack rate-limit 200  //指定ARP报文的限速值，每秒允许通过的ARP报文的个数为200
[*Leaf1-Eth-Trunk10] stp edged-port enable
[*Leaf1-Eth-Trunk10] quit
[*Leaf1] interface eth-trunk 11
[*Leaf1-Eth-Trunk11] trunkport 100ge 1/0/6
[*Leaf1-Eth-Trunk11] mode lacp-static
[*Leaf1-Eth-Trunk11] lacp timeout fast
[*Leaf1-Eth-Trunk11] dfs-group 1 m-lag 2
[*Leaf1-Eth-Trunk11] arp anti-attack rate-limit 200  //指定ARP报文的限速值，每秒允许通过的ARP报文的个数为200
[*Leaf1-Eth-Trunk11] stp edged-port enable
[*Leaf1-Eth-Trunk11] quit
[*Leaf1] commit
[~Leaf1] monitor-link group 1
[*Leaf1-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf1-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf1-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf1-mtlk-group1] port eth-trunk 11 downlink 2
```



```
[*Leaf1-mtlk-group1] timer recover-time 60
[*Leaf1-mtlk-group1] quit
[*Leaf1] interface 100ge 1/0/5
[*Leaf1-100GE1/0/5] storm suppression unknown-unicast 5  //配置接入交换机端口未知单播抑制功能
[*Leaf1-100GE1/0/5] storm suppression multicast packets 1000  //配置接入交换机端口未知组播抑制功能
[*Leaf1-100GE1/0/5] storm suppression broadcast packets 1000  //配置接入交换机端口广播抑制功能
[*Leaf1-100GE1/0/5] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/5] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/5] quit
[*Leaf1] interface 100ge 1/0/6
[*Leaf1-100GE1/0/6] storm suppression unknown-unicast 5
[*Leaf1-100GE1/0/6] storm suppression multicast packets 1000
[*Leaf1-100GE1/0/6] storm suppression broadcast packets 1000
[*Leaf1-100GE1/0/6] port crc-statistics trigger error-down
[*Leaf1-100GE1/0/6] trap-threshold crc-statistics 100 interval 10
[*Leaf1-100GE1/0/6] quit
[*Leaf1] commit
[~Leaf2] stp mode rstp
[*Leaf2] stp v-stp enable
[*Leaf2] dfs-group 1
[*Leaf2-dfs-group-1] dual-active detection source ip 2.2.2.2 peer 1.1.1.1
[*Leaf2-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Leaf2-dfs-group-1] m-lag up-delay 240 auto-recovery interval 10
[*Leaf2-dfs-group-1] dual-active detection delay 0
[*Leaf2-dfs-group-1] quit
[*Leaf2] interface eth-trunk 1
[*Leaf2-Eth-Trunk1] trunkport 100ge 1/0/3
[*Leaf2-Eth-Trunk1] trunkport 100ge 1/0/4
[*Leaf2-Eth-Trunk1] mode lacp-static
[*Leaf2-Eth-Trunk1] peer-link 1
[*Leaf2-Eth-Trunk1] port vlan exclude 1   //配置peer-link接口不允许通过VLAN1
[*Leaf2-Eth-Trunk1] quit
[*Leaf2] interface eth-trunk 10
[*Leaf2-Eth-Trunk10] trunkport 100ge 1/0/5
[*Leaf2-Eth-Trunk10] mode lacp-static
[*Leaf2-Eth-Trunk10] lacp timeout fast
[*Leaf2-Eth-Trunk10] dfs-group 1 m-lag 1
[*Leaf2-Eth-Trunk10] arp anti-attack rate-limit 200
[*Leaf2-Eth-Trunk10] stp edged-port enable
[*Leaf2-Eth-Trunk10] quit
[*Leaf2] interface eth-trunk 11
[*Leaf2-Eth-Trunk11] trunkport 100ge 1/0/6
[*Leaf2-Eth-Trunk11] mode lacp-static
[*Leaf2-Eth-Trunk11] lacp timeout fast
[*Leaf2-Eth-Trunk11] dfs-group 1 m-lag 2
[*Leaf2-Eth-Trunk11] arp anti-attack rate-limit 200
[*Leaf2-Eth-Trunk11] stp edged-port enable
[*Leaf2-Eth-Trunk11] quit
[*Leaf2] commit
[~Leaf2] monitor-link group 1
[*Leaf2-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf2-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf2-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf2-mtlk-group1] port eth-trunk 11 downlink 2
[*Leaf2-mtlk-group1] timer recover-time 60
[*Leaf2-mtlk-group1] quit
[*Leaf2] interface 100ge 1/0/5
[*Leaf2-100GE1/0/5] storm suppression unknown-unicast 5
[*Leaf2-100GE1/0/5] storm suppression multicast packets 1000
[*Leaf2-100GE1/0/5] storm suppression broadcast packets 1000
[*Leaf2-100GE1/0/5] quit
[*Leaf2] interface 100ge 1/0/6
[*Leaf2-100GE1/0/6] storm suppression unknown-unicast 5
[*Leaf2-100GE1/0/6] storm suppression multicast packets 1000
[*Leaf2-100GE1/0/6] storm suppression broadcast packets 1000
[*Leaf2-100GE1/0/6] quit
[*Leaf2] commit
```

步骤3 在M-LAG设备中配置静态Bypass VXLAN隧道。

在M-LAG双归接入VXLAN的场景中，当下行一条链路发生故障时，业务流量需绕行M-LAG设备之间的Peer-link链路。因此，在该场景下M-LAG设备之间必须配置静态Bypass VXLAN隧道，将绕行的业务流量引导至Peer-link链路上。

下面以Leaf1和Leaf2配置为例，Leaf3、Leaf4的配置与之类似，这里不再赘述，具体配置请参考配置脚本。

配置Leaf1和Leaf2。

```
[~Leaf1] vlan 100  //本VLAN不能划分给其他业务使用，本例中以100举例
[*Leaf1-vlan100] m-lag peer-link reserved  //仅允许peer-link加入到该VLAN
[*Leaf1-vlan100] quit
[*Leaf1] interface vlanif 100
[*Leaf1-Vlanif100] reserved for vxlan bypass  //指定peer-link接口上VLANIF的IPv4地址只给Bypass VXLAN隧道使用
[*Leaf1-Vlanif100] ip address 10.10.10.1 30  //配置静态Bypass VXLAN隧道的源端IPv4地址
[*Leaf1-Vlanif100] quit
[*Leaf1] ip route-static 13.13.13.13 32 10.10.10.2 preference 1  //配置静态路由，打通Bypass VXLAN隧道
[*Leaf1] commit
[*Leaf1] interface nve 1
[*Leaf1-Nve1] pip-source 12.12.12.12 peer 13.13.13.13 bypass  //创建静态Bypass VXLAN隧道，指定源端地址和对端地址
[*Leaf1-Nve1] quit
[*Leaf1] commit
[~Leaf2] vlan 100
[*Leaf2-vlan100] m-lag peer-link reserved
[*Leaf2-vlan100] quit
[*Leaf2] interface vlanif 100
[*Leaf2-Vlanif100] reserved for vxlan bypass
[*Leaf2-Vlanif100] ip address 10.10.10.2 30
[*Leaf2-Vlanif100] quit
[*Leaf2] ip route-static 12.12.12.12 32 10.10.10.1 preference 1
[*Leaf2] commit
[*Leaf2] interface nve 1
[*Leaf2-Nve1] pip-source 13.13.13.13 peer 12.12.12.12 bypass
[*Leaf2-Nve1] quit
[*Leaf2] commit
```

步骤4 配置BGP EVPN，建立VXLAN隧道。

1. 配置BGP EVPN对等体关系。Spine1和Spine2作为leaf节点路由反射器。

   # 配置Spine1。Spine2的配置与Spine1类似，这里不再赘述。

```
[~Spine1] evpn-overlay enable
[*Spine1] bgp 100
[*Spine1] router-id 7.7.7.7
[*Spine1-bgp] peer 1.1.1.1 as-number 100
[*Spine1-bgp] peer 1.1.1.1 connect-interface LoopBack0
[*Spine1-bgp] peer 2.2.2.2 as-number 100
[*Spine1-bgp] peer 2.2.2.2 connect-interface LoopBack0
[*Spine1-bgp] peer 3.3.3.3 as-number 100
[*Spine1-bgp] peer 3.3.3.3 connect-interface LoopBack0
[*Spine1-bgp] peer 4.4.4.4 as-number 100
[*Spine1-bgp] peer 4.4.4.4 connect-interface LoopBack0
[*Spine1-bgp] peer 9.9.9.9 as-number 100
[*Spine1-bgp] peer 9.9.9.9 connect-interface LoopBack0
[*Spine1-bgp] peer 10.10.10.10 as-number 100
[*Spine1-bgp] peer 10.10.10.10 connect-interface LoopBack0
[*Spine1-bgp] l2vpn-family evpn
[*Spine1-bgp-af-evpn] bestroute add-path path-number 64  //使能BGP ADD-PATH功能
[*Spine1-bgp-af-evpn] peer 1.1.1.1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 1.1.1.1 reflect-client
[*Spine1-bgp-af-evpn] peer 1.1.1.1 advertise irb
[*Spine1-bgp-af-evpn] peer 1.1.1.1 advertise irbv6
[*Spine1-bgp-af-evpn] peer 1.1.1.1 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 1.1.1.1 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 2.2.2.2 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
```



```
[*Spine1-bgp-af-evpn] peer 2.2.2.2 advertise irb
[*Spine1-bgp-af-evpn] peer 2.2.2.2 advertise irbv6
[*Spine1-bgp-af-evpn] peer 2.2.2.2 reflect-client
[*Spine1-bgp-af-evpn] peer 2.2.2.2 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 2.2.2.2 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 3.3.3.3 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 3.3.3.3 advertise irb
[*Spine1-bgp-af-evpn] peer 3.3.3.3 advertise irbv6
[*Spine1-bgp-af-evpn] peer 3.3.3.3 reflect-client
[*Spine1-bgp-af-evpn] peer 3.3.3.3 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 3.3.3.3 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 4.4.4.4 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 4.4.4.4 advertise irb
[*Spine1-bgp-af-evpn] peer 4.4.4.4 advertise irbv6
[*Spine1-bgp-af-evpn] peer 4.4.4.4 reflect-client
[*Spine1-bgp-af-evpn] peer 4.4.4.4 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 4.4.4.4 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 9.9.9.9 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 9.9.9.9 advertise irb
[*Spine1-bgp-af-evpn] peer 9.9.9.9 advertise irbv6
[*Spine1-bgp-af-evpn] peer 9.9.9.9 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 9.9.9.9 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] peer 10.10.10.10 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Spine1-bgp-af-evpn] peer 10.10.10.10 advertise irb
[*Spine1-bgp-af-evpn] peer 10.10.10.10 advertise irbv6
[*Spine1-bgp-af-evpn] peer 10.10.10.10 capability-advertise add-path both
[*Spine1-bgp-af-evpn] peer 10.10.10.10 advertise add-path path-number 64
[*Spine1-bgp-af-evpn] undo policy vpn-target
[*Spine1-bgp-af-evpn] quit
[*Spine1-bgp] quit
[*Spine1] commit
```

配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] evpn-overlay enable
[*Leaf1] bgp 100
[*Leaf1] router-id 1.1.1.1
[*Leaf1-bgp] peer 7.7.7.7 as-number 100
[*Leaf1-bgp] peer 7.7.7.7 connect-interface LoopBack0
[*Leaf1-bgp] peer 8.8.8.8 as-number 100
[*Leaf1-bgp] peer 8.8.8.8 connect-interface LoopBack0
[*Leaf1-bgp] l2vpn-family evpn
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 advertise irb
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 advertise irbv6
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 advertise irb
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 advertise irbv6
[*Leaf1-bgp-af-evpn] quit
[*Leaf1-bgp] quit
[*Leaf1] commit
```

2. 配置VPN实例及EVPN实例。

   # 配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] ip vpn-instance vpn1
[*Leaf1-vpn-instance-vpn1] vxlan vni 200
[*Leaf1-vpn-instance-vpn1] ipv4-family
[*Leaf1-vpn-instance-vpn1-af-ipv4] route-distinguisher 1.1.1.1:1
[*Leaf1-vpn-instance-vpn1-af-ipv4] vpn-target 0:1 evpn
[*Leaf1-vpn-instance-vpn1-af-ipv4] quit
[*Leaf1-vpn-instance-vpn1] ipv6-family
[*Leaf1-vpn-instance-vpn1-af-ipv6] route-distinguisher 1.1.1.1:1
[*Leaf1-vpn-instance-vpn1-af-ipv6] vpn-target 0:1 evpn
[*Leaf1-vpn-instance-vpn1-af-ipv6] quit
```

```
[*Leaf1-vpn-instance-vpn1] quit
[*Leaf1] bridge-domain 10
[*Leaf1-bd10] vxlan vni 110
[*Leaf1-bd10] evpn
[*Leaf1-bd10-evpn] route-distinguisher 1.1.1.1:110
[*Leaf1-bd10-evpn] vpn-target 0:110
[*Leaf1-bd10-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd10-evpn] quit
[*Leaf1-bd10] quit
[*Leaf1] bridge-domain 20
[*Leaf1-bd20] vxlan vni 120
[*Leaf1-bd20] evpn
[*Leaf1-bd20-evpn] route-distinguisher 1.1.1.1:120
[*Leaf1-bd20-evpn] vpn-target 0:120
[*Leaf1-bd20-evpn] vpn-target 0:1 export-extcommunity
[*Leaf1-bd20-evpn] quit
[*Leaf1-bd20] quit
[*Leaf1] interface nve 1
[*Leaf1-Nve1] source 5.5.5.5  //组建M-LAG的两台设备上配置的NVE接口的IP地址和MAC地址需要相同
[*Leaf1-Nve1] mac-address 00e0-fc00-0111
[*Leaf1-Nve1] vni 110 head-end peer-list protocol bgp
[*Leaf1-Nve1] vni 120 head-end peer-list protocol bgp
[*Leaf1-Nve1] quit
[*Leaf1] commit
```

3. 配置三层网关。

   # 配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] interface vbdif10  //Leaf和DCGW上同一VBDIF的IP地址和MAC地址需要相同
[*Leaf1-Vbdif10] ip binding vpn-instance vpn1
[*Leaf1-Vbdif10] ip address 10.1.1.1 24
[*Leaf1-Vbdif10] ipv6 enable
[*Leaf1-Vbdif10] ipv6 address fc00:1::1 64
[*Leaf1-Vbdif10] arp broadcast-detect enable
[*Leaf1-Vbdif10] ipv6 nd na glean
[*Leaf1-Vbdif10] vxlan anycast-gateway enable
[*Leaf1-Vbdif10] arp collect host enable
[*Leaf1-Vbdif10] ipv6 nd collect host enable
[*Leaf1-Vbdif10] mac-address 00e0-fc00-0101
[*Leaf1-Vbdif10] quit
[*Leaf1] interface vbdif20
[*Leaf1-Vbdif20] ip binding vpn-instance vpn1
[*Leaf1-Vbdif20] ip address 10.2.1.1 24
[*Leaf1-Vbdif20] ipv6 enable
[*Leaf1-Vbdif20] ipv6 address fc00:2::1 64
[*Leaf1-Vbdif20] arp broadcast-detect enable
[*Leaf1-Vbdif20] ipv6 nd na glean
[*Leaf1-Vbdif20] vxlan anycast-gateway enable
[*Leaf1-Vbdif20] arp collect host enable
[*Leaf1-Vbdif20] ipv6 nd collect host enable
[*Leaf1-Vbdif20] mac-address 00e0-fc00-0102
[*Leaf1-Vbdif20] quit
[*Leaf1] commit
```

4. 配置业务接入点

   # 配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] interface eth-trunk 10.10 mode l2
[*Leaf1-Eth-Trunk10.10] encapsulation dot1q vid 10
[*Leaf1-Eth-Trunk10.10] bridge-domain 10
[*Leaf1-Eth-Trunk10.10] quit
[*Leaf1] interface eth-trunk 11.20 mode l2
[*Leaf1-Eth-Trunk11.20] encapsulation dot1q vid 20
[*Leaf1-Eth-Trunk11.20] bridge-domain 20
[*Leaf1-Eth-Trunk11.20] quit
[*Leaf1] commit
```

步骤5 在Leaf上配置BFD，检测Leaf和VNF之间链路。

配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] bfd toipu1_v4 bind peer-ip 10.1.1.2 vpn-instance vpn1 interface vbdif10 source-ip 10.1.1.1 one-arm-echo
[*Leaf1-bfd-session-toipu1_v4] discriminator local 10
[*Leaf1-bfd-session-toipu1_v4] detect-multiplier 6
[*Leaf1-bfd-session-toipu1_v4] min-echo-rx-interval 300
[*Leaf1-bfd-session-toipu1_v4] quit
[*Leaf1] bfd toipu2_v4 bind peer-ip 10.2.1.2 vpn-instance vpn1 interface vbdif20 source-ip 10.2.1.1 one-arm-echo
[*Leaf1-bfd-session-toipu2_v4] discriminator local 30
[*Leaf1-bfd-session-toipu2_v4] detect-multiplier 6
[*Leaf1-bfd-session-toipu2_v4] min-echo-rx-interval 300
[*Leaf1-bfd-session-toipu2_v4] quit
[*Leaf1] bfd toipu1_v6 bind peer-ipv6 fc00:1::2 vpn-instance vpn1 interface vbdif10 source-ipv6 fc00:1::1 one-arm-echo
[*Leaf1-bfd-session-toipu1_v6] discriminator local 50
[*Leaf1-bfd-session-toipu1_v6] detect-multiplier 6
[*Leaf1-bfd-session-toipu1_v6] min-echo-rx-interval 300
[*Leaf1-bfd-session-toipu1_v6] quit
[*Leaf1] bfd toipu2_v6 bind peer-ipv6 fc00:2::2 vpn-instance vpn1 interface vbdif20 source-ipv6 fc00:2::1 one-arm-echo
[*Leaf1-bfd-session-toipu2_v6] discriminator local 70
[*Leaf1-bfd-session-toipu2_v6] detect-multiplier 6
[*Leaf1-bfd-session-toipu2_v6] min-echo-rx-interval 300
[*Leaf1-bfd-session-toipu2_v6] quit
[*Leaf1] bfd
[*Leaf1-bfd] bfd forwarding match remote-discriminator 20  //指定M-LAG场景下的BFD远端标识符,即对应Leaf2的discriminator local值
[*Leaf1-bfd] bfd forwarding match remote-discriminator 40
[*Leaf1-bfd] bfd forwarding match remote-discriminator 60
[*Leaf1-bfd] bfd forwarding match remote-discriminator 80
[*Leaf1-bfd] quit
[*Leaf1] commit
```

步骤6 配置Leaf通往VNF的私网静态路由，并配置BGP EVPN引入私网静态路由，然后配置L3VPN实例应用路由策略，使这些静态私网路由保持原有下一跳。

配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] route-policy import_static_policy permit node 100  //配置一个路由策略，在将静态路由引入BGP时增加团体属性
[*Leaf1-route-policy] if-match tag 82345
[*Leaf1-route-policy] apply community 82345 additive
[*Leaf1-route-policy] quit
[*Leaf1] route-policy import_static_policy permit node 999
[*Leaf1-route-policy] quit
[*Leaf1] ip community-filter basic apply_gwip_route index 10 permit 82345  //创建团体属性过滤器
[*Leaf1] ip community-filter basic suppress_route index 10 permit 82346
[*Leaf1] route-policy export_policy deny node 10  //配置一个路由策略，用于过滤/允许路由通过
[*Leaf1-route-policy] if-match community-filter suppress_route
[*Leaf1-route-policy] quit
[*Leaf1] route-policy export_policy permit node 20
[*Leaf1-route-policy] if-match community-filter apply_gwip_route
[*Leaf1-route-policy] apply gateway-ip origin-nexthop  //设置路由的下一跳地址作为网关IP地址。在L2GW/L3GW上配置L3VPN实例向EVPN发布可以到达VNF的私网静态路由时，需要先创建路由策略，该路由策略可以过滤出L3VPN实例中可以到达VNF的私网静态路由。
[*Leaf1-route-policy] apply ipv6 gateway-ip origin-nexthop
[*Leaf1-route-policy] quit
[*Leaf1] route-policy export_policy permit node 999
[*Leaf1-route-policy] quit
[*Leaf1] ip route-static vpn-instance vpn1 172.16.1.1 32 10.1.1.2 preference 255 tag 82345 track bfd-session toipu1_v4 inter-protocol-ecmp  //此处配置的Tag值需要与路由策略相匹配，以实现Leaf将路由发布到DCGW时路由的下一跳地址为网关IP地址，实现非对称转发
[*Leaf1] ip route-static vpn-instance vpn1 172.16.1.1 32 10.2.1.2 preference 255 tag 82345 track bfd-session toipu2_v4 inter-protocol-ecmp
[*Leaf1] ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 fc00:1::2 preference 255 tag 82345 track bfd-session toipu1_v6 inter-protocol-ecmp
[*Leaf1] ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 fc00:2::2 preference 255 tag 82345 track bfd-session toipu2_v6 inter-protocol-ecmp
[*Leaf1] bgp 100
[*Leaf1-bgp] ipv4-family vpn-instance vpn1
```

```
[*Leaf1-bgp-vpn1] import-route static route-policy import_static_policy
[*Leaf1-bgp-vpn1] advertise l2vpn evpn import-route-multipath    //发布所有目的地址相同的路由
[*Leaf1-bgp-vpn1] quit
[*Leaf1-bgp] ipv6-family vpn-instance vpn1
[*Leaf1-bgp-6-vpn1] import-route static route-policy import_static_policy
[*Leaf1-bgp-6-vpn1] advertise l2vpn evpn import-route-multipath
[*Leaf1-bgp-6-vpn1] quit
[*Leaf1-bgp] quit
[*Leaf1] commit
```

步骤7 配置路由负载分担。

配置Leaf1。Leaf2、Leaf3和Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] bgp 100
[*Leaf1-bgp] ipv4-family vpn-instance vpn1
[*Leaf1-bgp-vpn1] maximum load-balancing mixed 64  //配置BGP与其他协议路由形成负载分担的最大等价路由条数
[*Leaf1-bgp-vpn1] quit
[*Leaf1-bgp] ipv6-family vpn-instance vpn1
[*Leaf1-bgp-6-vpn1] maximum load-balancing mixed 64
[*Leaf1-bgp-6-vpn1] quit
[*Leaf1-bgp] l2vpn-family evpn
[*Leaf1-bgp-af-evpn] bestroute add-path path-number 64  //使能BGP ADD-PATH功能
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 capability-advertise add-path both
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 advertise add-path path-number 64
[*Leaf1-bgp-af-evpn] peer 7.7.7.7 route-policy export_policy export  //对RR使用路由策略
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 capability-advertise add-path both
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 advertise add-path path-number 64
[*Leaf1-bgp-af-evpn] peer 8.8.8.8 route-policy export_policy export
[*Leaf1-bgp-af-evpn] quit
[*Leaf1-bgp] quit
[*Leaf1] load-balance ecmp
[*Leaf1-ecmp] vxlan-overlay network local-preference enable  //配置VXLAN网络侧流量命中路由后只往本地出接口（非VXLAN类型出接口）转发
[*Leaf1-ecmp] commit
```

----结束

检查配置结果

配置完成后，在Leaf上执行**display ip routing-table vpn-instance vpn1**命令可以看到从DCGW收到的路由，其下一跳为Anycast VTEP地址。以Leaf1为例：

```
[~Leaf1] display ip routing-table vpn-instance vpn1
Proto: Protocol        Pre: Preference
Route Flags: R - relay, D - download to fib, T - to vpn-instance, B - black hole route
------------------------------------------------------------------------------
Routing Table : vpn1
        Destinations : 16       Routes : 18

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

      0.0.0.0/0   IBGP    255  0           RD  11.11.11.11     VXLAN
     10.1.1.0/24  Direct  0    0           D   10.1.1.1        Vbdif10
     10.1.1.1/32  Direct  0    0           D   127.0.0.1       Vbdif10
   10.1.1.255/32  Direct  0    0           D   127.0.0.1       Vbdif10
     10.2.1.0/24  Direct  0    0           D   10.2.1.1        Vbdif20
     10.2.1.1/32  Direct  0    0           D   127.0.0.1       Vbdif20
   10.2.1.255/32  Direct  0    0           D   127.0.0.1       Vbdif20
     10.3.1.0/24  Direct  0    0           D   10.3.1.1        Vbdif30
     10.3.1.1/32  Direct  0    0           D   127.0.0.1       Vbdif30
   10.3.1.255/32  Direct  0    0           D   127.0.0.1       Vbdif30
     10.4.1.0/24  Direct  0    0           D   10.4.1.1        Vbdif40
     10.4.1.1/32  Direct  0    0           D   127.0.0.1       Vbdif40
   10.4.1.255/32  Direct  0    0           D   127.0.0.1       Vbdif40
    172.16.1.1/32  Static  255  0           RD  10.1.1.2        Vbdif10
                   Static  255  0           RD  10.2.1.2        Vbdif20
```



```
    172.16.2.1/32  IBGP    255  0           RD  10.3.1.2        VXLAN
                   IBGP    255  0           RD  10.4.1.2        VXLAN
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

在DCGW上可以查看到VRF的路由表信息，下一跳为IPU的IP地址。

配置脚本

- Spine1的配置脚本

```
#
sysname Spine1
#
evpn-overlay enable
#
bfd
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.2.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.3.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 undo portswitch
 ip address 192.168.4.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 undo portswitch
 ip address 192.168.9.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
#
interface 100GE1/0/6
 undo portswitch
 ip address 192.168.10.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
#
interface LoopBack0
 ip address 7.7.7.7 255.255.255.255
#
bgp 100
 router-id 7.7.7.7
```

```
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack0
 peer 2.2.2.2 as-number 100
 peer 2.2.2.2 connect-interface LoopBack0
 peer 3.3.3.3 as-number 100
 peer 3.3.3.3 connect-interface LoopBack0
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack0
 peer 9.9.9.9 as-number 100
 peer 9.9.9.9 connect-interface LoopBack0
 peer 10.10.10.10 as-number 100
 peer 10.10.10.10 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 1.1.1.1 enable
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
  peer 4.4.4.4 enable
  peer 9.9.9.9 enable
  peer 10.10.10.10 enable
 #
 l2vpn-family evpn
  undo policy vpn-target
  bestroute add-path path-number 64
  peer 1.1.1.1 enable
  peer 1.1.1.1 advertise irb
  peer 1.1.1.1 advertise irbv6
  peer 1.1.1.1 reflect-client
  peer 1.1.1.1 capability-advertise add-path both
  peer 1.1.1.1 advertise add-path path-number 64
  peer 2.2.2.2 enable
  peer 2.2.2.2 advertise irb
  peer 2.2.2.2 advertise irbv6
  peer 2.2.2.2 reflect-client
  peer 2.2.2.2 capability-advertise add-path both
  peer 2.2.2.2 advertise add-path path-number 64
  peer 3.3.3.3 enable
  peer 3.3.3.3 advertise irb
  peer 3.3.3.3 advertise irbv6
  peer 3.3.3.3 reflect-client
  peer 3.3.3.3 capability-advertise add-path both
  peer 3.3.3.3 advertise add-path path-number 64
  peer 4.4.4.4 enable
  peer 4.4.4.4 advertise irb
  peer 4.4.4.4 advertise irbv6
  peer 4.4.4.4 reflect-client
  peer 4.4.4.4 capability-advertise add-path both
  peer 4.4.4.4 advertise add-path path-number 64
  peer 9.9.9.9 enable
  peer 9.9.9.9 advertise irb
  peer 9.9.9.9 advertise irbv6
  peer 9.9.9.9 capability-advertise add-path both
  peer 9.9.9.9 advertise add-path path-number 64
  peer 10.10.10.10 enable
  peer 10.10.10.10 advertise irb
  peer 10.10.10.10 advertise irbv6
  peer 10.10.10.10 capability-advertise add-path both
  peer 10.10.10.10 advertise add-path path-number 64
 #
 ospf 1 router-id 7.7.7.7
  bfd all-interfaces enable
  bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
  spf-schedule-interval intelligent-timer 50 50 50
  lsa-originate-interval intelligent-timer 500 50 100
  lsa-arrival-interval intelligent-timer 50 50 50
  area 0.0.0.0
  network 7.7.7.7 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.2.0 0.0.0.255
```



```
 network 192.168.3.0 0.0.0.255
 network 192.168.4.0 0.0.0.255
 network 192.168.9.0 0.0.0.255
 network 192.168.10.0 0.0.0.255
#
return
```

- Spine2的配置脚本

```
#
sysname Spine2
#
evpn-overlay enable
#
bfd
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.5.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.6.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.7.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 undo portswitch
 ip address 192.168.8.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 undo portswitch
 ip address 192.168.11.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
#
interface 100GE1/0/6
 undo portswitch
 ip address 192.168.12.1 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 780000
#
interface LoopBack0
 ip address 8.8.8.8 255.255.255.255
#
bgp 100
 router-id 8.8.8.8
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack0
```

```
 peer 2.2.2.2 as-number 100
 peer 2.2.2.2 connect-interface LoopBack0
 peer 3.3.3.3 as-number 100
 peer 3.3.3.3 connect-interface LoopBack0
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack0
 peer 9.9.9.9 as-number 100
 peer 9.9.9.9 connect-interface LoopBack0
 peer 10.10.10.10 as-number 100
 peer 10.10.10.10 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 1.1.1.1 enable
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
  peer 4.4.4.4 enable
  peer 9.9.9.9 enable
  peer 10.10.10.10 enable
 #
 l2vpn-family evpn
  undo policy vpn-target
  bestroute add-path path-number 64
  peer 1.1.1.1 enable
  peer 1.1.1.1 advertise irb
  peer 1.1.1.1 advertise irbv6
  peer 1.1.1.1 reflect-client
  peer 1.1.1.1 capability-advertise add-path both
  peer 1.1.1.1 advertise add-path path-number 64
  peer 2.2.2.2 enable
  peer 2.2.2.2 advertise irb
  peer 2.2.2.2 advertise irbv6
  peer 2.2.2.2 reflect-client
  peer 2.2.2.2 capability-advertise add-path both
  peer 2.2.2.2 advertise add-path path-number 64
  peer 3.3.3.3 enable
  peer 3.3.3.3 advertise irb
  peer 3.3.3.3 advertise irbv6
  peer 3.3.3.3 reflect-client
  peer 3.3.3.3 capability-advertise add-path both
  peer 3.3.3.3 advertise add-path path-number 64
  peer 4.4.4.4 enable
  peer 4.4.4.4 advertise irb
  peer 4.4.4.4 advertise irbv6
  peer 4.4.4.4 reflect-client
  peer 4.4.4.4 capability-advertise add-path both
  peer 4.4.4.4 advertise add-path path-number 64
  peer 9.9.9.9 enable
  peer 9.9.9.9 advertise irb
  peer 9.9.9.9 advertise irbv6
  peer 9.9.9.9 capability-advertise add-path both
  peer 9.9.9.9 advertise add-path path-number 64
  peer 10.10.10.10 enable
  peer 10.10.10.10 advertise irb
  peer 10.10.10.10 advertise irbv6
  peer 10.10.10.10 capability-advertise add-path both
  peer 10.10.10.10 advertise add-path path-number 64
 #
 ospf 1 router-id 8.8.8.8
  bfd all-interfaces enable
  bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
  spf-schedule-interval intelligent-timer 50 50 50
  lsa-originate-interval intelligent-timer 500 50 100
  lsa-arrival-interval intelligent-timer 50 50 50
  area 0.0.0.0
   network 8.8.8.8 0.0.0.0
   network 192.168.5.0 0.0.0.255
   network 192.168.6.0 0.0.0.255
   network 192.168.7.0 0.0.0.255
   network 192.168.8.0 0.0.0.255
```

```
 network 192.168.11.0 0.0.0.255
 network 192.168.12.0 0.0.0.255
#
return
```

- Leaf1的配置脚本

```
#
sysname Leaf1
#
dfs-group 1
 priority 150
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 1.1.1.1 peer 2.2.2.2
 m-lag up-delay 240 auto-recovery interval 10
 dual-active detection delay 0
#
load-balance ecmp
 vxlan-overlay network local-preference enable
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 1.1.1.1:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 1.1.1.1:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 200
#
bfd
 bfd forwarding match remote-discriminator 20
 bfd forwarding match remote-discriminator 40
 bfd forwarding match remote-discriminator 60
 bfd forwarding match remote-discriminator 80
#
bridge-domain 10
 vxlan vni 110
 evpn
  route-distinguisher 1.1.1.1:110
  vpn-target 0:110 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:110 import-extcommunity
#
bridge-domain 20
 vxlan vni 120
 evpn
  route-distinguisher 1.1.1.1:120
  vpn-target 0:120 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:120 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
 ipv6 address FC00:1::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0101
 ipv6 nd collect host enable
 ipv6 nd na glean
```

```
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.2.1.1 255.255.255.0
 ipv6 address FC00:2::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0102
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.10.10.1 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 1
 arp anti-attack rate-limit 200
#
interface Eth-Trunk10.10 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface Eth-Trunk11
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 2
 arp anti-attack rate-limit 200
#
interface Eth-Trunk11.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.5.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
```

```
interface 100GE1/0/4
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 eth-trunk 10
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/6
 eth-trunk 11
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface LoopBack0
 ip address 1.1.1.1 255.255.255.255
#
interface LoopBack1
 ip address 5.5.5.5 255.255.255.255
#
interface LoopBack2
 ip address 12.12.12.12 255.255.255.255
#
interface Nve1
 source 5.5.5.5
 pip-source 12.12.12.12 peer 13.13.13.13 bypass
 vni 110 head-end peer-list protocol bgp
 vni 120 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0111
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk11 downlink 2
 timer recover-time 60
#
bfd toipu1_v4 bind peer-ip 10.1.1.2 vpn-instance vpn1 interface Vbdif10 source-ip 10.1.1.1 one-arm-
echo
 discriminator local 10
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu1_v6 bind peer-ipv6 FC00:1::2 vpn-instance vpn1 interface Vbdif10 source-ipv6 FC00:1::1 one-
arm-echo
 discriminator local 50
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu2_v4 bind peer-ip 10.2.1.2 vpn-instance vpn1 interface Vbdif20 source-ip 10.2.1.1 one-arm-
echo
 discriminator local 30
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu2_v6 bind peer-ipv6 FC00:2::2 vpn-instance vpn1 interface Vbdif20 source-ipv6 FC00:2::1 one-
arm-echo
 discriminator local 70
 detect-multiplier 6
 min-echo-rx-interval 300
#
bgp 100
```

```
 router-id 1.1.1.1
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing mixed 64
  advertise l2vpn evpn import-route-multipath
 #
 ipv6-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing mixed 64
  advertise l2vpn evpn import-route-multipath
 #
 l2vpn-family evpn
  policy vpn-target
  bestroute add-path path-number 64
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy export_policy export
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 capability-advertise add-path both
  peer 7.7.7.7 advertise add-path path-number 64
  peer 8.8.8.8 enable
  peer 8.8.8.8 route-policy export_policy export
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 capability-advertise add-path both
  peer 8.8.8.8 advertise add-path path-number 64
#
ospf 1 router-id 1.1.1.1
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
 area 0.0.0.0
  network 1.1.1.1 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.5.0 0.0.0.255
#
ip community-filter basic apply_gwip_route index 10 permit 82345
ip community-filter basic suppress_route index 10 permit 82346
#
route-policy export_policy deny node 10
 if-match community-filter suppress_route
#
route-policy export_policy permit node 20
 if-match community-filter apply_gwip_route
 apply gateway-ip origin-nexthop
 apply ipv6 gateway-ip origin-nexthop
#
route-policy export_policy permit node 999
#
route-policy import_static_policy permit node 100
 if-match tag 82345
 apply community 82345 additive
#
route-policy import_static_policy permit node 999
#
ip route-static 13.13.13.13 32 10.10.10.2 preference 1
ip route-static vpn-instance vpn1 172.16.1.1 255.255.255.255 10.1.1.2 preference 255 tag 82345 track
```



```
bfd-session toipu1_v4 inter-protocol-ecmp
ip route-static vpn-instance vpn1 172.16.1.1 255.255.255.255 10.2.1.2 preference 255 tag 82345 track
bfd-session toipu2_v4 inter-protocol-ecmp
#
ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 FC00:1::2 preference 255 tag 82345 track bfd-
session toipu1_v6 inter-protocol-ecmp
ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 FC00:2::2 preference 255 tag 82345 track bfd-
session toipu2_v6 inter-protocol-ecmp
#
return
```

- Leaf2的配置脚本

```
#
sysname Leaf2
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 2.2.2.2 peer 1.1.1.1
 m-lag up-delay 240 auto-recovery interval 10
 dual-active detection delay 0
#
load-balance ecmp
 vxlan-overlay network local-preference enable
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 2.2.2.2:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 2.2.2.2:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 200
#
bfd
 bfd forwarding match remote-discriminator 10
 bfd forwarding match remote-discriminator 30
 bfd forwarding match remote-discriminator 50
 bfd forwarding match remote-discriminator 70
#
bridge-domain 10
 vxlan vni 110
 evpn
  route-distinguisher 2.2.2.2:110
  vpn-target 0:110 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:110 import-extcommunity
#
bridge-domain 20
 vxlan vni 120
 evpn
  route-distinguisher 2.2.2.2:120
  vpn-target 0:120 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:120 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.1.1.1 255.255.255.0
```



```
 ipv6 address FC00:1::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0101
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.2.1.1 255.255.255.0
 ipv6 address FC00:2::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0102
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.10.10.2 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 1
 arp anti-attack rate-limit 200
#
interface Eth-Trunk10.10 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface Eth-Trunk11
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 2
 arp anti-attack rate-limit 200
#
interface Eth-Trunk11.20 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.2.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.6.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
```

```
interface 100GE1/0/3
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 eth-trunk 10
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/6
 eth-trunk 11
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface LoopBack0
 ip address 2.2.2.2 255.255.255.255
#
interface LoopBack1
 ip address 5.5.5.5 255.255.255.255
#
interface LoopBack2
 ip address 13.13.13.13 255.255.255.255
#
interface Nve1
 source 5.5.5.5
 pip-source 13.13.13.13 peer 12.12.12.12 bypass
 vni 110 head-end peer-list protocol bgp
 vni 120 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0111
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk11 downlink 2
 timer recover-time 60
#
bfd toipu1_v4 bind peer-ip 10.1.1.2 vpn-instance vpn1 interface Vbdif10 source-ip 10.1.1.1 one-arm-
echo
 discriminator local 20
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu1_v6 bind peer-ipv6 FC00:1::2 vpn-instance vpn1 interface Vbdif10 source-ipv6 FC00:1::1 one-
arm-echo
 discriminator local 60
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu2_v4 bind peer-ip 10.2.1.2 vpn-instance vpn1 interface Vbdif20 source-ip 10.2.1.1 one-arm-
echo
 discriminator local 40
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu2_v6 bind peer-ipv6 FC00:2::2 vpn-instance vpn1 interface Vbdif20 source-ipv6 FC00:2::1 one-
arm-echo
```

```
 discriminator local 80
 detect-multiplier 6
 min-echo-rx-interval 300
#
bgp 100
 router-id 2.2.2.2
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing mixed 64
  advertise l2vpn evpn import-route-multipath
 #
 ipv6-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing mixed 64
  advertise l2vpn evpn import-route-multipath
 #
 l2vpn-family evpn
  policy vpn-target
  bestroute add-path path-number 64
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy export_policy export
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 capability-advertise add-path both
  peer 7.7.7.7 advertise add-path path-number 64
  peer 8.8.8.8 enable
  peer 8.8.8.8 route-policy export_policy export
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 capability-advertise add-path both
  peer 8.8.8.8 advertise add-path path-number 64
#
ospf 1 router-id 2.2.2.2
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
 area 0.0.0.0
  network 2.2.2.2 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.2.0 0.0.0.255
  network 192.168.6.0 0.0.0.255
#
ip community-filter basic apply_gwip_route index 10 permit 82345
ip community-filter basic suppress_route index 10 permit 82346
#
route-policy export_policy deny node 10
 if-match community-filter suppress_route
#
route-policy export_policy permit node 20
 if-match community-filter apply_gwip_route
 apply gateway-ip origin-nexthop
 apply ipv6 gateway-ip origin-nexthop
#
route-policy export_policy permit node 999
#
route-policy import_static_policy permit node 100
 if-match tag 82345
 apply community 82345 additive
```

```
#
route-policy import_static_policy permit node 999
#
ip route-static 12.12.12.12 32 10.10.10.1 preference 1
ip route-static vpn-instance vpn1 172.16.1.1 255.255.255.255 10.1.1.2 preference 255 tag 82345 track
bfd-session toipu1_v4 inter-protocol-ecmp
ip route-static vpn-instance vpn1 172.16.1.1 255.255.255.255 10.2.1.2 preference 255 tag 82345 track
bfd-session toipu2_v4 inter-protocol-ecmp
#
ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 FC00:1::2 preference 255 tag 82345 track bfd-
session toipu1_v6 inter-protocol-ecmp
ipv6 route-static vpn-instance vpn1 2001:db8:1::1 128 FC00:2::2 preference 255 tag 82345 track bfd-
session toipu2_v6 inter-protocol-ecmp
#
return
```

- Leaf3的配置脚本

```
#
sysname Leaf3
#
dfs-group 1
 priority 150
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 3.3.3.3 peer 4.4.4.4
 m-lag up-delay 240 auto-recovery interval 10
 dual-active detection delay 0
#
load-balance ecmp
 vxlan-overlay network local-preference enable
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 3.3.3.3:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 1.1.1.1:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 200
#
bfd
 bfd forwarding match remote-discriminator 20
 bfd forwarding match remote-discriminator 40
 bfd forwarding match remote-discriminator 60
 bfd forwarding match remote-discriminator 80
#
bridge-domain 30
 vxlan vni 130
 evpn
  route-distinguisher 3.3.3.3:130
  vpn-target 0:130 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:130 import-extcommunity
#
bridge-domain 40
 vxlan vni 140
 evpn
  route-distinguisher 3.3.3.3:140
  vpn-target 0:140 export-extcommunity
  vpn-target 0:1 export-extcommunity
```

```
 vpn-target 0:140 import-extcommunity
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.3.1.1 255.255.255.0
 ipv6 address FC00:3::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0103
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif40
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.4.1.1 255.255.255.0
 ipv6 address FC00:4::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0104
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.10.10.5 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 1
 arp anti-attack rate-limit 200
#
interface Eth-Trunk10.30 mode l2
 encapsulation dot1q vid 30
 bridge-domain 30
#
interface Eth-Trunk11
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 2
 arp anti-attack rate-limit 200
#
interface Eth-Trunk11.40 mode l2
 encapsulation dot1q vid 40
 bridge-domain 40
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.3.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.7.2 255.255.255.0
```

```
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 eth-trunk 10
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/6
 eth-trunk 11
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface LoopBack0
 ip address 3.3.3.3 255.255.255.255
#
interface LoopBack1
 ip address 6.6.6.6 255.255.255.255
#
interface LoopBack2
 ip address 14.14.14.14 255.255.255.255
#
interface Nve1
 source 6.6.6.6
 pip-source 14.14.14.14 peer 15.15.15.15 bypass
 vni 130 head-end peer-list protocol bgp
 vni 140 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0112
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk11 downlink 2
 timer recover-time 60
#
bfd toipu3_v4 bind peer-ip 10.3.1.2 vpn-instance vpn1 interface Vbdif30 source-ip 10.3.1.1 one-arm-
echo
 discriminator local 10
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu3_v6 bind peer-ipv6 FC00:3::2 vpn-instance vpn1 interface Vbdif30 source-ipv6 FC00:3::1 one-
arm-echo
 discriminator local 50
 detect-multiplier 6
 min-echo-rx-interval 200
#
bfd toipu4_v4 bind peer-ip 10.4.1.2 vpn-instance vpn1 interface Vbdif40 source-ip 10.4.1.1 one-arm-
echo
```

```
 discriminator local 30
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu4_v6 bind peer-ipv6 FC00:4::2 vpn-instance vpn1 interface Vbdif40 source-ipv6 FC00:4::1 one-arm-echo
 discriminator local 70
 detect-multiplier 6
 min-echo-rx-interval 300
#
bgp 100
 router-id 3.3.3.3
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing mixed 64
  advertise l2vpn evpn import-route-multipath
 #
 ipv6-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing mixed 64
  advertise l2vpn evpn import-route-multipath
 #
 l2vpn-family evpn
  policy vpn-target
  bestroute add-path path-number 64
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy export_policy export
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 capability-advertise add-path both
  peer 7.7.7.7 advertise add-path path-number 64
  peer 8.8.8.8 enable
  peer 8.8.8.8 route-policy export_policy export
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 capability-advertise add-path both
  peer 8.8.8.8 advertise add-path path-number 64
#
ospf 1 router-id 3.3.3.3
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
  area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.3.0 0.0.0.255
  network 192.168.7.0 0.0.0.255
#
ip community-filter basic apply_gwip_route index 10 permit 82345
ip community-filter basic suppress_route index 10 permit 82346
#
route-policy export_policy deny node 10
 if-match community-filter suppress_route
#
route-policy export_policy permit node 20
 if-match community-filter apply_gwip_route
 apply gateway-ip origin-nexthop
 apply ipv6 gateway-ip origin-nexthop
```



```
#
route-policy export_policy permit node 999
#
route-policy import_static_policy permit node 100
 if-match tag 82345
 apply community 82345 additive
#
route-policy import_static_policy permit node 999
#
ip route-static 15.15.15.15 32 10.10.10.6 preference 1
ip route-static vpn-instance vpn1 172.16.2.1 255.255.255.255 10.3.1.2 preference 255 tag 82345 track
bfd-session toipu3_v4 inter-protocol-ecmp
ip route-static vpn-instance vpn1 172.16.2.1 255.255.255.255 10.4.1.2 preference 255 tag 82345 track
bfd-session toipu4_v4 inter-protocol-ecmp
#
ipv6 route-static vpn-instance vpn1 2001:db8:2::1 128 FC00:3::2 preference 255 tag 82345 track bfd-
session toipu3_v6 inter-protocol-ecmp
ipv6 route-static vpn-instance vpn1 2001:db8:2::1 128 FC00:4::2 preference 255 tag 82345 track bfd-
session toipu4_v6 inter-protocol-ecmp
#
return
```

- Leaf4的配置脚本

```
#
sysname Leaf4
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
 dual-active detection source ip 4.4.4.4 peer 3.3.3.3
 m-lag up-delay 240 auto-recovery interval 10
 dual-active detection delay 0
#
load-balance ecmp
 vxlan-overlay network local-preference enable
#
vlan 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 4.4.4.4:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 ipv6-family
  route-distinguisher 1.1.1.1:1
  vpn-target 0:1 export-extcommunity evpn
  vpn-target 0:1 import-extcommunity evpn
 vxlan vni 200
#
bfd
 bfd forwarding match remote-discriminator 10
 bfd forwarding match remote-discriminator 30
 bfd forwarding match remote-discriminator 50
 bfd forwarding match remote-discriminator 70
#
bridge-domain 30
 vxlan vni 130
 evpn
  route-distinguisher 4.4.4.4:130
  vpn-target 0:130 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:130 import-extcommunity
#
bridge-domain 40
```

```
 vxlan vni 140
 evpn
  route-distinguisher 4.4.4.4:140
  vpn-target 0:140 export-extcommunity
  vpn-target 0:1 export-extcommunity
  vpn-target 0:140 import-extcommunity
#
interface Vbdif30
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.3.1.1 255.255.255.0
 ipv6 address FC00:3::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0103
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vbdif40
 ip binding vpn-instance vpn1
 ipv6 enable
 ip address 10.4.1.1 255.255.255.0
 ipv6 address FC00:4::1/64
 arp broadcast-detect enable
 mac-address 00e0-fc00-0104
 ipv6 nd collect host enable
 ipv6 nd na glean
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.10.10.6 255.255.255.252
 reserved for vxlan bypass
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 1
 arp anti-attack rate-limit 200
#
interface Eth-Trunk10.30 mode l2
 encapsulation dot1q vid 30
 bridge-domain 30
#
interface Eth-Trunk11
 stp edged-port enable
 mode lacp-static
 lacp timeout fast
 dfs-group 1 m-lag 2
 arp anti-attack rate-limit 200
#
interface Eth-Trunk11.40 mode l2
 encapsulation dot1q vid 40
 bridge-domain 40
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.4.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
```



```
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.8.2 255.255.255.0
 ospf network-type p2p
 ospf peer hold-max-cost timer 800000
 qos phb marking dscp enable
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/3
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/4
 eth-trunk 1
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/5
 eth-trunk 10
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface 100GE1/0/6
 eth-trunk 11
 storm suppression unknown-unicast 5
 storm suppression multicast packets 1000
 storm suppression broadcast packets 1000
 port crc-statistics trigger error-down
 trap-threshold crc-statistics 100 interval 10
#
interface LoopBack0
 ip address 4.4.4.4 255.255.255.255
#
interface LoopBack1
 ip address 6.6.6.6 255.255.255.255
#
interface LoopBack2
 ip address 15.15.15.15 255.255.255.255
#
interface Nve1
 source 6.6.6.6
 pip-source 15.15.15.15 peer 14.14.14.14 bypass
 vni 130 head-end peer-list protocol bgp
 vni 140 head-end peer-list protocol bgp
 mac-address 00e0-fc00-0112
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk11 downlink 2
 timer recover-time 60
#
bfd toipu3_v4 bind peer-ip 10.3.1.2 vpn-instance vpn1 interface Vbdif30 source-ip 10.3.1.1 one-arm-
echo
 discriminator local 20
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu3_v6 bind peer-ipv6 FC00:3::2 vpn-instance vpn1 interface Vbdif30 source-ipv6 FC00:3::1 one-
arm-echo
 discriminator local 60
```



```
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu4_v4 bind peer-ip 10.4.1.2 vpn-instance vpn1 interface Vbdif40 source-ip 10.4.1.1 one-arm-
echo
 discriminator local 40
 detect-multiplier 6
 min-echo-rx-interval 300
#
bfd toipu4_v6 bind peer-ipv6 FC00:4::2 vpn-instance vpn1 interface Vbdif40 source-ipv6 FC00:4::1 one-
arm-echo
 discriminator local 80
 detect-multiplier 6
 min-echo-rx-interval 300
#
bgp 100
 router-id 4.4.4.4
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack0
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
  peer 8.8.8.8 enable
 #
 ipv4-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing mixed 64
  advertise l2vpn evpn import-route-multipath
 #
 ipv6-family vpn-instance vpn1
  import-route static route-policy import_static_policy
  maximum load-balancing mixed 64
  advertise l2vpn evpn import-route-multipath
 #
 l2vpn-family evpn
  policy vpn-target
  bestroute add-path path-number 64
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy export_policy export
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 advertise irbv6
  peer 7.7.7.7 capability-advertise add-path both
  peer 7.7.7.7 advertise add-path path-number 64
  peer 8.8.8.8 enable
  peer 8.8.8.8 route-policy export_policy export
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 advertise irbv6
  peer 8.8.8.8 capability-advertise add-path both
  peer 8.8.8.8 advertise add-path path-number 64
#
ospf 1 router-id 4.4.4.4
 bfd all-interfaces enable
 bfd all-interfaces min-tx-interval 300 min-rx-interval 300 detect-multiplier 6
 spf-schedule-interval intelligent-timer 50 50 50
 lsa-originate-interval intelligent-timer 500 50 100
 lsa-arrival-interval intelligent-timer 50 50 50
  area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.4.0 0.0.0.255
  network 192.168.8.0 0.0.0.255
#
ip community-filter basic apply_gwip_route index 10 permit 82345
ip community-filter basic suppress_route index 10 permit 82346
#
route-policy export_policy deny node 10
 if-match community-filter suppress_route
```

```
#
route-policy export_policy permit node 20
 if-match community-filter apply_gwip_route
 apply gateway-ip origin-nexthop
 apply ipv6 gateway-ip origin-nexthop
#
route-policy export_policy permit node 999
#
route-policy import_static_policy permit node 100
 if-match tag 82345
 apply community 82345 additive
#
route-policy import_static_policy permit node 999
#
ip route-static 14.14.14.14 32 10.10.10.5 preference 1
ip route-static vpn-instance vpn1 172.16.2.1 255.255.255.255 10.3.1.2 preference 255 tag 82345 track bfd-session toipu3_v4 inter-protocol-ecmp
ip route-static vpn-instance vpn1 172.16.2.1 255.255.255.255 10.4.1.2 preference 255 tag 82345 track bfd-session toipu4_v4 inter-protocol-ecmp
#
ipv6 route-static vpn-instance vpn1 2001:db8:2::1 128 FC00:3::2 preference 255 tag 82345 track bfd-session toipu3_v6 inter-protocol-ecmp
ipv6 route-static vpn-instance vpn1 2001:db8:2::1 128 FC00:4::2 preference 255 tag 82345 track bfd-session toipu4_v6 inter-protocol-ecmp
#
return
```

1.8 配置 M-LAG Lite 示例

适用产品和版本

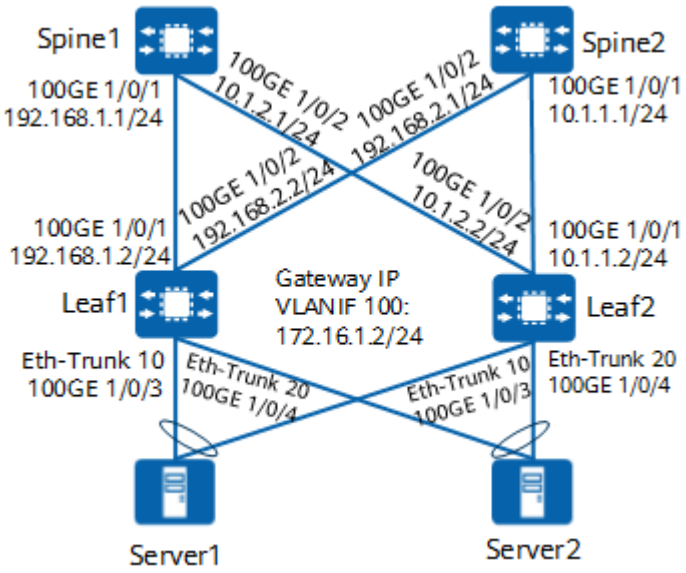
CloudEngine系列交换机V300R020C00或更高版本。

如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

如**图1-8**所示，在数据中心内部客户希望在接入层实现设备独立，做到控制面隔离以及设备故障隔离，且版本升级不会相互影响的前提下能够双活转发处理。跨设备链路聚合（M-LAG Lite）方式配置，能够确保多台设备间控制面的解耦，且设备间无须配置心跳链路来同步协议报文即可达到双活转发的目的。

图 1-8 配置 M-LAG Lite 组网图



配置思路

采用如下的思路配置：

1. 配置服务器使用跨设备链路聚合（M-LAG Lite）方式接入Leaf交换机。

   为了保证Eth-Trunk协商成功，需要保证Leaf1和Leaf2具有相同的LACP系统优先级、Eth-Trunk接口具有相同的接口ID和相同的LACP系统ID。同时，还要求两台Leaf上的Eth-Trunk成员口在LACP协议中的编号不同，以防止LACP协商不成功。

   说明

   如果通过支持STP的设备使用跨设备链路聚合（M-LAG Lite）方式接入Leaf交换机，需要在该设备上去使能STP，同时在leaf的下行口上配置**stp edged-port enable**，否则在该设备上会出现STP震荡。

2. 配置各个接口的IP地址。
3. 配置Leaf层交换机VLANIF接口相同的IP地址和MAC地址，作为服务器的双活网关。
   - Leaf层设备必须是三层网关角色，不能是二层透传设备，即组网部署上实现三层到边。因为如果三层网关设置在Spine设备，服务器以M-LAG Lite方式接入Leaf，那么在Spine网关设备上学习到的服务器的ARP表项就会产生两个出口，出现MAC漂移现象，所以服务器以M-LAG Lite方式接入的Leaf层设备必须是组网部署中的三层网关设备。
   - 为实现上述的三层到边，使得服务器到Leaf的流量全部进行三层转发（即使在同一二层域内的流量也进行三层转发），需要在Leaf层交换机上使能ARP任意代理。
   - Leaf1和Leaf2配置的网关IP地址相同，在Spine上会形成到Leaf的ECMP负载分担。当服务器到Leaf之间的链路发生故障，比如Server1到Leaf1之间的链路故障，双归变为单归时，下行访问Server1的流量可能会因为被ECMP Hash到Leaf1而发生断流。因此，要求Leaf的网关接口能够向路由邻居发布ARP Vlink直连路由，将ARP转换的主机路由发布给Spine，然后Leaf上再配置相应的动态路由协议引入直连路由。上述故障发生时，Leaf1上对应Server1的ARP表项被删除，ARP Vlink直连路由消失，Spine下行访问Server1的流量转发给Leaf2，保证流量转发正常。

- 为了保证Server1和Server2的正常互访，Spine学习到Leaf发布的ARP Vlink直连路由后，也需要将此路由发布给其他Leaf。非故障场景下，Leaf上会同时存在ARP转换的主机路由（此路由不会被下发到转发表项）以及Spine邻居发布的主机路由，请设置ARP转换的主机路由具有较高的优先级，以保证指导转发的是本地的ARP表项；当服务器到Leaf之间的链路发生故障，比如Server1到Leaf1之间的链路故障，Leaf1上对应Server1的ARP表项被删除，ARP Vlink直连路由消失，Server2访问Server1的流量路径变为：Server2 -> Leaf1 -> Spine1/Spine2 -> Leaf2 -> Server1。

4. 配置Leaf层与Spine层交换机之间使能路由协议，建立邻居关系。
5. 配置Leaf层交换机Monitor Link关联上行接口和下行接口，避免因上行链路故障导致用户侧流量无法转发而丢弃。

操作步骤

步骤1 配置跨设备LACP模式链路聚合。

配置Leaf1

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf1  //修改设备名称为Leaf1
[*HUAWEI] commit
[~Leaf1] vlan batch 100
[*Leaf1] interface eth-trunk 10
[*Leaf1-Eth-Trunk10] trunkport 100ge 1/0/3
[*Leaf1-Eth-Trunk10] mode lacp-static
[*Leaf1-Eth-Trunk10] lacp system-id 00e0-fc00-0000  //在Leaf1与Leaf2上配置相同的LACP系统ID
[*Leaf1-Eth-Trunk10] port link-type trunk
[*Leaf1-Eth-Trunk10] port trunk pvid vlan 100
[*Leaf1-Eth-Trunk10] port trunk allow-pass vlan 100
[*Leaf1-Eth-Trunk10] quit
[*Leaf1] interface eth-trunk 20
[*Leaf1-Eth-Trunk20] trunkport 100ge 1/0/4
[*Leaf1-Eth-Trunk20] mode lacp-static
[*Leaf1-Eth-Trunk20] lacp system-id 00e0-fc00-0001  //在Leaf1与Leaf2上配置相同的LACP系统ID
[*Leaf1-Eth-Trunk20] port link-type trunk
[*Leaf1-Eth-Trunk20] port trunk pvid vlan 100
[*Leaf1-Eth-Trunk20] port trunk allow-pass vlan 100
[*Leaf1-Eth-Trunk20] quit
[*Leaf1] lacp priority 100  //在Leaf1与Leaf2上配置相同的LACP系统优先级
[*Leaf1] commit
```

配置Leaf2

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf2  //修改设备名称为Leaf2
[*HUAWEI] commit
[~Leaf2] vlan batch 100
[*Leaf2] interface eth-trunk 10
[*Leaf2-Eth-Trunk10] trunkport 100ge 1/0/3
[*Leaf2-Eth-Trunk10] mode lacp-static
[*Leaf2-Eth-Trunk10] lacp system-id 00e0-fc00-0000  //在Leaf1与Leaf2上配置相同的LACP系统ID
[*Leaf2-Eth-Trunk10] lacp port-id-extension enable  //在leaf2上配置Eth-Trunk成员接口编号扩展，使成员接口编号均增加32768
[*Leaf2-Eth-Trunk10] port link-type trunk
[*Leaf2-Eth-Trunk10] port trunk pvid vlan 100
[*Leaf2-Eth-Trunk10] port trunk allow-pass vlan 100
[*Leaf2-Eth-Trunk10] quit
[*Leaf2] interface eth-trunk 20
[*Leaf2-Eth-Trunk20] trunkport 100ge 1/0/4
[*Leaf2-Eth-Trunk20] mode lacp-static
[*Leaf2-Eth-Trunk20] lacp system-id 00e0-fc00-0001  //在Leaf1与Leaf2上配置相同的LACP系统ID
[*Leaf2-Eth-Trunk20] lacp port-id-extension enable  //在leaf2上配置Eth-Trunk成员接口编号扩展，使成员接口编号均增加32768
[*Leaf2-Eth-Trunk20] port link-type trunk
```



```
[*Leaf2-Eth-Trunk20] port trunk pvid vlan 100
[*Leaf2-Eth-Trunk20] port trunk allow-pass vlan 100
[*Leaf2-Eth-Trunk20] quit
[*Leaf2] lacp priority 100  //在Leaf1与Leaf2上配置相同的LACP系统优先级
[*Leaf2] commit
```

步骤2 配置接口以及接口的IP地址。

配置Spine1

```
<HUAWEI> system-view
[~HUAWEI] sysname Spine1  //修改设备名称为Spine1
[*HUAWEI] commit
[~Spine1] interface 100ge 1/0/1
[~Spine1-100GE1/0/1] undo portswitch
[*Spine1-100GE1/0/1] port-isolate l3 enable  //使能端口三层隔离功能
[*Spine1-100GE1/0/1] ip address 192.168.1.1 24
[*Spine1-100GE1/0/1] quit
[*Spine1] interface 100ge 1/0/2
[*Spine1-100GE1/0/2] undo portswitch
[*Spine1-100GE1/0/2] port-isolate l3 enable  //使能端口三层隔离功能
[*Spine1-100GE1/0/2] ip address 10.1.2.1 24
[*Spine1-100GE1/0/2] quit
[*Spine1] commit
```

配置Spine2

```
<HUAWEI> system-view
[~HUAWEI] sysname Spine2  //修改设备名称为Spine2
[*HUAWEI] commit
[~Spine2] interface 100ge 1/0/1
[~Spine2-100GE1/0/1] undo portswitch
[*Spine2-100GE1/0/1] port-isolate l3 enable  //使能端口三层隔离功能
[*Spine2-100GE1/0/1] ip address 10.10.1.1 24
[*Spine2-100GE1/0/1] quit
[*Spine2] interface 100ge 1/0/2
[*Spine2-100GE1/0/2] undo portswitch
[*Spine2-100GE1/0/2] port-isolate l3 enable  //使能端口三层隔离功能
[*Spine2-100GE1/0/2] ip address 192.168.2.1 24
[*Spine2-100GE1/0/2] quit
[*Spine2] commit
```

配置Leaf1

```
[~Leaf1] interface 100ge 1/0/1
[~Leaf1-100GE1/0/1] undo portswitch
[*Leaf1-100GE1/0/1] port-isolate l3 enable  //使能端口三层隔离功能
[*Leaf1-100GE1/0/1] ip address 192.168.1.2 24
[*Leaf1-100GE1/0/1] quit
[*Leaf1] interface 100ge 1/0/2
[*Leaf1-100GE1/0/2] undo portswitch
[*Leaf1-100GE1/0/2] port-isolate l3 enable  //使能端口三层隔离功能
[*Leaf1-100GE1/0/2] ip address 192.168.2.2 24
[*Leaf1-100GE1/0/2] quit
[*Leaf1] commit
```

配置Leaf2

```
[~Leaf2] interface 100ge 1/0/1
[~Leaf2-100GE1/0/1] undo portswitch
[*Leaf2-100GE1/0/1] port-isolate l3 enable  //使能端口三层隔离功能
[*Leaf2-100GE1/0/1] ip address 10.1.1.2 24
[*Leaf2-100GE1/0/1] quit
[*Leaf2] interface 100ge 1/0/2
[*Leaf2-100GE1/0/2] undo portswitch
[*Leaf2-100GE1/0/2] port-isolate l3 enable  //使能端口三层隔离功能
[*Leaf2-100GE1/0/2] ip address 10.1.2.2 24
[*Leaf2-100GE1/0/2] quit
[*Leaf2] commit
```


步骤3 配置Leaf层交换机VLANIF接口相同的IP地址和MAC地址，作为接入服务器的双活网关。

配置Leaf1

```
[~Leaf1] interface vlanif 100
[*Leaf1-Vlanif100] ip address 172.16.1.2 24
[*Leaf1-Vlanif100] mac-address 0000-5e00-0101
[*Leaf1-Vlanif100] arp timeout 90  //配置动态ARP表项的老化超时时间为90秒
[*Leaf1-Vlanif100] arp proxy anyway enable  //使能ARP任意代答功能，服务器之间的ARP请求报文，在Leaf层上被捕获后，会使用自己的三层网关接口的接口MAC/IP进行ARP应答，这样服务器的上行流量到达Leaf层后，都会进行三层转发
[*Leaf1-Vlanif100] arp delete trigger link-down enable  //使能链路Down时ARP表项快速删除功能,当成员端口Down时，该端口下的ARP表项就会立即删除，不再等待ARP探测
[*Leaf1-Vlanif100] arp direct-route enable  //使能接口发布ARP Vlink直连路由功能，然后通过路由协议引入直连路由，可以将ARP转换的主机路由发布给路由邻居，避免在链路故障时形成路由黑洞
[*Leaf1-Vlanif100] arp direct-route preference 1  //配置ARP Vlink直连路由的优先级为1
[*Leaf1-Vlanif100] arp direct-route delay 120  //(可选)为了避免由于ARP Vlink直连路由建立慢，但是路由引流快导致的流量丢失问题，可以配置延迟发布ARP Vlink直连路由的时间
[*Leaf1-Vlanif100] quit
[*Leaf1] commit
```

配置Leaf2

```
[~Leaf2] interface vlanif 100
[*Leaf2-Vlanif100] ip address 172.16.1.2 24
[*Leaf2-Vlanif100] mac-address 0000-5e00-0101
[*Leaf2-Vlanif100] arp timeout 90  //配置动态ARP表项的老化超时时间为90秒
[*Leaf2-Vlanif100] arp proxy anyway enable  //使能ARP任意代答功能，服务器之间的ARP请求报文，在Leaf层上被捕获后，会使用自己的三层网关接口的接口MAC/IP进行ARP应答，这样服务器的上行流量到达Leaf层后，都会进行三层转发
[*Leaf2-Vlanif100] arp delete trigger link-down enable  //使能链路Down时ARP表项快速删除功能,当成员端口Down时，该端口下的ARP表项就会立即删除，不再等待ARP探测
[*Leaf2-Vlanif100] arp direct-route enable  //使能接口发布ARP Vlink直连路由功能，然后通过路由协议引入直连路由，可以将ARP转换的主机路由发布给路由邻居，避免在链路故障时形成路由黑洞
[*Leaf2-Vlanif100] arp direct-route preference 1  //配置ARP Vlink直连路由的优先级为1
[*Leaf2-Vlanif100] arp direct-route delay 120  //(可选)为了避免由于ARP Vlink直连路由建立慢，但是路由引流快导致的流量丢失问题，可以配置延迟发布ARP Vlink直连路由的时间
[*Leaf2-Vlanif100] quit
[*Leaf2] commit
```

说明

对于负载分担接入Leaf层交换机的服务器来说，由于HASH选路的原因，服务器发送的ARP应答或者ARP请求报文只会发送给Leaf1设备，而Leaf2设备无法收到服务器发送的ARP报文，导致Leaf2上总是无法学习到服务器的ARP表项，所以服务器双归负载分担接入场景下，依赖服务器可以定期在两个网卡的接口上都发送ARP请求报文。若服务器不具备该功能，可以在Leaf层交换机上执行**arp smart-discover enable**命令用来使能ARP主动探测功能，主动发现本地下挂的主机设备。

步骤4 汇聚层和接入层交换机间配置BGP路由协议，实现三层互通。

配置Spine1

```
[~Spine1] bgp 65009
[*Spine1-bgp] group leaf external
[*Spine1-bgp] peer 192.168.1.2 as-number 65020
[*Spine1-bgp] peer 192.168.1.2 group leaf
[*Spine1-bgp] peer 10.1.2.2 as-number 65021
[*Spine1-bgp] peer 10.1.2.2 group leaf
[*Spine1-bgp] load-balancing as-path-relax  //路由在形成负载分担时不比较相同长度的AS-Path属性，确保不同Leaf发给Spine的路由可以形成负载分担
[*Spine1-bgp] timer keepalive 10 hold 30
[*Spine1-bgp] preference 20 200 10
[*Spine1-bgp] quit
[*Spine1] commit
```

配置Spine2

```
[~Spine2] bgp 65009
[*Spine2-bgp] group leaf external
[*Spine2-bgp] peer 192.168.2.2 as-number 65020
[*Spine2-bgp] peer 192.168.2.2 group leaf
[*Spine2-bgp] peer 10.1.1.2 as-number 65021
[*Spine2-bgp] peer 10.1.1.2 group leaf
[*Spine2-bgp] load-balancing as-path-relax  //路由在形成负载分担时不比较相同长度的AS-Path属性，确保不同Leaf发给Spine的路由可以形成负载分担
[*Spine2-bgp] timer keepalive 10 hold 30
[*Spine2-bgp] preference 20 200 10
[*Spine2-bgp] quit
[*Spine2] commit
```

配置Leaf1

```
[~Leaf1] bgp 65020
[*Leaf1-bgp] group spine external
[*Leaf1-bgp] peer spine as-number 65009
[*Leaf1-bgp] peer 192.168.1.1 as-number 65009
[*Leaf1-bgp] peer 192.168.1.1 group spine
[*Leaf1-bgp] peer 192.168.2.1 as-number 65009
[*Leaf1-bgp] peer 192.168.2.1 group spine
[*Leaf1-bgp] timer keepalive 10 hold 30
[*Leaf1-bgp] preference 20 200 10
[*Leaf1-bgp] import-route direct  //引入直连路由学习到的路由信息，可以根据实际组网情况配置路由策略过滤掉非必要的路由
[*Leaf1-bgp] quit
[*Leaf1] commit
```

配置Leaf2

```
[~Leaf2] bgp 65021
[*Leaf2-bgp] group spine external
[*Leaf2-bgp] peer spine as-number 65009
[*Leaf2-bgp] peer 10.1.2.1 as-number 65009
[*Leaf2-bgp] peer 10.1.2.1 group spine
[*Leaf2-bgp] peer 10.1.1.1 as-number 65009
[*Leaf2-bgp] peer 10.1.1.1 group spine
[*Leaf2-bgp] timer keepalive 10 hold 30
[*Leaf2-bgp] preference 20 200 10
[*Leaf2-bgp] import-route direct  //引入直连路由学习到的路由信息，可以根据实际组网情况配置路由策略过滤掉非必要的路由
[*Leaf2-bgp] quit
[*Leaf2] commit
```

说明

本文以Leaf1和Leaf2配置不同AS作为举例说明。对于Leaf1和Leaf2配置相同AS场景，需要增加配置命令**peer allow-as-loop**，以确保不同Leaf的路由经过Spine后，能将路由发布给对方。

步骤5 配置Leaf层交换机Monitor Link组关联上下行接口

配置Leaf1

```
[~Leaf1] monitor-link group 1
[*Leaf1-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf1-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf1-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf1-mtlk-group1] port eth-trunk 20 downlink 2
[*Leaf1-mtlk-group1] quit
[*Leaf1] commit
```

配置Leaf2

```
[~Leaf2] monitor-link group 1
[*Leaf2-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf2-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf2-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf2-mtlk-group1] port eth-trunk 20 downlink 2
```



```
[*Leaf2-mtlk-group1] quit
[*Leaf2] commit
```

----结束

检查配置结果

- 查看设备的Eth-Trunk信息，查看M-LAG Lite已与服务器协商成功，端口状态为Up。

```
[~Leaf1] display eth-trunk 10
Eth-Trunk10's state information is:
Local:
LAG ID: 10                    Working Mode: Static
Preempt Delay: Disabled         Hash Arithmetic: profile default
System Priority: 100            System ID: 00e0-fc00-0000
Least Active-linknumber: 1      Max Active-linknumber: 32
Operating Status: up            Number Of Up Ports In Trunk: 1
Timeout Period: Slow
PortKeyMode: Auto
--------------------------------------------------------------------------------
ActorPortName         Status   PortType PortPri PortNo PortKey PortState Weight
100GE1/0/3            Selected 100GE    32768   1      2625    10111100  1

Partner:
--------------------------------------------------------------------------------
ActorPortName         SysPri   SystemID        PortPri PortNo PortKey PortState
100GE1/0/3            32768    00e0-fc12-2401  32768   1      2625    10111100
[~Leaf1] display eth-trunk 20
Eth-Trunk20's state information is:
Local:
LAG ID: 20                    Working Mode: Static
Preempt Delay: Disabled         Hash Arithmetic: profile default
System Priority: 100            System ID: 00e0-fc00-0001
Least Active-linknumber: 1      Max Active-linknumber: 32
Operating Status: up            Number Of Up Ports In Trunk: 1
Timeout Period: Slow
PortKeyMode: Auto
--------------------------------------------------------------------------------
ActorPortName         Status   PortType PortPri PortNo PortKey PortState Weight
100GE1/0/4            Selected 100GE    32768   2      5185    10111100  1

Partner:
--------------------------------------------------------------------------------
ActorPortName         SysPri   SystemID        PortPri PortNo PortKey PortState
100GE1/0/4            32768    00e0-fc39-8b01  32768   1      5185    10111100
```

- 查看设备路由关系建立情况。

```
[~Spine1] display bgp routing-table
 BGP Local router ID is 10.1.1.190
 Status codes: * - valid, > - best, d - damped, h - history,
               i - internal, s - suppressed, S - Stale
 Origin      : i - IGP, e - EGP, ? - incomplete


 Total Number of Routes: 6
      Network            NextHop                MED        LocPrf    PrefVal Path/Ogn
      10.1.2.0/24        10.1.2.2                 0                    0      65021?
*>    172.16.1.0/24      10.1.2.2                 0                    0      65021?
*                        192.168.1.2              0                    0      65020?
*>    10.1.1.0/24        10.1.2.2                 0                    0      65021?
*                        192.168.1.2              0                    0      65020?
      192.168.1.0/24     192.168.1.2              0                    0      65020?
[~Spine1] display ip routing-table
Proto: Protocol        Pre: Preference
Route Flags: R - relay, D - download to fib, T - to vpn-instance, B - black hole route
------------------------------------------------------------------------------
Routing Table : _public_
         Destinations : 15        Routes : 15
```

```
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        0.0.0.0/0   Static  60   0           RD  10.1.1.1        MEth0/0/0
       10.1.2.0/24  Direct  0    0           D   10.1.2.1        100GE1/0/2
       10.1.2.1/32  Direct  0    0           D   127.0.0.1       100GE1/0/2
     10.1.2.255/32  Direct  0    0           D   127.0.0.1       100GE1/0/2
      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
127.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
     172.16.1.0/24  EBGP    20   0           RD  10.1.2.2        100GE1/0/2
       10.1.1.0/24  Direct  0    0           D   10.1.1.190      MEth0/0/0
     10.1.1.190/32  Direct  0    0           D   127.0.0.1       MEth0/0/0
     10.1.1.255/32  Direct  0    0           D   127.0.0.1       MEth0/0/0
    192.168.1.0/24  Direct  0    0           D   192.168.1.1     100GE1/0/1
    192.168.1.1/32  Direct  0    0           D   127.0.0.1       100GE1/0/1
  192.168.1.255/32  Direct  0    0           D   127.0.0.1       100GE1/0/1
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0?
```

配置脚本

- Spine1的配置脚本

```
#
sysname Spine1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
 port-isolate l3 enable
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.2.1 255.255.255.0
 port-isolate l3 enable
#
bgp 65009
 timer keepalive 10 hold 30
 group leaf external
 peer 10.1.2.2 as-number 65021
 peer 10.1.2.2 group leaf
 peer 192.168.1.2 as-number 65020
 peer 192.168.1.2 group leaf
 load-balancing as-path-relax
 #
 ipv4-family unicast
  preference 20 200 10
  peer leaf enable
  peer 10.1.2.2 enable
  peer 10.1.2.2 group leaf
  peer 192.168.1.2 enable
  peer 192.168.1.2 group leaf
#
return
```

- Spine2的配置脚本

```
#
sysname Spine2
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.1 255.255.255.0
 port-isolate l3 enable
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.2.1 255.255.255.0
 port-isolate l3 enable
#
bgp 65009
```

```
 timer keepalive 10 hold 30
 group leaf external
 peer 10.1.1.2 as-number 65021
 peer 10.1.1.2 group leaf
 peer 192.168.2.2 as-number 65020
 peer 192.168.2.2 group leaf
 load-balancing as-path-relax
 #
 ipv4-family unicast
  preference 20 200 10
  peer leaf enable
  peer 10.1.1.2 enable
  peer 10.1.1.2 group leaf
  peer 192.168.2.2 enable
  peer 192.168.2.2 group leaf
#
return
```

- Leaf1的配置脚本

```
#
sysname Leaf1
#
vlan batch 100
#
lacp priority 100
#
interface Vlanif100
 ip address 172.16.1.2 255.255.255.0
 arp timeout 90
 arp proxy anyway enable
 mac-address 0000-5e00-0101
 arp delete trigger link-down enable
 arp direct-route enable
 arp direct-route preference 1
 arp direct-route delay 120
#
interface Eth-Trunk10
 port link-type trunk
 port trunk pvid vlan 100
 port trunk allow-pass vlan 100
 mode lacp-static
 lacp system-id 00e0-fc00-0000
#
interface Eth-Trunk20
 port link-type trunk
 port trunk pvid vlan 100
 port trunk allow-pass vlan 100
 mode lacp-static
 lacp system-id 00e0-fc00-0001
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
 port-isolate l3 enable
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.2.2 255.255.255.0
 port-isolate l3 enable
#
interface 100GE1/0/3
 eth-trunk 10
#
interface 100GE1/0/4
 eth-trunk 20
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
```

```
 port Eth-Trunk20 downlink 2
#
bgp 65020
 timer keepalive 10 hold 30
 group spine external
 peer spine as-number 65009
 peer 192.168.1.1 as-number 65009
 peer 192.168.1.1 group spine
 peer 192.168.2.1 as-number 65009
 peer 192.168.2.1 group spine
 #
 ipv4-family unicast
  preference 20 200 10
  import-route direct
  peer spine enable
  peer 192.168.1.1 enable
  peer 192.168.1.1 group spine
  peer 192.168.2.1 enable
  peer 192.168.2.1 group spine
#
return
```

- Leaf2的配置脚本

```
#
sysname Leaf2
#
vlan batch 100
#
lacp priority 100
#
interface Vlanif100
 ip address 172.16.1.2 255.255.255.0
 arp timeout 90
 arp proxy anyway enable
 mac-address 0000-5e00-0101
 arp delete trigger link-down enable
 arp direct-route enable
 arp direct-route preference 1
 arp direct-route delay 120
#
interface Eth-Trunk10
 port link-type trunk
 port trunk pvid vlan 100
 port trunk allow-pass vlan 100
 mode lacp-static
 lacp system-id 00e0-fc00-0000
 lacp port-id-extension enable
#
interface Eth-Trunk20
 port link-type trunk
 port trunk pvid vlan 100
 port trunk allow-pass vlan 100
 mode lacp-static
 lacp system-id 00e0-fc00-0001
 lacp port-id-extension enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.2 255.255.255.0
 port-isolate l3 enable
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.2.2 255.255.255.0
 port-isolate l3 enable
#
interface 100GE1/0/3
 eth-trunk 10
 #
interface 100GE1/0/4
```



```
 eth-trunk 20
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk20 downlink 2
#
bgp 65021
 timer keepalive 10 hold 30
 group spine external
 peer spine as-number 65009
 peer 10.1.1.1 as-number 65009
 peer 10.1.1.1 group spine
 peer 10.1.2.1 as-number 65009
 peer 10.1.2.1 group spine
 #
 ipv4-family unicast
  preference 20 200 10
  import-route direct
  peer spine enable
  peer 10.1.1.1 enable
  peer 10.1.1.1 group spine
  peer 10.1.2.1 enable
  peer 10.1.2.1 group spine
#
return
```

1.9 配置 IPv6 M-LAG Lite 示例

适用产品和版本

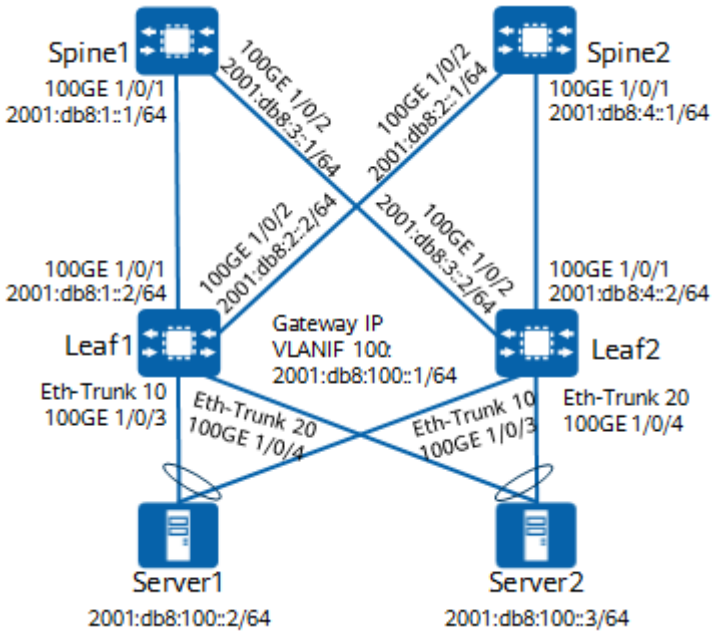
CloudEngine系列交换机V300R020C00或更高版本。

如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

如**图1-9**所示，在IPv6网络中，数据中心内部客户希望在接入层实现设备独立，做到控制面隔离以及设备故障隔离，且版本升级不会相互影响的前提下能够双活转发处理。通过配置M-LAG Lite，能够确保多台设备间控制面的解耦，且设备间无须配置心跳链路来同步协议报文即可达到双活转发的目的。

图 1-9 配置 IPv6 M-LAG Lite 组网图



配置思路

说明

本示例仅包含IPv6 M-LAG Lite的配置，如需配置IPv4&IPv6双协议栈下的M-LAG Lite，请将本示例与**1.8 配置M-LAG Lite示例**结合配置。

采用如下的思路配置：

1. 配置服务器使用M-LAG Lite方式接入Leaf交换机。

   为了保证Eth-Trunk协商成功，需要保证Leaf1和Leaf2具有相同的LACP系统优先级、Eth-Trunk接口具有相同的接口ID和相同的LACP系统ID。同时，还要求两台Leaf上的Eth-Trunk成员口在LACP协议中的编号不同，以防止LACP协商不成功。

   说明

   如果通过支持STP的设备使用跨设备链路聚合（M-LAG Lite）方式接入Leaf交换机，需要在该设备上去使能STP，同时在leaf的下行口上配置**stp edged-port enable**，否则在该设备上会出现STP震荡。

2. 配置各个接口的IPv6地址。
3. 配置Leaf层交换机VLANIF接口相同的IPv6地址和MAC地址，作为服务器的双活网关。
   - Leaf层设备必须是三层网关角色，不能是二层透传设备，即组网部署上实现三层到边。因为如果三层网关设置在Spine设备，服务器以M-LAG Lite方式接入Leaf，那么在Spine网关设备上学习到的服务器的ND表项就会产生两个出口，出现MAC漂移现象，所以服务器以M-LAG Lite方式接入的Leaf层设备必须是组网部署中的三层网关设备。
   - 为实现上述的三层到边，使得服务器到Leaf的流量全部进行三层转发（即使在同一二层域内的流量也进行三层转发），需要在Leaf层交换机上使能ND任意代理。
   - Leaf1和Leaf2配置的网关IPv6地址相同，在Spine上会形成到Leaf的ECMP负载分担。当服务器到Leaf之间的链路发生故障，比如Server1到Leaf1之间的

链路故障，双归变为单归时，下行访问Server1的流量可能会因为被ECMP Hash到Leaf1而发生断流。因此，要求Leaf的网关接口能够向路由邻居发布NDP Vlink直连路由，将ND转换的主机路由发布给Spine，然后Leaf上再配置相应的动态路由协议引入直连路由。上述故障发生时，Leaf1上对应Server1的ND表项被删除，NDP Vlink直连路由消失，Spine下行访问Server1的流量转发给Leaf2，保证流量转发正常。

- 为了保证Server1和Server2的正常互访，Spine学习到Leaf发布的NDP Vlink直连路由后，也需要将此路由发布给其他Leaf。非故障场景下，Leaf上会同时存在ND转换的主机路由（此路由不会被下发到转发表项）以及Spine邻居发布的主机路由，请设置ND转换的主机路由具有较高的优先级，以保证指导转发的是本地的ND表项；当服务器到Leaf之间的链路发生故障，比如Server1到Leaf1之间的链路故障，Leaf1上对应Server1的ND表项被删除，NDP Vlink直连路由消失，Server2访问Server1的流量路径变为：Server2 -> Leaf1 -> Spine1/Spine2 -> Leaf2 -> Server1。

4. 配置Leaf层与Spine层交换机之间使能路由协议，建立邻居关系。
5. 配置Leaf层交换机Monitor Link关联上行接口和下行接口，避免因上行链路故障导致用户侧流量无法转发而被丢弃。

操作步骤

步骤1 配置跨设备LACP模式链路聚合。

配置Leaf1

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf1  //修改设备名称为Leaf1
[*HUAWEI] commit
[~Leaf1] vlan batch 100
[*Leaf1] interface eth-trunk 10
[*Leaf1-Eth-Trunk10] trunkport 100ge 1/0/3
[*Leaf1-Eth-Trunk10] mode lacp-static
[*Leaf1-Eth-Trunk10] lacp system-id 00e0-fc00-0000  //在Leaf1与Leaf2上配置相同的LACP系统ID
[*Leaf1-Eth-Trunk10] port link-type trunk
[*Leaf1-Eth-Trunk10] port trunk pvid vlan 100
[*Leaf1-Eth-Trunk10] port trunk allow-pass vlan 100
[*Leaf1-Eth-Trunk10] quit
[*Leaf1] interface eth-trunk 20
[*Leaf1-Eth-Trunk20] trunkport 100ge 1/0/4
[*Leaf1-Eth-Trunk20] mode lacp-static
[*Leaf1-Eth-Trunk20] lacp system-id 00e0-fc00-0001  //在Leaf1与Leaf2上配置相同的LACP系统ID
[*Leaf1-Eth-Trunk20] port link-type trunk
[*Leaf1-Eth-Trunk20] port trunk pvid vlan 100
[*Leaf1-Eth-Trunk20] port trunk allow-pass vlan 100
[*Leaf1-Eth-Trunk20] quit
[*Leaf1] lacp priority 100  //在Leaf1与Leaf2上配置相同的LACP系统优先级
[*Leaf1] commit
```

配置Leaf2

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf2  //修改设备名称为Leaf2
[*HUAWEI] commit
[~Leaf2] vlan batch 100
[*Leaf2] interface eth-trunk 10
[*Leaf2-Eth-Trunk10] trunkport 100ge 1/0/3
[*Leaf2-Eth-Trunk10] mode lacp-static
[*Leaf2-Eth-Trunk10] lacp system-id 00e0-fc00-0000  //在Leaf1与Leaf2上配置相同的LACP系统ID
[*Leaf2-Eth-Trunk10] lacp port-id-extension enable  //在leaf2上配置Eth-Trunk成员接口编号扩展，使成员接口编号增加32768，以防止出现两台Leaf上的Eth-Trunk成员口在LACP协议中编号相同的情况
[*Leaf2-Eth-Trunk10] port link-type trunk
[*Leaf2-Eth-Trunk10] port trunk pvid vlan 100
[*Leaf2-Eth-Trunk10] port trunk allow-pass vlan 100
```

```
[*Leaf2-Eth-Trunk10] quit
[*Leaf2] interface eth-trunk 20
[*Leaf2-Eth-Trunk20] trunkport 100ge 1/0/4
[*Leaf2-Eth-Trunk20] mode lacp-static
[*Leaf2-Eth-Trunk20] lacp system-id 00e0-fc00-0001  //在Leaf1与Leaf2上配置相同的LACP系统ID
[*Leaf2-Eth-Trunk20] lacp port-id-extension enable  //在leaf2上配置Eth-Trunk成员接口编号扩展，使成员接口编号增加32768，以防止出现两台Leaf上的Eth-Trunk成员口在LACP协议中编号相同的情况
[*Leaf2-Eth-Trunk20] port link-type trunk
[*Leaf2-Eth-Trunk20] port trunk pvid vlan 100
[*Leaf2-Eth-Trunk20] port trunk allow-pass vlan 100
[*Leaf2-Eth-Trunk20] quit
[*Leaf2] lacp priority 100  //在Leaf1与Leaf2上配置相同的LACP系统优先级
[*Leaf2] commit
```

步骤2 配置接口以及接口的IPv6地址。

配置Spine1

```
<HUAWEI> system-view
[~HUAWEI] sysname Spine1  //修改设备名称为Spine1
[*HUAWEI] commit
[~Spine1] interface 100ge 1/0/1
[~Spine1-100GE1/0/1] undo portswitch
[*Spine1-100GE1/0/1] port-isolate l3 enable
[*Spine1-100GE1/0/1] ipv6 enable
[*Spine1-100GE1/0/1] ipv6 address 2001:db8:1::1 64
[*Spine1-100GE1/0/1] quit
[*Spine1] interface 100ge 1/0/2
[*Spine1-100GE1/0/2] undo portswitch
[*Spine1-100GE1/0/2] port-isolate l3 enable
[*Spine1-100GE1/0/2] ipv6 enable
[*Spine1-100GE1/0/2] ipv6 address 2001:db8:3::1 64
[*Spine1-100GE1/0/2] quit
[*Spine1] commit
```

配置Spine2

```
<HUAWEI> system-view
[~HUAWEI] sysname Spine2  //修改设备名称为Spine2
[*HUAWEI] commit
[~Spine2] interface 100ge 1/0/1
[~Spine2-100GE1/0/1] undo portswitch
[*Spine2-100GE1/0/1] port-isolate l3 enable
[*Spine2-100GE1/0/1] ipv6 enable
[*Spine2-100GE1/0/1] ipv6 address 2001:db8:4::1 64
[*Spine2-100GE1/0/1] quit
[*Spine2] interface 100ge 1/0/2
[*Spine2-100GE1/0/2] undo portswitch
[*Spine2-100GE1/0/2] port-isolate l3 enable
[*Spine2-100GE1/0/2] ipv6 enable
[*Spine2-100GE1/0/2] ipv6 address 2001:db8:2::1 64
[*Spine2-100GE1/0/2] quit
[*Spine2] commit
```

配置Leaf1

```
[~Leaf1] interface 100ge 1/0/1
[~Leaf1-100GE1/0/1] undo portswitch
[*Leaf1-100GE1/0/1] port-isolate l3 enable
[*Leaf1-100GE1/0/1] ipv6 enable
[*Leaf1-100GE1/0/1] ipv6 address 2001:db8:1::2 64
[*Leaf1-100GE1/0/1] quit
[*Leaf1] interface 100ge 1/0/2
[*Leaf1-100GE1/0/2] undo portswitch
[*Leaf1-100GE1/0/2] port-isolate l3 enable
[*Leaf1-100GE1/0/2] ipv6 enable
[*Leaf1-100GE1/0/2] ipv6 address 2001:db8:2::2 64
[*Leaf1-100GE1/0/2] quit
[*Leaf1] commit
```


配置Leaf2

```
[~Leaf2] interface 100ge 1/0/1
[~Leaf2-100GE1/0/1] undo portswitch
[*Leaf2-100GE1/0/1] port-isolate l3 enable
[*Leaf2-100GE1/0/1] ipv6 enable
[*Leaf2-100GE1/0/1] ipv6 address 2001:db8:4::2 64
[*Leaf2-100GE1/0/1] quit
[*Leaf2] interface 100ge 1/0/2
[*Leaf2-100GE1/0/2] undo portswitch
[*Leaf2-100GE1/0/2] port-isolate l3 enable
[*Leaf2-100GE1/0/2] ipv6 enable
[*Leaf2-100GE1/0/2] ipv6 address 2001:db8:3::2 64
[*Leaf2-100GE1/0/2] quit
[*Leaf2] commit
```

步骤3 配置Leaf层交换机VLANIF接口相同的IPv6地址和MAC地址，作为接入服务器的双活网关。

配置Leaf1

```
[~Leaf1] interface vlanif 100
[*Leaf1-Vlanif100] ipv6 enable
[*Leaf1-Vlanif100] ipv6 address 2001:db8:100::1 64
[*Leaf1-Vlanif100] ipv6 address FE80::300:3EFF:FE11:985 link-local
[*Leaf1-Vlanif100] mac-address 0000-5e00-0101
[*Leaf1-Vlanif100] ipv6 nd proxy anyway enable  //使能ND任意代答功能。Leaf捕获服务器发送的NS请求报文后，会使用自己的三层网关接口的接口MAC进行NA应答，这样服务器的上行流量到达Leaf层后，都会进行三层转发
[*Leaf1-Vlanif100] ipv6 nd delete trigger link-down enable  //使能链路Down时ND表项快速删除功能，当成员端口Down时，该端口下的ND表项就会立即删除
[*Leaf1-Vlanif100] ipv6 nd direct-route enable  //使能接口发布NDP Vlink直连路由功能，然后通过路由协议引入直连路由，可以将ND转换的主机路由发布给路由邻居，避免在链路故障时形成路由黑洞
[*Leaf1-Vlanif100] ipv6 nd direct-route preference 1  //配置NDP Vlink直连路由的优先级为1
[*Leaf1-Vlanif100] ipv6 nd direct-route delay 120  //(可选)为了避免由于IPv6 NDP Vlink直连路由建立慢，但是路由引流快导致的流量丢失问题，可以配置延迟发布IPv6 NDP Vlink直连路由的时间
[*Leaf1-Vlanif100] quit
[*Leaf1] commit
```

配置Leaf2

```
[~Leaf2] interface vlanif 100
[*Leaf2-Vlanif100] ipv6 enable
[*Leaf2-Vlanif100] ipv6 address 2001:db8:100::1 64
[*Leaf2-Vlanif100] ipv6 address FE80::300:3EFF:FE11:985 link-local
[*Leaf2-Vlanif100] mac-address 0000-5e00-0101
[*Leaf2-Vlanif100] ipv6 nd proxy anyway enable  //使能ND任意代答功能。Leaf捕获服务器发送的NS请求报文后，会使用自己的三层网关接口的接口MAC进行NA应答，这样服务器的上行流量到达Leaf层后，都会进行三层转发
[*Leaf2-Vlanif100] ipv6 nd delete trigger link-down enable  //使能链路Down时ND表项快速删除功能，当成员端口Down时，该端口下的ND表项就会立即删除
[*Leaf2-Vlanif100] ipv6 nd direct-route enable  //使能接口发布NDP Vlink直连路由功能，然后通过路由协议引入直连路由，可以将ND转换的主机路由发布给路由邻居，避免在链路故障时形成路由黑洞
[*Leaf2-Vlanif100] ipv6 nd direct-route preference 1  //配置NDP Vlink直连路由的优先级为1
[*Leaf2-Vlanif100] ipv6 nd direct-route delay 120  //(可选)为了避免由于IPv6 NDP Vlink直连路由建立慢，但是路由引流快导致的流量丢失问题，可以配置延迟发布IPv6 NDP Vlink直连路由的时间
[*Leaf2-Vlanif100] quit
[*Leaf2] commit
```

步骤4 汇聚层和接入层交换机间配置BGP路由协议，实现三层互通。

配置Spine1

```
[~Spine1] bgp 65009
[*Spine1-bgp] router-id 172.16.1.1
[*Spine1-bgp] group leaf external
[*Spine1-bgp] peer 2001:db8:1::2 as-number 65020
[*Spine1-bgp] peer 2001:db8:1::2 group leaf
[*Spine1-bgp] peer 2001:db8:3::2 as-number 65021
```

```
[*Spine1-bgp] peer 2001:db8:3::2 group leaf
[*Spine1-bgp] timer keepalive 10 hold 30
[*Spine1-bgp] ipv6-family unicast
[*Spine1-bgp-af-ipv6] load-balancing as-path-relax  //路由在形成负载分担时不比较相同长度的AS-Path属性，确保不同Leaf发给Spine的路由可以形成负载分担
[*Spine1-bgp-af-ipv6] preference 20 200 10
[*Spine1-bgp-af-ipv6] peer 2001:db8:1::2 enable
[*Spine1-bgp-af-ipv6] peer 2001:db8:3::2 enable
[*Spine1-bgp-af-ipv6] network 2001:db8:1:: 64
[*Spine1-bgp-af-ipv6] network 2001:db8:3:: 64
[*Spine1-bgp-af-ipv6] maximum load-balancing 32
[*Spine1-bgp-af-ipv6] quit
[*Spine1-bgp] quit
[*Spine1] commit
```

配置Spine2

```
[~Spine2] bgp 65009
[*Spine2-bgp] router-id 172.16.2.1
[*Spine2-bgp] group leaf external
[*Spine2-bgp] peer 2001:db8:2::2 as-number 65020
[*Spine2-bgp] peer 2001:db8:2::2 group leaf
[*Spine2-bgp] peer 2001:db8:4::2 as-number 65021
[*Spine2-bgp] peer 2001:db8:4::2 group leaf
[*Spine2-bgp] timer keepalive 10 hold 30
[*Spine2-bgp] ipv6-family unicast
[*Spine2-bgp-af-ipv6] load-balancing as-path-relax  //路由在形成负载分担时不比较相同长度的AS-Path属性，确保不同Leaf发给Spine的路由可以形成负载分担
[*Spine2-bgp-af-ipv6] preference 20 200 10
[*Spine2-bgp-af-ipv6] peer 2001:db8:2::2 enable
[*Spine2-bgp-af-ipv6] peer 2001:db8:4::2 enable
[*Spine2-bgp-af-ipv6] network 2001:db8:2:: 64
[*Spine2-bgp-af-ipv6] network 2001:db8:4:: 64
[*Spine2-bgp-af-ipv6] maximum load-balancing 32
[*Spine2-bgp-af-ipv6] quit
[*Spine2-bgp] quit
[*Spine2] commit
```

配置Leaf1

```
[~Leaf1] bgp 65020
[*Leaf1-bgp] router-id 172.16.3.1
[*Leaf1-bgp] group spine external
[*Leaf1-bgp] peer spine as-number 65009
[*Leaf1-bgp] peer 2001:db8:1::1 as-number 65009
[*Leaf1-bgp] peer 2001:db8:1::1 group spine
[*Leaf1-bgp] peer 2001:db8:2::1 as-number 65009
[*Leaf1-bgp] peer 2001:db8:2::1 group spine
[*Leaf1-bgp] timer keepalive 10 hold 30
[*Leaf1-bgp] ipv6-family unicast
[*Leaf1-bgp-af-ipv6] preference 20 200 10
[*Leaf1-bgp-af-ipv6] peer 2001:db8:1::1 enable
[*Leaf1-bgp-af-ipv6] peer 2001:db8:2::1 enable
[*Leaf1-bgp-af-ipv6] network 2001:db8:1:: 64
[*Leaf1-bgp-af-ipv6] network 2001:db8:2:: 64
[*Leaf1-bgp-af-ipv6] network 2001:db8:100:: 64
[*Leaf1-bgp-af-ipv6] import-route direct //引入直连路由学习到的路由信息，可以根据实际组网情况配置路由策略过滤掉非必要的路由
[*Leaf1-bgp-af-ipv6] maximum load-balancing 32
[*Leaf1-bgp-af-ipv6] quit
[*Leaf1-bgp] quit
[*Leaf1] commit
```

配置Leaf2

```
[~Leaf2] bgp 65021
[*Leaf2-bgp] router-id 172.16.4.1
[*Leaf2-bgp] group spine external
[*Leaf2-bgp] peer spine as-number 65009
[*Leaf2-bgp] peer 2001:db8:3::1 as-number 65009
```



```
[*Leaf2-bgp] peer 2001:db8:3::1 group spine
[*Leaf2-bgp] peer 2001:db8:4::1 as-number 65009
[*Leaf2-bgp] peer 2001:db8:4::1 group spine
[*Leaf2-bgp] timer keepalive 10 hold 30
[*Leaf2-bgp] ipv6-family unicast
[*Leaf2-bgp-af-ipv6] preference 20 200 10
[*Leaf2-bgp-af-ipv6] peer 2001:db8:3::1 enable
[*Leaf2-bgp-af-ipv6] peer 2001:db8:4::1 enable
[*Leaf2-bgp-af-ipv6] network 2001:db8:3:: 64
[*Leaf2-bgp-af-ipv6] network 2001:db8:4:: 64
[*Leaf2-bgp-af-ipv6] network 2001:db8:100:: 64
[*Leaf2-bgp-af-ipv6] import-route direct //引入直连路由学习到的路由信息，可以根据实际组网情况配置路由策略过滤掉非必要的路由
[*Leaf2-bgp-af-ipv6] maximum load-balancing 32
[*Leaf2-bgp-af-ipv6] quit
[*Leaf2-bgp] quit
[*Leaf2] commit
```

说明

本文以Leaf1和Leaf2配置不同AS作为举例说明。对于Leaf1和Leaf2配置相同AS场景，需要增加配置命令**peer allow-as-loop**，以确保不同Leaf的路由经过Spine后，能将路由发布给对方。

步骤5 配置Leaf层交换机Monitor Link组关联上下行接口

配置Leaf1

```
[~Leaf1] monitor-link group 1
[*Leaf1-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf1-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf1-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf1-mtlk-group1] port eth-trunk 20 downlink 2
[*Leaf1-mtlk-group1] timer recover-time 60
[*Leaf1-mtlk-group1] quit
[*Leaf1] commit
```

配置Leaf2

```
[~Leaf2] monitor-link group 1
[*Leaf2-mtlk-group1] port 100ge 1/0/1 uplink
[*Leaf2-mtlk-group1] port 100ge 1/0/2 uplink
[*Leaf2-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf2-mtlk-group1] port eth-trunk 20 downlink 2
[*Leaf2-mtlk-group1] timer recover-time 60
[*Leaf2-mtlk-group1] quit
[*Leaf2] commit
```

----结束

检查配置结果

- 查看设备的Eth-Trunk信息，查看M-LAG Lite已与服务器协商成功，端口状态为Up。

```
[~Leaf1] display eth-trunk 10
Eth-Trunk10's state information is:
Local:
LAG ID: 10                    Working Mode: Static
Preempt Delay: Disabled       Hash Arithmetic: profile default
System Priority: 100          System ID: 00e0-fc00-0000
Least Active-linknumber: 1    Max Active-linknumber: 32
Operating Status: up          Number Of Up Ports In Trunk: 1
Timeout Period: Slow
PortKeyMode: Auto
--------------------------------------------------------------------------------
ActorPortName         Status   PortType PortPri PortNo PortKey PortState Weight
100GE1/0/3            Selected 100GE    32768   1      2625    10111100  1

Partner:
```

```
--------------------------------------------------------------------------------
ActorPortName          SysPri   SystemID        PortPri PortNo PortKey PortState
100GE1/0/3             32768    00e0-fc12-2401  32768   1      2625    10111100
[~Leaf1] display eth-trunk 20
Eth-Trunk20's state information is:
Local:
LAG ID: 20                    Working Mode: Static
Preempt Delay: Disabled         Hash Arithmetic: profile default
System Priority: 100            System ID: 00e0-fc00-0001
Least Active-linknumber: 1      Max Active-linknumber: 32
Operating Status: up            Number Of Up Ports In Trunk: 1
Timeout Period: Slow
PortKeyMode: Auto
--------------------------------------------------------------------------------
ActorPortName          Status   PortType PortPri PortNo PortKey PortState Weight
100GE1/0/4             Selected 100GE    32768   2      5185    10111100  1

Partner:
--------------------------------------------------------------------------------
ActorPortName          SysPri   SystemID        PortPri PortNo PortKey PortState
100GE1/0/4             32768    00e0-fc39-8b01  32768   1      5185    10111100
```

- 查看Server1和Server2是否可以互通。

  # 配置Server1和Server2的IPv6地址分别为2001:db8:100::2/64和2001:db8:100::3/64。

  # 配置完成后，Server1和Server2之间可以互相Ping通。

配置脚本

- Spine1的配置脚本

```
#
sysname Spine1
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1::1/64
 port-isolate l3 enable
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:3::1/64
 port-isolate l3 enable
#
bgp 65009
 router-id 172.16.1.1
 timer keepalive 10 hold 30
 group leaf external
 peer 2001:DB8:1::2 as-number 65020
 peer 2001:DB8:1::2 group leaf
 peer 2001:DB8:3::2 as-number 65021
 peer 2001:DB8:3::2 group leaf
 #
 ipv6-family unicast
  preference 20 200 10
  network 2001:DB8:1:: 64
  network 2001:DB8:3:: 64
  maximum load-balancing 32
  peer 2001:DB8:1::2 enable
  peer 2001:DB8:3::2 enable
  load-balancing as-path-relax
#
return
```

- Spine2的配置脚本

```
#
sysname Spine2
```



```
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:4::1/64
 port-isolate l3 enable
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:2::1/64
 port-isolate l3 enable
#
bgp 65009
 router-id 172.16.2.1
 timer keepalive 10 hold 30
 group leaf external
 peer 2001:DB8:2::2 as-number 65020
 peer 2001:DB8:2::2 group leaf
 peer 2001:DB8:4::2 as-number 65021
 peer 2001:DB8:4::2 group leaf
 #
 ipv6-family unicast
  preference 20 200 10
  network 2001:DB8:2:: 64
  network 2001:DB8:4:: 64
  maximum load-balancing 32
  peer 2001:DB8:2::2 enable
  peer 2001:DB8:4::2 enable
  load-balancing as-path-relax
#
return
```

- Leaf1的配置脚本

```
#
sysname Leaf1
#
vlan batch 100
#
lacp priority 100
#
interface Vlanif100
 ipv6 enable
 ipv6 address 2001:DB8:100::1/64
 ipv6 address FE80::300:3EFF:FE11:985 link-local
 mac-address 0000-5e00-0101
 ipv6 nd proxy anyway enable
 ipv6 nd delete trigger link-down enable
 ipv6 nd direct-route enable
 ipv6 nd direct-route delay 120
 ipv6 nd direct-route preference 1
#
interface Eth-Trunk10
 port link-type trunk
 port trunk pvid vlan 100
 port trunk allow-pass vlan 100
 mode lacp-static
 lacp system-id 00e0-fc00-0000
#
interface Eth-Trunk20
 port link-type trunk
 port trunk pvid vlan 100
 port trunk allow-pass vlan 100
 mode lacp-static
 lacp system-id 00e0-fc00-0001
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:1::2/64
```



```
 port-isolate l3 enable
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:2::2/64
 port-isolate l3 enable
#
interface 100GE1/0/3
 eth-trunk 10
#
interface 100GE1/0/4
 eth-trunk 20
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk20 downlink 2
 timer recover-time 60
#
bgp 65020
 router-id 172.16.3.1
 timer keepalive 10 hold 30
 group spine external
 peer spine as-number 65009
 peer 2001:DB8:1::1 as-number 65009
 peer 2001:DB8:1::1 group spine
 peer 2001:DB8:2::1 as-number 65009
 peer 2001:DB8:2::1 group spine
 #
 ipv6-family unicast
  preference 20 200 10
  network 2001:DB8:1:: 64
  network 2001:DB8:2:: 64
  network 2001:DB8:100:: 64
  import-route direct
  maximum load-balancing 32
  peer 2001:DB8:1::1 enable
  peer 2001:DB8:2::1 enable
#
return
```

- Leaf2的配置脚本

```
#
sysname Leaf2
#
vlan batch 100
#
lacp priority 100
#
interface Vlanif100
 ipv6 enable
 ipv6 address 2001:DB8:100::1/64
 ipv6 address FE80::300:3EFF:FE11:985 link-local
 mac-address 0000-5e00-0101
 ipv6 nd proxy anyway enable
 ipv6 nd delete trigger link-down enable
 ipv6 nd direct-route enable
 ipv6 nd direct-route delay 120
 ipv6 nd direct-route preference 1
#
interface Eth-Trunk10
 port link-type trunk
 port trunk pvid vlan 100
 port trunk allow-pass vlan 100
 mode lacp-static
 lacp system-id 00e0-fc00-0000
 lacp port-id-extension enable
#
```

```
interface Eth-Trunk20
 port link-type trunk
 port trunk pvid vlan 100
 port trunk allow-pass vlan 100
 mode lacp-static
 lacp system-id 00e0-fc00-0001
 lacp port-id-extension enable
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:4::2/64
 port-isolate l3 enable
#
interface 100GE1/0/2
 undo portswitch
 ipv6 enable
 ipv6 address 2001:DB8:3::2/64
 port-isolate l3 enable
#
interface 100GE1/0/3
 eth-trunk 10
#
interface 100GE1/0/4
 eth-trunk 20
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/2 uplink
 port Eth-Trunk10 downlink 1
 port Eth-Trunk20 downlink 2
 timer recover-time 60
#
bgp 65021
 router-id 172.16.4.1
 timer keepalive 10 hold 30
 group spine external
 peer spine as-number 65009
 peer 2001:DB8:3::1 as-number 65009
 peer 2001:DB8:3::1 group spine
 peer 2001:DB8:4::1 as-number 65009
 peer 2001:DB8:4::1 group spine
 #
 ipv6-family unicast
  preference 20 200 10
  network 2001:DB8:3:: 64
  network 2001:DB8:4:: 64
  network 2001:DB8:100:: 64
  import-route direct
  maximum load-balancing 32
  peer 2001:DB8:3::1 enable
  peer 2001:DB8:4::1 enable
#
return
```

1.10 配置 M-LAG 和透明防火墙综合应用示例

适用产品和版本

- CloudEngine系列交换机V300R020C00或更高版本。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

客户希望构建一个稳定的大二层网络，要求双归接入保证可靠性，链路之间进行负载分担提高链路利用率。同时为了满足服务器业务的安全性，串联透明防火墙（SeGW）提供安全防护功能。

- 核心层和汇聚层采用口字型组网，满足安全网关来回路径一致的要求。
- 汇聚层安全网关设备采用透明模式接入，并启用双机热备功能，采用负载分担方式工作，保证业务不中断。
- 汇聚层和接入层部署M-LAG，形成无环拓扑。

部署后的组网如图1-10所示。

- 核心层的Core交换机两台设备间通过10GE链路聚合；
- 汇聚层防火墙与上下游设备之间通过GE接口连接；
- 汇聚层交换机与上下游设备之间通过10GE接口连接；
- 接入层设备由多台设备组成，与汇聚层之间通过10GE接口连接。

本示例中，交换机以CE16804为例，安全网关设备以USG9520为例。

图 1-10 配置 M-LAG 和透明防火墙综合应用物理组网图

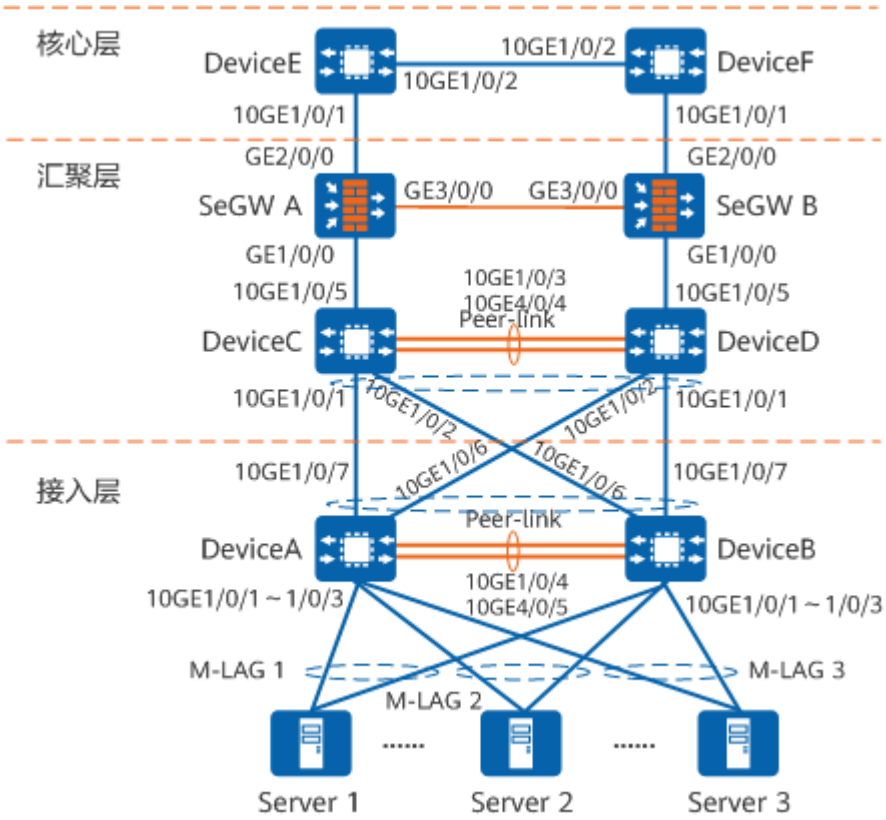


表 1-11 数据准备表

设备名称	接口	IP地址	虚拟MAC地址
DeviceA	管理网口	10.1.1.1/24	-
DeviceB	管理网口	10.1.1.2/24	-
DeviceC	管理网口	10.2.1.1/24	-
	VLANIF11	10.3.1.1/24	0000-5e00-0101
	VLANIF200	10.4.1.1/24	-
	VLANIF300	10.6.1.1/24	-
DeviceD	管理网口	10.2.1.2/24	-
	VLANIF11	10.3.1.1/24	0000-5e00-0101
	VLANIF200	10.5.1.1/24	-
	VLANIF300	10.6.1.2/24	-
DeviceE	VLANIF200	10.4.1.2/24	-
	VLANIF400	10.7.1.1/24	-
DeviceF	VLANIF200	10.5.1.2/24	-
	VLANIF400	10.7.1.2/24	-
SeGWA	GigabitEthernet3/0/0	10.10.0.1/24	-
SeGWB	GigabitEthernet3/0/0	10.10.0.2/24	-

配置思路

采用如下的思路配置：

1. 在汇聚层和接入层交换机DeviceA和DeviceB之间、DeviceC和DeviceD之间配置M-LAG，实现双归接入，正常工作时链路进行负载分担且汇聚层任何一台设备故障对业务均没有影响，保证业务的高可靠性。
   - 在汇聚层配置交换机DeviceC和DeviceD为根桥设备，并在其下行接口上配置根保护功能，保证其接口能够正常转发流量。在接入层交换机配置DeviceA和DeviceB与用户终端相连的接口为边缘端口来加快网络拓扑的收敛时间，并配置BPDU保护功能来加强网络的稳定性。
   - 在汇聚层交换机DeviceC和DeviceD上创建VLANIF接口并配置IP地址，在VLANIF接口上配置相同的IP和虚拟MAC地址，实现双活网关。
2. 在汇聚层配置安全网关设备采用透明模式接入，并启用双机热备功能，采用负载分担方式工作，保证业务不中断。
3. 在核心层和汇聚层交换机DeviceC、DeviceD、DeviceE、DeviceF上使能OSPF，实现三层互通。

操作步骤

步骤1 配置M-LAG

1. 分别在DeviceA、DeviceB、DeviceC和DeviceD上配置双主检测链路、V-STP、DFS Group、peer-link和M-LAG成员接口。

   双主检测链路通过管理网口互通，DFS Group绑定的管理网口IP地址要保证可以相互通信，管理网口下绑定VPN实例，保证双主检测报文与业务流量隔离。

   Peer-Link链路Eth-Trunk接口的成员口建议跨板部署，避免单板单点故障导致Peer-link故障。

   # 配置DeviceA。

   配置接入层交换机连接服务器的Eth-Trunk接口配置成边缘端口，并配置BPDU保护功能。

   服务器上行连接交换机的端口需要绑定在一个聚合链路中且链路聚合模式需要和交换机侧的聚合模式匹配。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] stp mode rstp
[*DeviceA] stp v-stp enable
[*DeviceA] stp flush disable
[*DeviceA] ip vpn-instance VRF-A     //创建VRF-A
[*DeviceA-vpn-instance-VRF-A] ipv4-family
[*DeviceA-vpn-instance-VRF-A-af-ipv4] route-distinguisher 100:1
[*DeviceA-vpn-instance-VRF-A-af-ipv4] vpn-target 111:1 both
[*DeviceA-vpn-instance-VRF-A-af-ipv4] quit
[*DeviceA-vpn-instance-VRF-A] quit
[*DeviceA] interface meth 0/0/0
[*DeviceA-MEth0/0/0] ip binding vpn-instance VRF-A     //将管理网口绑定至VRF-A
[*DeviceA-MEth0/0/0] ip address 10.1.1.1 24
[*DeviceA-MEth0/0/0] quit
[*DeviceA] dfs-group 1
[*DeviceA-dfs-group-1] dual-active detection source ip 10.1.1.1 vpn-instance VRF-A peer ip
10.1.1.2    //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceA-dfs-group-1] priority 150
[*DeviceA-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
```

```
[*DeviceA-dfs-group-1] quit
[*DeviceA] interface eth-trunk 0
[*DeviceA-Eth-Trunk0] trunkport 10ge 1/0/4
[*DeviceA-Eth-Trunk0] trunkport 10ge 4/0/5     //Peer-Link链路Eth-Trunk接口的成员口跨板部署
[*DeviceA-Eth-Trunk0] mode lacp-static
[*DeviceA-Eth-Trunk0] peer-link 1
[*DeviceA-Eth-Trunk0] port vlan exclude 1
[*DeviceA-Eth-Trunk0] quit
[*DeviceA] vlan batch 11
[*DeviceA] interface eth-trunk 10
[*DeviceA-Eth-Trunk10] mode lacp-dynamic
[*DeviceA-Eth-Trunk10] port link-type access
[*DeviceA-Eth-Trunk10] port default vlan 11
[*DeviceA-Eth-Trunk10] trunkport 10ge 1/0/1
[*DeviceA-Eth-Trunk10] dfs-group 1 m-lag 1
[*DeviceA-Eth-Trunk10] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceA-Eth-Trunk10] quit
[*DeviceA] interface eth-trunk 20
[*DeviceA-Eth-Trunk20] mode lacp-dynamic
[*DeviceA-Eth-Trunk20] port link-type access
[*DeviceA-Eth-Trunk20] port default vlan 11
[*DeviceA-Eth-Trunk20] trunkport 10ge 1/0/2
[*DeviceA-Eth-Trunk20] dfs-group 1 m-lag 2
[*DeviceA-Eth-Trunk20] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceA-Eth-Trunk20] quit
[*DeviceA] interface eth-trunk 30
[*DeviceA-Eth-Trunk30] mode lacp-dynamic
[*DeviceA-Eth-Trunk30] port link-type access
[*DeviceA-Eth-Trunk30] port default vlan 11
[*DeviceA-Eth-Trunk30] trunkport 10ge 1/0/3
[*DeviceA-Eth-Trunk30] dfs-group 1 m-lag 3
[*DeviceA-Eth-Trunk30] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceA-Eth-Trunk30] quit
[*DeviceA] stp bpdu-protection     //使能设备边缘端口的BPDU保护功能
[*DeviceA] interface eth-trunk 40
[*DeviceA-Eth-Trunk40] mode lacp-static
[*DeviceA-Eth-Trunk40] port link-type trunk
[*DeviceA-Eth-Trunk40] undo port trunk allow-pass vlan 1
[*DeviceA-Eth-Trunk40] port trunk allow-pass vlan 11
[*DeviceA-Eth-Trunk40] trunkport 10ge 1/0/6 to 1/0/7
[*DeviceA-Eth-Trunk40] dfs-group 1 m-lag 4
[*DeviceA-Eth-Trunk40] quit
[*DeviceA] lacp m-lag priority 10
[*DeviceA] lacp m-lag system-id 00e0-fc00-0000
[*DeviceA] interface 10ge 1/0/9
[*DeviceA-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceA-10GE1/0/9] quit
[*DeviceA] commit
```

配置DeviceB。

配置接入层交换机连接服务器的Eth-Trunk接口配置成边缘端口，并配置BPDU保护功能。

服务器上行连接交换机的端口需要绑定在一个聚合链路中且链路聚合模式需要和交换机侧的聚合模式匹配。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*HUAWEI] commit
[~DeviceB] stp mode rstp
[*DeviceB] stp v-stp enable
[*DeviceB] stp flush disable
[*DeviceB] ip vpn-instance VRF-A     //创建VRF-A
[*DeviceB-vpn-instance-VRF-A] ipv4-family
[*DeviceB-vpn-instance-VRF-A-af-ipv4] route-distinguisher 100:2
[*DeviceB-vpn-instance-VRF-A-af-ipv4] vpn-target 111:1 both
[*DeviceB-vpn-instance-VRF-A-af-ipv4] quit
[*DeviceB-vpn-instance-VRF-A] quit
```

```
[*DeviceB] interface meth 0/0/0
[*DeviceB-MEth0/0/0] ip binding vpn-instance VRF-A     //将管理网口绑定至VRF-A
[*DeviceB-MEth0/0/0] ip address 10.1.1.2 24
[*DeviceB-MEth0/0/0] quit
[*DeviceB] dfs-group 1
[*DeviceB-dfs-group-1] dual-active detection source ip 10.1.1.2 vpn-instance VRF-A peer ip
10.1.1.1     //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceB-dfs-group-1] priority 120
[*DeviceB-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*DeviceB-dfs-group-1] quit
[*DeviceB] interface eth-trunk 0
[*DeviceB-Eth-Trunk0] trunkport 10ge 1/0/4
[*DeviceB-Eth-Trunk0] trunkport 10ge 4/0/5
[*DeviceB-Eth-Trunk0] mode lacp-static
[*DeviceB-Eth-Trunk0] peer-link 1
[*DeviceB-Eth-Trunk0] port vlan exclude 1
[*DeviceB-Eth-Trunk0] quit
[*DeviceB] vlan batch 11
[*DeviceB] interface eth-trunk 10
[*DeviceB-Eth-Trunk10] mode lacp-dynamic
[*DeviceB-Eth-Trunk10] port link-type access
[*DeviceB-Eth-Trunk10] port default vlan 11
[*DeviceB-Eth-Trunk10] trunkport 10ge 1/0/1
[*DeviceB-Eth-Trunk10] dfs-group 1 m-lag 1
[*DeviceB-Eth-Trunk10] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceB-Eth-Trunk10] quit
[*DeviceB] interface eth-trunk 20
[*DeviceB-Eth-Trunk20] mode lacp-dynamic
[*DeviceB-Eth-Trunk20] port link-type access
[*DeviceB-Eth-Trunk20] port default vlan 11
[*DeviceB-Eth-Trunk20] trunkport 10ge 1/0/2
[*DeviceB-Eth-Trunk20] dfs-group 1 m-lag 2
[*DeviceB-Eth-Trunk20] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceB-Eth-Trunk20] quit
[*DeviceB] interface eth-trunk 30
[*DeviceB-Eth-Trunk30] mode lacp-dynamic
[*DeviceB-Eth-Trunk30] port link-type access
[*DeviceB-Eth-Trunk30] port default vlan 11
[*DeviceB-Eth-Trunk30] trunkport 10ge 1/0/3
[*DeviceB-Eth-Trunk30] dfs-group 1 m-lag 3
[*DeviceB-Eth-Trunk30] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceB-Eth-Trunk30] quit
[*DeviceB] stp bpdu-protection     //使能设备边缘端口的BPDU保护功能
[*DeviceB] interface eth-trunk 40
[*DeviceB-Eth-Trunk40] mode lacp-static
[*DeviceB-Eth-Trunk40] port link-type trunk
[*DeviceB-Eth-Trunk40] undo port trunk allow-pass vlan 1
[*DeviceB-Eth-Trunk40] port trunk allow-pass vlan 11
[*DeviceB-Eth-Trunk40] trunkport 10ge 1/0/6 to 1/0/7
[*DeviceB-Eth-Trunk40] dfs-group 1 m-lag 4
[*DeviceB-Eth-Trunk40] quit
[*DeviceB] lacp m-lag priority 10
[*DeviceB] lacp m-lag system-id 00e0-fc00-0000
[*DeviceB] interface 10ge 1/0/9
[*DeviceB-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceB-10GE1/0/9] quit
[*DeviceB] commit
```

配置DeviceC。

配置汇聚层交换机为STP网络的根桥设备，连接接入层交换机的Eth-Trunk接口配置根保护功能，保证其接口能够正常转发流量。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceC
[*HUAWEI] commit
[~DeviceC] stp mode rstp
[*DeviceC] stp root primary     //配置汇聚层设备为STP网络的根桥
[*DeviceC] stp bridge-address 00e0-fc00-5678     //配置根桥的桥MAC（DFS主设备的MAC地址）
[*DeviceC] stp v-stp enable
```



```
[*DeviceC] stp flush disable
[*DeviceC] ip vpn-instance VRF-B     //创建VRF-B
[*DeviceC-vpn-instance-VRF-B] ipv4-family
[*DeviceC-vpn-instance-VRF-B-af-ipv4] route-distinguisher 101:1
[*DeviceC-vpn-instance-VRF-B-af-ipv4] vpn-target 111:1 both
[*DeviceC-vpn-instance-VRF-B-af-ipv4] quit
[*DeviceC-vpn-instance-VRF-B] quit
[*DeviceC] interface meth 0/0/0
[*DeviceC-MEth0/0/0] ip binding vpn-instance VRF-B     //将管理网口绑定至VRF-B
[*DeviceC-MEth0/0/0] ip address 10.2.1.1 24
[*DeviceC-MEth0/0/0] quit
[*DeviceC] dfs-group 1
[*DeviceC-dfs-group-1] dual-active detection source ip 10.2.1.1 vpn-instance VRF-B peer ip
10.2.1.2     //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceC-dfs-group-1] priority 150
[*DeviceC-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*DeviceC-dfs-group-1] quit
[*DeviceC] interface eth-trunk 0
[*DeviceC-Eth-Trunk0] trunkport 10ge 1/0/3
[*DeviceC-Eth-Trunk0] trunkport 10ge 4/0/4     //Peer-Link链路Eth-Trunk接口的成员口跨板部署
[*DeviceC-Eth-Trunk0] mode lacp-static
[*DeviceC-Eth-Trunk0] peer-link 1
[*DeviceC-Eth-Trunk0] port vlan exclude 1
[*DeviceC-Eth-Trunk0] quit
[*DeviceC] vlan batch 11
[*DeviceC] interface eth-trunk 30
[*DeviceC-Eth-Trunk30] mode lacp-static
[*DeviceC-Eth-Trunk30] port link-type trunk
[*DeviceC-Eth-Trunk30] undo port trunk allow-pass vlan 1
[*DeviceC-Eth-Trunk30] port trunk allow-pass vlan 11
[*DeviceC-Eth-Trunk30] trunkport 10ge 1/0/1 to 1/0/2
[*DeviceC-Eth-Trunk30] dfs-group 1 m-lag 1
[*DeviceC-Eth-Trunk30] stp root-protection   //使能当前端口的根保护功能
[*DeviceC-Eth-Trunk30] quit
[*DeviceC] lacp m-lag priority 10
[*DeviceC] lacp m-lag system-id 00e0-fc00-0001
[*DeviceC] commit
```

配置DeviceD。

配置汇聚层交换机为STP网络的根桥设备，连接接入层交换机的Eth-Trunk接口配置根保护功能，保证其接口能够正常转发流量。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceD
[*HUAWEI] commit
[~DeviceD] stp mode rstp
[*DeviceD] stp root primary     //配置汇聚层设备为STP网络的根桥
[*DeviceD] stp bridge-address 00e0-fc00-5678     //配置根桥的桥MAC（DFS主设备的MAC地址）
[*DeviceD] stp v-stp enable
[*DeviceD] stp flush disable
[*DeviceD] ip vpn-instance VRF-B     //创建VRF-B
[*DeviceD-vpn-instance-VRF-B] ipv4-family
[*DeviceD-vpn-instance-VRF-B-af-ipv4] route-distinguisher 101:2
[*DeviceD-vpn-instance-VRF-B-af-ipv4] vpn-target 111:1 both
[*DeviceD-vpn-instance-VRF-B-af-ipv4] quit
[*DeviceD-vpn-instance-VRF-B] quit
[*DeviceD] interface meth 0/0/0
[*DeviceD-MEth0/0/0] ip binding vpn-instance VRF-B     //将管理网口绑定至VRF-B
[*DeviceD-MEth0/0/0] ip address 10.2.1.2 24
[*DeviceD-MEth0/0/0] quit
[*DeviceD] dfs-group 1
[*DeviceD-dfs-group-1] dual-active detection source ip 10.2.1.2 vpn-instance VRF-B peer ip
10.2.1.1     //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceD-dfs-group-1] priority 120
[*DeviceD-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*DeviceD-dfs-group-1] quit
[*DeviceD] interface eth-trunk 0
[*DeviceD-Eth-Trunk0] trunkport 10ge 1/0/3
[*DeviceD-Eth-Trunk0] trunkport 10ge 4/0/4     //Peer-Link链路Eth-Trunk接口的成员口跨板部署
```

```
[*DeviceD-Eth-Trunk0] mode lacp-static
[*DeviceD-Eth-Trunk0] peer-link 1
[*DeviceD-Eth-Trunk0] port vlan exclude 1
[*DeviceD-Eth-Trunk0] quit
[*DeviceD] vlan batch 11
[*DeviceD] interface eth-trunk 30
[*DeviceD-Eth-Trunk30] mode lacp-static
[*DeviceD-Eth-Trunk30] port link-type trunk
[*DeviceD-Eth-Trunk30] undo port trunk allow-pass vlan 1
[*DeviceD-Eth-Trunk30] port trunk allow-pass vlan 11
[*DeviceD-Eth-Trunk30] trunkport 10ge 1/0/1 to 1/0/2
[*DeviceD-Eth-Trunk30] dfs-group 1 m-lag 1
[*DeviceD-Eth-Trunk30] stp root-protection //使能当前端口的根保护功能
[*DeviceD-Eth-Trunk30] quit
[*DeviceD] lacp m-lag priority 10
[*DeviceD] lacp m-lag system-id 00e0-fc00-0001
[*DeviceD] commit
```

2. 在DeviceC和DeviceD上创建VLANIF接口并配置IP地址和虚拟MAC地址，配置双活网关 。

   # 配置DeviceC。

   ```
   [~DeviceC] interface vlanif 11
   [*DeviceC-Vlanif11] ip address 10.3.1.1 24
   [*DeviceC-Vlanif11] mac-address 0000-5e00-0101
   [*DeviceC-Vlanif11] quit
   [*DeviceC] commit
   ```

   # 配置DeviceD。

   ```
   [~DeviceD] interface vlanif 11
   [*DeviceD-Vlanif11] ip address 10.3.1.1 24
   [*DeviceD-Vlanif11] mac-address 0000-5e00-0101
   [*DeviceD-Vlanif11] quit
   [*DeviceD] commit
   ```

步骤2 配置SeGWA和SeGWB为透明模式双机热备份

1. 将SeGWA和SeGWB的各业务接口切换成二层接口并加入同一VLAN。

   # 配置SeGWA。

   ```
   <USG9000> system-view
   [USG9000] sysname SeGWA
   [SeGWA] interface GigabitEthernet 1/0/0
   [SeGWA-GigabitEthernet1/0/0] portswitch
   [SeGWA-GigabitEthernet1/0/0] quit
   [SeGWA] interface GigabitEthernet 2/0/0
   [SeGWA-GigabitEthernet2/0/0] portswitch
   [SeGWA-GigabitEthernet2/0/0] quit
   [SeGWA] vlan 200
   [SeGWA-vlan200] port GigabitEthernet 1/0/0
   [SeGWA-vlan200] port GigabitEthernet 2/0/0
   [SeGWA-vlan200] quit
   ```

   # 配置SeGWB。

   ```
   <USG9000> system-view
   [USG9000] sysname SeGWB
   [SeGWB] interface GigabitEthernet 1/0/0
   [SeGWB-GigabitEthernet1/0/0] portswitch
   [SeGWB-GigabitEthernet1/0/0] quit
   [SeGWB] interface GigabitEthernet 2/0/0
   [SeGWB-GigabitEthernet2/0/0] portswitch
   [SeGWB-GigabitEthernet2/0/0] quit
   [SeGWB] vlan 200
   [SeGWB-vlan200] port GigabitEthernet 1/0/0
   [SeGWB-vlan200] port GigabitEthernet 2/0/0
   [SeGWB-vlan200] quit
   ```

2. 配置SeGWA和SeGWB的心跳接口的IP地址。

   # 配置SeGWA。

   ```
   [SeGWA] interface GigabitEthernet 3/0/0
   [SeGWA-GigabitEthernet3/0/0] ip address 10.10.0.1 24
   [SeGWA-GigabitEthernet3/0/0] quit
   ```

   # 配置SeGWB。

   ```
   [SeGWB] interface GigabitEthernet 3/0/0
   [SeGWB-GigabitEthernet3/0/0] ip address 10.10.0.2 24
   [SeGWB-GigabitEthernet3/0/0] quit
   ```

3. 将SeGWA和SeGWB的上行业务接口加入untrust区域，下行业务接口加入trust区域，心跳口加入dmz区域。

   # 配置SeGWA。

   ```
   [SeGWA] firewall zone untrust
   [SeGWA-zone-untrust] add interface GigabitEthernet 2/0/0
   [SeGWA-zone-untrust] quit
   [SeGWA] firewall zone trust
   [SeGWA-zone-trust] add interface GigabitEthernet 1/0/0
   [SeGWA-zone-trust] quit
   [SeGWA] firewall zone dmz
   [SeGWA-zone-dmz] add interface GigabitEthernet 3/0/0
   [SeGWA-zone-dmz] quit
   ```

   # 配置SeGWB。

   ```
   [SeGWB] firewall zone untrust
   [SeGWB-zone-untrust] add interface GigabitEthernet 2/0/0
   [SeGWB-zone-untrust] quit
   [SeGWB] firewall zone trust
   [SeGWB-zone-trust] add interface GigabitEthernet 1/0/0
   [SeGWB-zone-trust] quit
   [SeGWB] firewall zone dmz
   [SeGWB-zone-dmz] add interface GigabitEthernet 3/0/0
   [SeGWB-zone-dmz] quit
   ```

4. 配置VGMP组监控业务接口所在VLAN，指定心跳接口，启用双机热备。

   # 配置SeGWA。

   ```
   [SeGWA] hrp track vlan 200
   [SeGWA] hrp interface GigabitEthernet 3/0/0 remote 10.10.0.2
   [SeGWA] hrp enable
   [SeGWA] hrp mirror session enable
   ```

   # 配置SeGWB。

   ```
   [SeGWB] hrp track vlan 200
   [SeGWB] hrp interface GigabitEthernet 3/0/0 remote 10.10.0.1
   [SeGWB] hrp enable
   [SeGWB] hrp mirror session enable
   ```

5. 双机热备功能配置完成后，需要在SeGWA上配置安全策略、IPS、攻击防范等安全功能。SeGWA的配置会自动备份到SeGWB。具体配置请参考安全网关设备的相关资料，这里不做具体介绍。

步骤3 在DeviceC、DeviceD、DeviceE和DeviceF上使能OSPF。

1. 配置DeviceC、DeviceD、DeviceE和DeviceF上的接口加入VLAN及对应VLANIF接口的IP地址。

   # 配置DeviceC。

   ```
   [~DeviceC] vlan batch 200 300
   [*DeviceC] interface 10ge 1/0/5
   ```

```
[*DeviceC-10GE1/0/5] port link-type trunk
[*DeviceC-10GE1/0/5] undo port trunk allow-pass vlan 1
[*DeviceC-10GE1/0/5] port trunk allow-pass vlan 200
[*DeviceC-10GE1/0/5] quit
[*DeviceC] interface eth-trunk 0
[*DeviceC-Eth-Trunk0] port vlan exclude 200     //配置peer-link接口不允许通过VLAN200
[*DeviceC-Eth-Trunk0] quit
[*DeviceC] interface vlanif 200
[*DeviceC-Vlanif200] ospf network-type p2p
[*DeviceC-Vlanif200] ip address 10.4.1.1 24
[*DeviceC-Vlanif200] quit
[*DeviceC] interface vlanif 300
[*DeviceC-Vlanif300] ospf network-type p2p
[*DeviceC-Vlanif300] ip address 10.6.1.1 24
[*DeviceC-Vlanif300] quit
[*DeviceC] interface 10ge 1/0/9
[*DeviceC-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceC-10GE1/0/9] quit
[*DeviceC] commit
```

配置DeviceD。

```
[~DeviceD] vlan batch 200 300
[*DeviceD] interface 10ge 1/0/5
[*DeviceD-10GE1/0/5] port link-type trunk
[*DeviceD-10GE1/0/5] undo port trunk allow-pass vlan 1
[*DeviceD-10GE1/0/5] port trunk allow-pass vlan 200
[*DeviceD-10GE1/0/5] quit
[*DeviceD] interface eth-trunk 0
[*DeviceD-Eth-Trunk0] port vlan exclude 200     //配置peer-link接口不允许通过VLAN200
[*DeviceD-Eth-Trunk0] quit
[*DeviceD] interface vlanif 200
[*DeviceD-Vlanif200] ospf network-type p2p
[*DeviceD-Vlanif200] ip address 10.5.1.1 24
[*DeviceD-Vlanif200] quit
[*DeviceD] interface vlanif 300
[*DeviceD-Vlanif300] ospf network-type p2p
[*DeviceD-Vlanif300] ip address 10.6.1.2 24
[*DeviceD-Vlanif300] quit
[*DeviceD] interface 10ge 1/0/9
[*DeviceD-10GE1/0/9] shutdown
[*DeviceD-10GE1/0/9] quit
[*DeviceD] commit
```

配置DeviceE。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceE
[*HUAWEI] commit
[~DeviceE] vlan batch 200 400
[*DeviceE] interface 10ge 1/0/1
[*DeviceE-10GE1/0/1] port link-type trunk
[*DeviceE-10GE1/0/1] undo port trunk allow-pass vlan 1
[*DeviceE-10GE1/0/1] port trunk allow-pass vlan 200
[*DeviceE-10GE1/0/1] quit
[*DeviceE] interface 10ge 1/0/2
[*DeviceE-10GE1/0/2] port link-type trunk
[*DeviceE-10GE1/0/2] undo port trunk allow-pass vlan 1
[*DeviceE-10GE1/0/2] port trunk allow-pass vlan 400
[*DeviceE-10GE1/0/2] quit
[*DeviceE] interface vlanif 200
[*DeviceE-Vlanif200] ospf network-type p2p
[*DeviceE-Vlanif200] ip address 10.4.1.2 24
[*DeviceE-Vlanif200] quit
[*DeviceE] interface vlanif 400
[*DeviceE-Vlanif400] ospf network-type p2p
[*DeviceE-Vlanif400] ip address 10.7.1.1 24
[*DeviceE-Vlanif400] quit
[*DeviceE] interface 10ge 1/0/9
[*DeviceE-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
```



```
[*DeviceE-10GE1/0/9] quit
[*DeviceE] commit
```

配置DeviceF。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceF
[*HUAWEI] commit
[~DeviceF] vlan batch 200 400
[*DeviceF] interface 10ge 1/0/1
[*DeviceF-10GE1/0/1] port link-type trunk
[*DeviceF-10GE1/0/1] undo port trunk allow-pass vlan 1
[*DeviceF-10GE1/0/1] port trunk allow-pass vlan 200
[*DeviceF-10GE1/0/1] quit
[*DeviceF] interface 10ge 1/0/2
[*DeviceF-10GE1/0/2] port link-type trunk
[*DeviceF-10GE1/0/2] undo port trunk allow-pass vlan 1
[*DeviceF-10GE1/0/2] port trunk allow-pass vlan 400
[*DeviceF-10GE1/0/2] quit
[*DeviceF] interface vlanif 200
[*DeviceF-Vlanif200] ospf network-type p2p
[*DeviceF-Vlanif200] ip address 10.5.1.2 24
[*DeviceF-Vlanif200] quit
[*DeviceF] interface vlanif 400
[*DeviceF-Vlanif400] ospf network-type p2p
[*DeviceF-Vlanif400] ip address 10.7.1.2 24
[*DeviceF-Vlanif400] quit
[*DeviceF] interface 10ge 1/0/9
[*DeviceF-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceF-10GE1/0/9] quit
[*DeviceF] commit
```

2. 配置DeviceC、DeviceD、DeviceE和DeviceF的OSPF功能，使三层可以通信。

   # 配置DeviceC。

   ```
   [~DeviceC] ospf 1
   [~DeviceC-ospf-1] import-route direct     //引入直连路由学习到的路由信息，可以根据实际组网情况配置路由策略过滤掉非必要的路由
   [*DeviceC-ospf-1] area 0
   [*DeviceC-ospf-1-area-0.0.0.0] network 10.4.1.0 0.0.0.255
   [*DeviceC-ospf-1-area-0.0.0.0] network 10.6.1.0 0.0.0.255
   [*DeviceC-ospf-1-area-0.0.0.0] quit
   [*DeviceC-ospf-1] quit
   [*DeviceC] commit
   ```

   # 配置DeviceD。

   ```
   [~DeviceD] ospf 1
   [~DeviceD-ospf-1] import-route direct     //引入直连路由学习到的路由信息，可以根据实际组网情况配置路由策略过滤掉非必要的路由
   [*DeviceD-ospf-1] area 0
   [*DeviceD-ospf-1-area-0.0.0.0] network 10.5.1.0 0.0.0.255
   [*DeviceD-ospf-1-area-0.0.0.0] network 10.6.1.0 0.0.0.255
   [*DeviceD-ospf-1-area-0.0.0.0] quit
   [*DeviceD-ospf-1] quit
   [*DeviceD] commit
   ```

   # 配置DeviceE。

   ```
   [~DeviceE] ospf 1
   [*DeviceE-ospf-1] area 0
   [*DeviceE-ospf-1-area-0.0.0.0] network 10.4.1.0 0.0.0.255
   [*DeviceE-ospf-1-area-0.0.0.0] network 10.7.1.0 0.0.0.255
   [*DeviceE-ospf-1-area-0.0.0.0] quit
   [*DeviceE-ospf-1] quit
   [*DeviceE] commit
   ```

   # 配置DeviceF。


```
[~DeviceF] ospf 1
[*DeviceF-ospf-1] area 0
[*DeviceF-ospf-1-area-0.0.0.0] network 10.5.1.0 0.0.0.255
[*DeviceF-ospf-1-area-0.0.0.0] network 10.7.1.0 0.0.0.255
[*DeviceF-ospf-1-area-0.0.0.0] quit
[*DeviceF-ospf-1] quit
[*DeviceF] commit
```

----结束

检查配置结果

1. 执行命令**display dfs-group**，查看M-LAG的相关信息。

   # 查看DFS Group编号为1的M-LAG信息。

```
[~DeviceA] display dfs-group 1 m-lag
*                : Local node
Heart beat state     : OK
Node 1 *
  Dfs-Group ID       : 1
  Priority           : 150
  Dual-active Address : 10.1.1.1
  VPN-Instance       :  public net
  State              : Master
  Causation          : -
  System ID          : 00e0-fc00-1234
  SysName            : DeviceA
  Version            : V300R023C00
  Device Type        :  CE16800
Node 2
  Dfs-Group ID       : 1
  Priority           : 120
  Dual-active Address : 10.1.1.2
  VPN-Instance       :  public net
  State              : Backup
  Causation          : -
  System ID          : 00e0-fc00-1235
  SysName            : DeviceB
  Version            : V300R023C00
  Device Type        : CE16800
[~DeviceC] display dfs-group 1 m-lag
*                : Local node
Heart beat state     : OK
Node 1 *
  Dfs-Group ID       : 1
  Priority           : 150
  Dual-active Address : 10.2.1.1
  VPN-Instance       :  public net
  State              : Master
  Causation          : -
  System ID          : 00e0-fc00-5678
  SysName            : DeviceA
  Version            : V300R023C00
  Device Type        :  CE16800
Node 2
  Dfs-Group ID       : 1
  Priority           : 120
  Dual-active Address : 10.2.1.2
  VPN-Instance       :  public net
  State              : Backup
  Causation          : -
  System ID          : 00e0-fc00-5679
  SysName            : DeviceB
  Version            : V300R023C00
  Device Type        : CE16800
```

   # 查看DeviceA上的M-LAG信息。

```
[~DeviceA] display dfs-group 1 node 1 m-lag brief
* - Local node
```

```
M-Lag ID    Interface      Port State   Status            Consistency-check
      1     Eth-Trunk 10   Up           active(*)-active  --
      2     Eth-Trunk 20   Up           active(*)-active  --
      3     Eth-Trunk 30   Up           active(*)-active  --
      4     Eth-Trunk 40   Up           active(*)-active  --
```

查看DeviceC上的M-LAG信息。

```
[~DeviceC] display dfs-group 1 node 2 m-lag brief
* - Local node

M-Lag ID    Interface      Port State   Status            Consistency-check
      1     Eth-Trunk 30   Up           active(*)-active  --
```

通过以上显示信息可以看到，“Heart beat state”的状态是“OK”，表明双主检测状态正常；DeviceA和DeviceC作为Node 1，优先级为150，“State”的状态是“Master”；DeviceB和DeviceD作为Node 2，优先级为120，“State”的状态是“Backup”。同时“Causation”的状态是“-”，Node 1的“Port State”状态为“Up”，Node 2的“Port State”状态为“Up”，且Node 1和Node 2的M-LAG状态均为“active”，表明M-LAG的配置正确。

2. 在SeGWA上执行**display hrp state**命令，检查当前HRP的状态，显示以下信息表示HRP建立成功。

```
HRP_M[SeGWA] display hrp state
 Role: active， peer: active
 Running priority: 51008， peer: 51008
 Core state: normal， peer: normal
 Backup channel usage: 0%
 Stable time: 0 days， 18 hours， 41 minutes
```

配置脚本

- DeviceA的配置脚本

```
#
sysname DeviceA
#
dfs-group 1
 priority 150
 dual-active detection source ip 10.1.1.1 vpn-instance VRF-A peer 10.1.1.2
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 11
#
stp mode rstp
stp v-stp enable
stp bpdu-protection
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0000
lacp m-lag priority 10
#
ip vpn-instance VRF-A
 ipv4-family
  route-distinguisher 100:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface MEth0/0/0
 ip binding vpn-instance VRF-A
 ip address 10.1.1.1 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
```

```
interface Eth-Trunk10
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 1
#
interface Eth-Trunk20
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 2
#
interface Eth-Trunk30
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 3
#
interface Eth-Trunk40
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
 mode lacp-static
 dfs-group 1 m-lag 4
#
interface 10GE1/0/1
 eth-trunk 10
#
interface 10GE1/0/2
 eth-trunk 20
#
interface 10GE1/0/3
 eth-trunk 30
#
interface 10GE1/0/4
 eth-trunk 0
#
interface 10GE1/0/6
 eth-trunk 40
#
interface 10GE1/0/7
 eth-trunk 40
#
interface 10GE1/0/9
 shutdown
#
interface 10GE4/0/5
 eth-trunk 0
#
return
```

- DeviceB的配置文件

```
#
sysname DeviceB
#
dfs-group 1
 priority 120
 dual-active detection source ip 10.1.1.2 vpn-instance VRF-A peer 10.1.1.1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 11
#
stp mode rstp
stp v-stp enable
stp bpdu-protection
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0000
lacp m-lag priority 10
```



```
#
ip vpn-instance VRF-A
 ipv4-family
  route-distinguisher 100:2
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface MEth0/0/0
 ip binding vpn-instance VRF-A
 ip address 10.1.1.2 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 1
#
interface Eth-Trunk20
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 2
#
interface Eth-Trunk30
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 3
#
interface Eth-Trunk40
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
 mode lacp-static
 dfs-group 1 m-lag 4
#
interface 10GE1/0/1
 eth-trunk 10
#
interface 10GE1/0/2
 eth-trunk 20
#
interface 10GE1/0/3
 eth-trunk 30
#
interface 10GE1/0/4
 eth-trunk 0
#
interface 10GE1/0/6
 eth-trunk 40
#
interface 10GE1/0/7
 eth-trunk 40
#
interface 10GE1/0/9
 shutdown
#
interface 10GE4/0/5
 eth-trunk 0
#
return
```

- DeviceC的配置文件

```
#
sysname DeviceC
```



```
#
dfs-group 1
 priority 150
 dual-active detection source ip 10.2.1.1 vpn-instance VRF-B peer 10.2.1.2
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 11 200 300
#
stp bridge-address 00e0-fc00-5678
stp mode rstp
stp v-stp enable
stp instance 0 root primary
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0001
lacp m-lag priority 10
#
ip vpn-instance VRF-B
 ipv4-family
  route-distinguisher 101:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface Vlanif11
 ip address 10.3.1.1 255.255.255.0
 mac-address 0000-5e00-0101
#
interface Vlanif200
 ip address 10.4.1.1 255.255.255.0
 ospf network-type p2p
#
interface Vlanif300
 ip address 10.6.1.1 255.255.255.0
 ospf network-type p2p
#
interface MEth0/0/0
 ip binding vpn-instance VRF-B
 ip address 10.2.1.1 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1 200
#
interface Eth-Trunk30
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
 stp root-protection
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface 10GE1/0/1
 eth-trunk 30
#
interface 10GE1/0/2
 eth-trunk 30
#
interface 10GE1/0/3
 eth-trunk 0
#
interface 10GE1/0/5
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 200
#
interface 10GE1/0/9
 shutdown
```

```
#
interface 10GE4/0/4
 eth-trunk 0
#
ospf 1
 import-route direct
 area 0.0.0.0
  network 10.4.1.0 0.0.0.255
  network 10.6.1.0 0.0.0.255
#
return
```

- DeviceD的配置文件

```
#
sysname DeviceD
#
dfs-group 1
 priority 120
 dual-active detection source ip 10.2.1.2 vpn-instance VRF-B peer 10.2.1.1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 11 200 300
#
stp bridge-address 00e0-fc00-5678
stp mode rstp
stp v-stp enable
stp instance 0 root primary
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0001
lacp m-lag priority 10
#
ip vpn-instance VRF-B
 ipv4-family
  route-distinguisher 101:2
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface Vlanif11
 ip address 10.3.1.1 255.255.255.0
 mac-address 0000-5e00-0101
#
interface Vlanif200
 ip address 10.5.1.1 255.255.255.0
 ospf network-type p2p
#
interface Vlanif300
 ip address 10.6.1.2 255.255.255.0
 ospf network-type p2p
#
interface MEth0/0/0
 ip binding vpn-instance VRF-B
 ip address 10.2.1.2 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1 200
#
interface Eth-Trunk30
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
 stp root-protection
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface 10GE1/0/1
 eth-trunk 30
```



```
#
interface 10GE1/0/2
 eth-trunk 30
#
interface 10GE1/0/3
 eth-trunk 0
#
interface 10GE1/0/5
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 200
#
interface 10GE1/0/9
 shutdown
#
interface 10GE4/0/4
 eth-trunk 0
#
ospf 1
 import-route direct
 area 0.0.0.0
  network 10.5.1.0 0.0.0.255
  network 10.6.1.0 0.0.0.255
#
return
```

- DeviceE的配置文件

```
#
sysname DeviceE
#
vlan batch 200 400
#
interface Vlanif200
 ip address 10.4.1.2 255.255.255.0
 ospf network-type p2p
#
interface Vlanif400
 ip address 10.7.1.1 255.255.255.0
 ospf network-type p2p
#
interface 10GE1/0/1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 200
#
interface 10GE1/0/2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 400
#
interface 10GE1/0/9
 shutdown
#
ospf 1
 area 0.0.0.0
  network 10.4.1.0 0.0.0.255
  network 10.7.1.0 0.0.0.255
#
return
```

- DeviceF的配置文件

```
#
sysname DeviceF
#
vlan batch 200 400
#
interface Vlanif200
 ip address 10.5.1.2 255.255.255.0
 ospf network-type p2p
#
```



```
interface Vlanif400
 ip address 10.7.1.2 255.255.255.0
 ospf network-type p2p
#
interface 10GE1/0/1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 200
#
interface 10GE1/0/2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 400
#
interface 10GE1/0/9
 shutdown
#
ospf 1
 area 0.0.0.0
  network 10.5.1.0 0.0.0.255
  network 10.7.1.0 0.0.0.255
#
return
```

- SeGWA的配置文件

```
#
sysname SeGWA
#
 hrp enable
 hrp track vlan 200
 hrp mirror session enable
 hrp interface GigabitEthernet 3/0/0 remote 10.10.0.2
#
vlan 200
 port GigabitEthernet 1/0/0
 port GigabitEthernet 2/0/0
#
interface GigabitEthernet 1/0/0
 portswitch
#
interface GigabitEthernet 2/0/0
 portswitch
#
interface GigabitEthernet3/0/0
 ip address 10.10.0.1 24
#
firewall zone trust
 set priority 85
 add interface GigabitEthernet 1/0/0
#
firewall zone dmz
 set priority 50
 add interface GigabitEthernet 3/0/0
#
firewall zone untrust
 set priority 5
 add interface GigabitEthernet 2/0/0
#
return
```

- SeGWB的配置文件

```
#
sysname SeGWB
#
 hrp enable
 hrp track vlan 200
 hrp mirror session enable
 hrp interface GigabitEthernet 3/0/0 remote 10.10.0.1
#
vlan 200
```

```
 port GigabitEthernet 1/0/0
 port GigabitEthernet 2/0/0
#
interface GigabitEthernet 1/0/0
 portswitch
#
interface GigabitEthernet 2/0/0
 portswitch
#
interface GigabitEthernet3/0/0
 ip address 10.10.0.2 24
#
firewall zone trust
 set priority 85
 add interface GigabitEthernet 1/0/0
#
firewall zone dmz
 set priority 50
 add interface GigabitEthernet 3/0/0
#
firewall zone untrust
 set priority 5
 add interface GigabitEthernet 2/0/0
#
return
```

1.11 配置 M-LAG+旁挂防火墙综合应用示例

适用产品和版本

- CloudEngine系列交换机V300R020C00或更高版本。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

客户希望构建一个稳定的大二层网络。要求双归接入保证可靠性，同时链路之间进行负载分担提高链路利用率。同时为了满足服务器业务的安全性，在汇聚层旁挂防火墙（SeGW）提供安全防护功能。

- 核心层和汇聚层采用全交叉互联组网，形成等价多路径路由ECMP进行负载分担转发（与汇聚层下行部署的M-LAG形成本地优先转发）。
- 汇聚层安全网关设备采用路由模式（静态路由）接入，并启用双机热备功能，采用主备备份方式工作，增强网络的健壮性。
- 汇聚层和接入层部署M-LAG，形成无环拓扑。

部署后的组网如**图1-11**所示：

- 核心层的Core交换机与下游汇聚层设备间通过10GE连接；
- 汇聚层防火墙与上下游设备之间通过GE接口连接；
- 汇聚层交换机与上下游设备之间通过10GE接口连接；
- 接入层设备由多台设备组成，与汇聚层之间通过10GE接口连接。

本示例中，交换机以CE16804为例，安全网关设备以USG9520为例。

图 1-11 配置 M-LAG+旁挂防火墙综合应用组网图

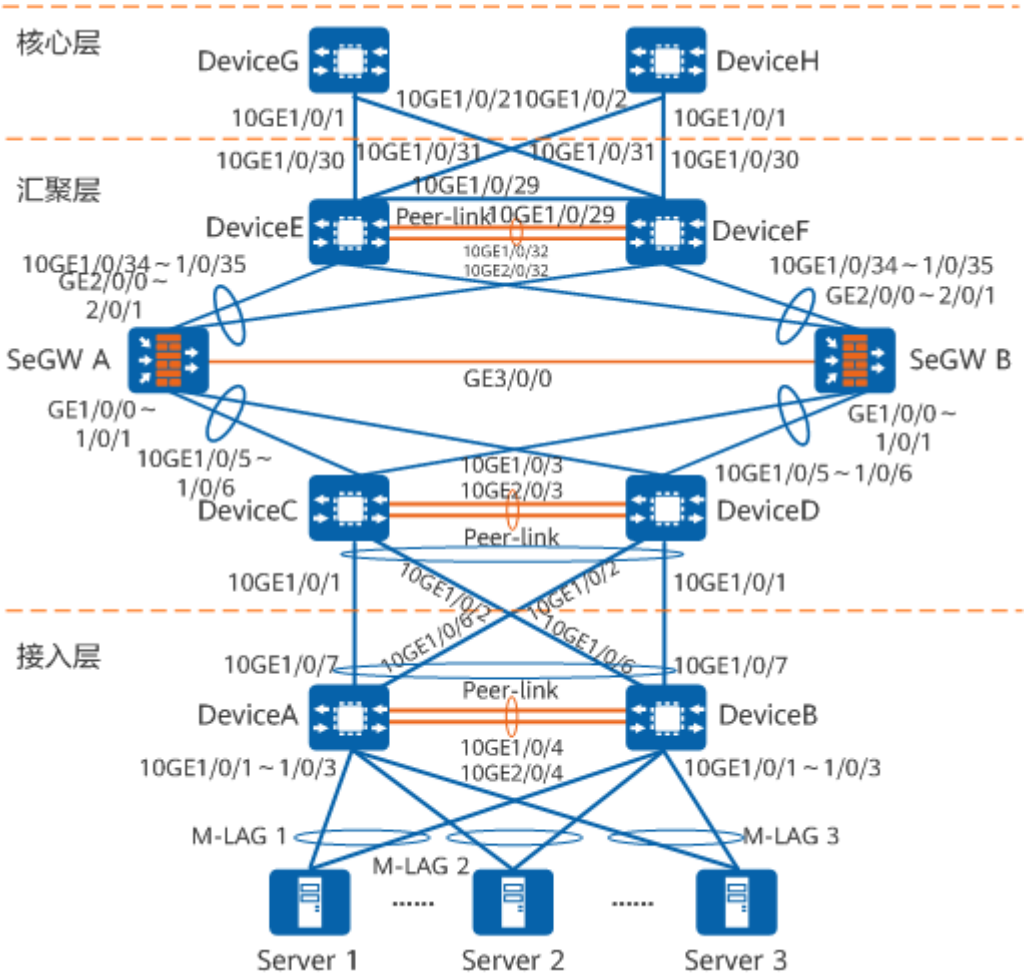


表 1-12 数据准备表

设备名称	接口	IP地址	虚拟MAC地址
DeviceA	管理网口	10.1.1.1/24	-
DeviceB	管理网口	10.1.1.2/24	-
DeviceC	管理网口	10.2.1.1/24	-
	VLANIF11	10.4.1.1/24	0000-5e00-0102
	VLANIF20	10.5.1.1/24	0000-5e00-0103
DeviceD	管理网口	10.2.1.2/24	-
	VLANIF11	10.4.1.1/24	0000-5e00-0102
	VLANIF20	10.5.1.1/24	0000-5e00-0103
DeviceE	管理网口	10.3.1.1/24	-
	VLANIF30	10.6.1.1/24	0000-5e00-0104

设备名称	接口	IP地址	虚拟MAC地址
	VLANIF40	10.7.2.1/24	-
	10GE1/0/29	10.7.1.1/24	-
	10GE1/0/30	10.8.1.1/24	-
	10GE1/0/31	10.8.2.1/24	-
DeviceF	管理网口	10.3.1.2/24	-
	VLANIF30	10.6.1.1/24	0000-5e00-0104
	VLANIF40	10.7.2.2/24	-
	10GE1/0/29	10.7.1.2/24	-
	10GE1/0/30	10.9.1.1/24	-
	10GE1/0/31	10.9.2.1/24	-
DeviceG	10GE1/0/1	10.8.1.2/24	-
	10GE1/0/2	10.9.2.2/24	-
DeviceH	10GE1/0/1	10.9.1.2/24	-
	10GE1/0/2	10.8.2.2/24	-
SeGWA	GigabitEthernet 3/0/0	10.10.0.1/24	-
	上行接口浮动IP	10.6.1.3/24	-
	下行接口浮动IP	10.5.1.3/24	-
SeGWB	GigabitEthernet 3/0/0	10.10.0.2/24	-
	上行接口浮动IP	10.6.1.3/24	-
	下行接口浮动IP	10.5.1.3/24	-
服务器所在网段	-	10.4.1.0/24	-

配置思路

采用如下的思路配置：

1. 在汇聚层和接入层交换机DeviceA和DeviceB之间、DeviceC和DeviceD之间、DeviceE和DeviceF之间配置M-LAG，实现双归接入，正常工作时链路进行负载分担且汇聚层任何一台设备故障对业务均没有影响，保证业务的高可靠性。
   - 在汇聚层配置交换机DeviceC和DeviceD为根桥设备，并在其下行接口上配置根保护功能，保证其接口能够正常转发流量。在接入层交换机配置DeviceA和DeviceB与用户终端相连的接口为边缘端口来加快网络拓扑的收敛时间，并配置BPDU保护功能来加强网络的稳定性。

- 在汇聚层交换机DeviceC和DeviceD、DeviceE和DeviceF上创建VLANIF接口并配置IP地址和MAC地址分别作为用户侧网关和防火墙的下一跳。

2. 配置安全网关设备采用路由模式（静态路由）接入，并启用双机热备功能，采用主备备份方式工作，增强网络的健壮性。
3. 在汇聚层和核心层交换机上使能OSPF。

操作步骤

步骤1 配置M-LAG。

1. 分别在DeviceA和DeviceB、DeviceC和DeviceD、DeviceE和DeviceF上配置V-STP、双主检测链路、DFS Group、peer-link和M-LAG成员接口。

   双主检测链路通过管理网口互通，DFS Group绑定的管理网口IP地址要保证可以相互通信，管理网口下绑定VPN实例，保证双主检测报文与业务流量隔离。

   Peer-Link链路Eth-Trunk接口的成员口建议跨板部署，避免单板单点故障导致Peer-link故障。

   # 配置DeviceA。

   配置接入层交换机连接服务器的Eth-Trunk接口配置成边缘端口，并配置BPDU保护功能。

   服务器上行连接交换机的端口需要绑定在一个聚合链路中且链路聚合模式需要和交换机侧的聚合模式匹配。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] stp mode rstp
[*DeviceA] stp v-stp enable
[*DeviceA] stp flush disable
[*DeviceA] ip vpn-instance VRF-A     //创建VRF-A
[*DeviceA-vpn-instance-VRF-A] ipv4-family
[*DeviceA-vpn-instance-VRF-A-af-ipv4] route-distinguisher 100:1
[*DeviceA-vpn-instance-VRF-A-af-ipv4] vpn-target 111:1 both
[*DeviceA-vpn-instance-VRF-A-af-ipv4] quit
[*DeviceA-vpn-instance-VRF-A] quit
[*DeviceA] interface meth 0/0/0
[*DeviceA-MEth0/0/0] ip binding vpn-instance VRF-A     //将管理网口绑定至VRF-A
[*DeviceA-MEth0/0/0] ip address 10.1.1.1 24
[*DeviceA-MEth0/0/0] quit
[*DeviceA] dfs-group 1
[*DeviceA-dfs-group-1] dual-active detection source ip 10.1.1.1 vpn-instance VRF-A peer ip
10.1.1.2    //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceA-dfs-group-1] priority 150
[*DeviceA-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*DeviceA-dfs-group-1] quit
[*DeviceA] interface eth-trunk 0
[*DeviceA-Eth-Trunk0] trunkport 10ge 1/0/4
[*DeviceA-Eth-Trunk0] trunkport 10ge 2/0/4     //Peer-Link链路Eth-Trunk接口的成员口跨板部署
[*DeviceA-Eth-Trunk0] mode lacp-static
[*DeviceA-Eth-Trunk0] peer-link 1
[*DeviceA-Eth-Trunk0] port vlan exclude 1
[*DeviceA-Eth-Trunk0] quit
[*DeviceA] vlan batch 11
[*DeviceA] interface eth-trunk 10
[*DeviceA-Eth-Trunk10] mode lacp-dynamic
[*DeviceA-Eth-Trunk10] port link-type access
[*DeviceA-Eth-Trunk10] port default vlan 11
[*DeviceA-Eth-Trunk10] trunkport 10ge 1/0/1
[*DeviceA-Eth-Trunk10] dfs-group 1 m-lag 1
[*DeviceA-Eth-Trunk10] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
```

```
[*DeviceA-Eth-Trunk10] quit
[*DeviceA] interface eth-trunk 20
[*DeviceA-Eth-Trunk20] mode lacp-dynamic
[*DeviceA-Eth-Trunk20] port link-type access
[*DeviceA-Eth-Trunk20] port default vlan 11
[*DeviceA-Eth-Trunk20] trunkport 10ge 1/0/2
[*DeviceA-Eth-Trunk20] dfs-group 1 m-lag 2
[*DeviceA-Eth-Trunk20] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceA-Eth-Trunk20] quit
[*DeviceA] interface eth-trunk 30
[*DeviceA-Eth-Trunk30] mode lacp-dynamic
[*DeviceA-Eth-Trunk30] port link-type access
[*DeviceA-Eth-Trunk30] port default vlan 11
[*DeviceA-Eth-Trunk30] trunkport 10ge 1/0/3
[*DeviceA-Eth-Trunk30] dfs-group 1 m-lag 3
[*DeviceA-Eth-Trunk30] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceA-Eth-Trunk30] quit
[*DeviceA] stp bpdu-protection     //使能设备边缘端口的BPDU保护功能
[*DeviceA] interface eth-trunk 40
[*DeviceA-Eth-Trunk40] mode lacp-static
[*DeviceA-Eth-Trunk40] port link-type trunk
[*DeviceA-Eth-Trunk40] undo port trunk allow-pass vlan 1
[*DeviceA-Eth-Trunk40] port trunk allow-pass vlan 11
[*DeviceA-Eth-Trunk40] trunkport 10ge 1/0/6 to 1/0/7
[*DeviceA-Eth-Trunk40] dfs-group 1 m-lag 4
[*DeviceA-Eth-Trunk40] quit
[*DeviceA] lacp m-lag priority 10
[*DeviceA] lacp m-lag system-id 00e0-fc00-0000
[*DeviceA] interface 10ge 1/0/9
[*DeviceA-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE1/0/9接口为例
[*DeviceA-10GE1/0/9] quit
[*DeviceA] commit
```

配置DeviceB。

配置接入层交换机连接服务器的Eth-Trunk接口配置成边缘端口，并配置BPDU保护功能。

服务器上行连接交换机的端口需要绑定在一个聚合链路中且链路聚合模式需要和交换机侧的聚合模式匹配。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*HUAWEI] commit
[~DeviceB] stp mode rstp
[*DeviceB] stp v-stp enable
[*DeviceB] stp flush disable
[*DeviceB] ip vpn-instance VRF-A     //创建VRF-A
[*DeviceB-vpn-instance-VRF-A] ipv4-family
[*DeviceB-vpn-instance-VRF-A-af-ipv4] route-distinguisher 100:2
[*DeviceB-vpn-instance-VRF-A-af-ipv4] vpn-target 111:1 both
[*DeviceB-vpn-instance-VRF-A-af-ipv4] quit
[*DeviceB-vpn-instance-VRF-A] quit
[*DeviceB] interface meth 0/0/0
[*DeviceB-MEth0/0/0] ip binding vpn-instance VRF-A     //将管理网口绑定至VRF-A
[*DeviceB-MEth0/0/0] ip address 10.1.1.2 24
[*DeviceB-MEth0/0/0] quit
[*DeviceB] dfs-group 1
[*DeviceB-dfs-group-1] dual-active detection source ip 10.1.1.2 vpn-instance VRF-A peer ip
10.1.1.1    //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceB-dfs-group-1] priority 120
[*DeviceB-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*DeviceB-dfs-group-1] quit
[*DeviceB] interface eth-trunk 0
[*DeviceB-Eth-Trunk0] trunkport 10ge 1/0/4
[*DeviceB-Eth-Trunk0] trunkport 10ge 2/0/4
[*DeviceB-Eth-Trunk0] mode lacp-static
[*DeviceB-Eth-Trunk0] peer-link 1
[*DeviceB-Eth-Trunk0] port vlan exclude 1
```

```
[*DeviceB-Eth-Trunk0] quit
[*DeviceB] vlan batch 11
[*DeviceB] interface eth-trunk 10
[*DeviceB-Eth-Trunk10] mode lacp-dynamic
[*DeviceB-Eth-Trunk10] port link-type access
[*DeviceB-Eth-Trunk10] port default vlan 11
[*DeviceB-Eth-Trunk10] trunkport 10ge 1/0/1
[*DeviceB-Eth-Trunk10] dfs-group 1 m-lag 1
[*DeviceB-Eth-Trunk10] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceB-Eth-Trunk10] quit
[*DeviceB] interface eth-trunk 20
[*DeviceB-Eth-Trunk20] mode lacp-dynamic
[*DeviceB-Eth-Trunk20] port link-type access
[*DeviceB-Eth-Trunk20] port default vlan 11
[*DeviceB-Eth-Trunk20] trunkport 10ge 1/0/2
[*DeviceB-Eth-Trunk20] dfs-group 1 m-lag 2
[*DeviceB-Eth-Trunk20] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceB-Eth-Trunk20] quit
[*DeviceB] interface eth-trunk 30
[*DeviceB-Eth-Trunk30] mode lacp-dynamic
[*DeviceB-Eth-Trunk30] port link-type access
[*DeviceB-Eth-Trunk30] port default vlan 11
[*DeviceB-Eth-Trunk30] trunkport 10ge 1/0/3
[*DeviceB-Eth-Trunk30] dfs-group 1 m-lag 3
[*DeviceB-Eth-Trunk30] stp edged-port enable     //配置该Eth-Trunk接口为边缘端口
[*DeviceB-Eth-Trunk30] quit
[*DeviceB] stp bpdu-protection     //使能设备边缘端口的BPDU保护功能
[*DeviceB] interface eth-trunk 40
[*DeviceB-Eth-Trunk40] mode lacp-static
[*DeviceB-Eth-Trunk40] port link-type trunk
[*DeviceB-Eth-Trunk40] undo port trunk allow-pass vlan 1
[*DeviceB-Eth-Trunk40] port trunk allow-pass vlan 11
[*DeviceB-Eth-Trunk40] trunkport 10ge 1/0/6 to 1/0/7
[*DeviceB-Eth-Trunk40] dfs-group 1 m-lag 4
[*DeviceB-Eth-Trunk40] quit
[*DeviceB] lacp m-lag priority 10
[*DeviceB] lacp m-lag system-id 00e0-fc00-0000
[*DeviceB] interface 10ge 1/0/9
[*DeviceB-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceB-10GE1/0/9] quit
[*DeviceB] commit
```

配置DeviceC。

配置汇聚层交换机DeviceC和DeviceD为STP网络的根桥设备，连接接入层交换机的Eth-Trunk接口配置根保护功能，保证其接口能够正常转发流量。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceC
[*HUAWEI] commit
[~DeviceC] stp mode rstp
[*DeviceC] stp root primary     //配置汇聚层设备为STP网络的根桥
[*DeviceC] stp bridge-address 00e0-fc00-1234     //配置根桥的桥MAC（DFS主设备的MAC地址）
[*DeviceC] stp v-stp enable
[*DeviceC] stp flush disable
[*DeviceC] ip vpn-instance VRF-B     //创建VRF-B
[*DeviceC-vpn-instance-VRF-B] ipv4-family
[*DeviceC-vpn-instance-VRF-B-af-ipv4] route-distinguisher 101:1
[*DeviceC-vpn-instance-VRF-B-af-ipv4] vpn-target 111:1 both
[*DeviceC-vpn-instance-VRF-B-af-ipv4] quit
[*DeviceC-vpn-instance-VRF-B] quit
[*DeviceC] interface meth 0/0/0
[*DeviceC-MEth0/0/0] ip binding vpn-instance VRF-B     //将管理网口绑定至VRF-B
[*DeviceC-MEth0/0/0] ip address 10.2.1.1 24
[*DeviceC-MEth0/0/0] quit
[*DeviceC] dfs-group 1
[*DeviceC-dfs-group-1] dual-active detection source ip 10.2.1.1 vpn-instance VRF-B peer ip
10.2.1.2     //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceC-dfs-group-1] priority 150
[*DeviceC-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
```

```
[*DeviceC-dfs-group-1] quit
[*DeviceC] interface eth-trunk 0
[*DeviceC-Eth-Trunk0] trunkport 10ge 1/0/3
[*DeviceC-Eth-Trunk0] trunkport 10ge 2/0/3     //Peer-Link链路Eth-Trunk接口的成员口跨板部署
[*DeviceC-Eth-Trunk0] mode lacp-static
[*DeviceC-Eth-Trunk0] peer-link 1
[*DeviceC-Eth-Trunk0] port vlan exclude 1
[*DeviceC-Eth-Trunk0] quit
[*DeviceC] vlan batch 11 20
[*DeviceC] interface eth-trunk 30
[*DeviceC-Eth-Trunk30] mode lacp-static
[*DeviceC-Eth-Trunk30] port link-type trunk
[*DeviceC-Eth-Trunk30] undo port trunk allow-pass vlan 1
[*DeviceC-Eth-Trunk30] port trunk allow-pass vlan 11
[*DeviceC-Eth-Trunk30] trunkport 10ge 1/0/1 to 1/0/2
[*DeviceC-Eth-Trunk30] dfs-group 1 m-lag 1
[*DeviceC-Eth-Trunk30] stp root-protection     //使能当前端口的根保护功能
[*DeviceC-Eth-Trunk30] quit
[*DeviceC] interface eth-trunk 40
[*DeviceC-Eth-Trunk40] mode lacp-static
[*DeviceC-Eth-Trunk40] port link-type trunk
[*DeviceC-Eth-Trunk40] undo port trunk allow-pass vlan 1
[*DeviceC-Eth-Trunk40] port trunk allow-pass vlan 20
[*DeviceC-Eth-Trunk40] trunkport 10ge 1/0/5
[*DeviceC-Eth-Trunk40] dfs-group 1 m-lag 2
[*DeviceC-Eth-Trunk40] quit
[*DeviceC] interface eth-trunk 50
[*DeviceC-Eth-Trunk50] mode lacp-static
[*DeviceC-Eth-Trunk50] port link-type trunk
[*DeviceC-Eth-Trunk50] undo port trunk allow-pass vlan 1
[*DeviceC-Eth-Trunk50] port trunk allow-pass vlan 20
[*DeviceC-Eth-Trunk50] trunkport 10ge 1/0/6
[*DeviceC-Eth-Trunk50] dfs-group 1 m-lag 3
[*DeviceC-Eth-Trunk50] quit
[*DeviceC] lacp m-lag priority 10
[*DeviceC] lacp m-lag system-id 00e0-fc00-0001
[*DeviceC] interface 10ge 1/0/9
[*DeviceC-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceC-10GE1/0/9] quit
[*DeviceC] commit
[*DeviceC] quit
```

配置DeviceD。

配置汇聚层交换机DeviceC和DeviceD为STP网络的根桥设备，连接接入层交换机的Eth-Trunk接口配置根保护功能，保证其接口能够正常转发流量。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceD
[*HUAWEI] commit
[~DeviceD] stp mode rstp
[*DeviceD] stp root primary     //配置汇聚层设备为STP网络的根桥
[*DeviceD] stp bridge-address 00e0-fc00-1234     //配置根桥的桥MAC
[*DeviceD] stp v-stp enable
[*DeviceD] stp flush disable
[*DeviceD] ip vpn-instance VRF-B     //创建VRF-B
[*DeviceD-vpn-instance-VRF-B] ipv4-family
[*DeviceD-vpn-instance-VRF-B-af-ipv4] route-distinguisher 101:2
[*DeviceD-vpn-instance-VRF-B-af-ipv4] vpn-target 111:1 both
[*DeviceD-vpn-instance-VRF-B-af-ipv4] quit
[*DeviceD-vpn-instance-VRF-B] quit
[*DeviceD] interface meth 0/0/0
[*DeviceD-MEth0/0/0] ip binding vpn-instance VRF-B     //将管理网口绑定至VRF-B
[*DeviceD-MEth0/0/0] ip address 10.2.1.2 24
[*DeviceD-MEth0/0/0] quit
[*DeviceD] dfs-group 1
[*DeviceD-dfs-group-1] dual-active detection source ip 10.2.1.2 vpn-instance VRF-B peer ip
10.2.1.1     //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceD-dfs-group-1] priority 120
[*DeviceD-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
```

```
[*DeviceD-dfs-group-1] quit
[*DeviceD] interface eth-trunk 0
[*DeviceD-Eth-Trunk0] trunkport 10ge 1/0/3
[*DeviceD-Eth-Trunk0] trunkport 10ge 2/0/3     //Peer-Link链路Eth-Trunk接口的成员口跨板部署
[*DeviceD-Eth-Trunk0] mode lacp-static
[*DeviceD-Eth-Trunk0] peer-link 1
[*DeviceD-Eth-Trunk0] port vlan exclude 1
[*DeviceD-Eth-Trunk0] quit
[*DeviceD] vlan batch 11 20
[*DeviceD] interface eth-trunk 30
[*DeviceD-Eth-Trunk30] mode lacp-static
[*DeviceD-Eth-Trunk30] port link-type trunk
[*DeviceD-Eth-Trunk30] undo port trunk allow-pass vlan 1
[*DeviceD-Eth-Trunk30] port trunk allow-pass vlan 11
[*DeviceD-Eth-Trunk30] trunkport 10ge 1/0/1 to 1/0/2
[*DeviceD-Eth-Trunk30] dfs-group 1 m-lag 1
[*DeviceD-Eth-Trunk30] stp root-protection    //使能当前端口的根保护功能
[*DeviceD-Eth-Trunk30] quit
[*DeviceD] interface eth-trunk 40
[*DeviceD-Eth-Trunk40] mode lacp-static
[*DeviceD-Eth-Trunk40] port link-type trunk
[*DeviceD-Eth-Trunk40] undo port trunk allow-pass vlan 1
[*DeviceD-Eth-Trunk40] port trunk allow-pass vlan 20
[*DeviceD-Eth-Trunk40] trunkport 10ge 1/0/5
[*DeviceD-Eth-Trunk40] dfs-group 1 m-lag 2
[*DeviceD-Eth-Trunk40] quit
[*DeviceD] interface eth-trunk 50
[*DeviceD-Eth-Trunk50] mode lacp-static
[*DeviceD-Eth-Trunk50] port link-type trunk
[*DeviceD-Eth-Trunk50] undo port trunk allow-pass vlan 1
[*DeviceD-Eth-Trunk50] port trunk allow-pass vlan 20
[*DeviceD-Eth-Trunk50] trunkport 10ge 1/0/6
[*DeviceD-Eth-Trunk50] dfs-group 1 m-lag 3
[*DeviceD-Eth-Trunk50] quit
[*DeviceD] lacp m-lag priority 10
[*DeviceD] lacp m-lag system-id 00e0-fc00-0001
[*DeviceD] interface 10ge 1/0/9
[*DeviceD-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceD-10GE1/0/9] quit
[*DeviceD] commit
[~DeviceD] quit
```

配置DeviceE。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceE
[*HUAWEI] commit
[~DeviceE] stp mode rstp
[*DeviceE] stp v-stp enable
[*DeviceE] stp flush disable
[*DeviceE] ip vpn-instance VRF-C     //创建VRF-C
[*DeviceE-vpn-instance-VRF-C] ipv4-family
[*DeviceE-vpn-instance-VRF-C-af-ipv4] route-distinguisher 102:1
[*DeviceE-vpn-instance-VRF-C-af-ipv4] vpn-target 111:1 both
[*DeviceE-vpn-instance-VRF-C-af-ipv4] quit
[*DeviceE-vpn-instance-VRF-C] quit
[*DeviceE] interface meth 0/0/0
[*DeviceE-MEth0/0/0] ip binding vpn-instance VRF-C     //将管理网口绑定至VRF-C
[*DeviceE-MEth0/0/0] ip address 10.3.1.1 24
[*DeviceE-MEth0/0/0] quit
[*DeviceE] dfs-group 1
[*DeviceE-dfs-group-1] dual-active detection source ip 10.3.1.1 vpn-instance VRF-C peer ip
10.3.1.2     //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceE-dfs-group-1] priority 150
[*DeviceE-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*DeviceE-dfs-group-1] quit
[*DeviceE] interface eth-trunk 0
[*DeviceE-Eth-Trunk0] trunkport 10ge 1/0/32
[*DeviceE-Eth-Trunk0] trunkport 10ge 2/0/32
[*DeviceE-Eth-Trunk0] mode lacp-static
```



```
[*DeviceE-Eth-Trunk0] peer-link 1
[*DeviceE-Eth-Trunk0] port vlan exclude 1
[*DeviceE-Eth-Trunk0] quit
[*DeviceE] vlan batch 30
[*DeviceE] interface eth-trunk 60
[*DeviceE-Eth-Trunk60] mode lacp-static
[*DeviceE-Eth-Trunk60] port link-type trunk
[*DeviceE-Eth-Trunk60] undo port trunk allow-pass vlan 1
[*DeviceE-Eth-Trunk60] port trunk allow-pass vlan 30
[*DeviceE-Eth-Trunk60] trunkport 10ge 1/0/34
[*DeviceE-Eth-Trunk60] dfs-group 1 m-lag 2
[*DeviceE-Eth-Trunk60] quit
[*DeviceE] interface eth-trunk 70
[*DeviceE-Eth-Trunk70] mode lacp-static
[*DeviceE-Eth-Trunk70] port link-type trunk
[*DeviceE-Eth-Trunk70] undo port trunk allow-pass vlan 1
[*DeviceE-Eth-Trunk70] port trunk allow-pass vlan 30
[*DeviceE-Eth-Trunk70] trunkport 10ge 1/0/35
[*DeviceE-Eth-Trunk70] dfs-group 1 m-lag 3
[*DeviceE-Eth-Trunk70] quit
[*DeviceE] lacp m-lag priority 10
[*DeviceE] lacp m-lag system-id 00e0-fc00-0002
[*DeviceE] interface 10ge 1/0/39
[*DeviceE-10GE1/0/39] shutdown     //关闭未使用的接口，此处以10GE 1/0/39接口为例
[*DeviceE-10GE1/0/39] quit
[*DeviceE] commit
[~DeviceE] quit
```

配置DeviceF。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceF
[*HUAWEI] commit
[~DeviceF] stp mode rstp
[*DeviceF] stp v-stp enable
[*DeviceF] stp flush disable
[*DeviceF] ip vpn-instance VRF-C     //创建VRF-C
[*DeviceF-vpn-instance-VRF-C] ipv4-family
[*DeviceF-vpn-instance-VRF-C-af-ipv4] route-distinguisher 102:2
[*DeviceF-vpn-instance-VRF-C-af-ipv4] vpn-target 111:1 both
[*DeviceF-vpn-instance-VRF-C-af-ipv4] quit
[*DeviceF-vpn-instance-VRF-C] quit
[*DeviceF] interface meth 0/0/0
[*DeviceF-MEth0/0/0] ip binding vpn-instance VRF-C     //将管理网口绑定至VRF-C
[*DeviceF-MEth0/0/0] ip address 10.3.1.2 24
[*DeviceF-MEth0/0/0] quit
[*DeviceF] dfs-group 1
[*DeviceF-dfs-group-1] dual-active detection source ip 10.3.1.2 vpn-instance VRF-C peer ip
10.3.1.1     //配置DFS Group绑定的IPv4地址和VPN实例
[*DeviceF-dfs-group-1] priority 120
[*DeviceF-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*DeviceF-dfs-group-1] quit
[*DeviceF] interface eth-trunk 0
[*DeviceF-Eth-Trunk0] trunkport 10ge 1/0/32
[*DeviceF-Eth-Trunk0] trunkport 10ge 2/0/32
[*DeviceF-Eth-Trunk0] mode lacp-static
[*DeviceF-Eth-Trunk0] peer-link 1
[*DeviceF-Eth-Trunk0] port vlan exclude 1
[*DeviceF-Eth-Trunk0] quit
[*DeviceF] vlan batch 30
[*DeviceF] interface eth-trunk 60
[*DeviceF-Eth-Trunk60] mode lacp-static
[*DeviceF-Eth-Trunk60] port link-type trunk
[*DeviceF-Eth-Trunk60] undo port trunk allow-pass vlan 1
[*DeviceF-Eth-Trunk60] port trunk allow-pass vlan 30
[*DeviceF-Eth-Trunk60] trunkport 10ge 1/0/34
[*DeviceF-Eth-Trunk60] dfs-group 1 m-lag 2
[*DeviceF-Eth-Trunk60] quit
[*DeviceF] interface eth-trunk 70
[*DeviceF-Eth-Trunk70] mode lacp-static
```



```
[*DeviceF-Eth-Trunk70] port link-type trunk
[*DeviceF-Eth-Trunk70] undo port trunk allow-pass vlan 1
[*DeviceF-Eth-Trunk70] port trunk allow-pass vlan 30
[*DeviceF-Eth-Trunk70] trunkport 10ge 1/0/35
[*DeviceF-Eth-Trunk70] dfs-group 1 m-lag 3
[*DeviceF-Eth-Trunk70] quit
[*DeviceF] commit
[*DeviceF] lacp m-lag priority 10
[*DeviceF] lacp m-lag system-id 00e0-fc00-0002
[*DeviceF] interface 10ge 1/0/39
[*DeviceF-10GE1/0/39] shutdown     //关闭未使用的接口，此处以10GE 1/0/39接口为例
[*DeviceF-10GE1/0/39] quit
[*DeviceF] commit
[~DeviceF] quit
```

2. 在DeviceC和DeviceD、DeviceE和DeviceF上创建VLANIF接口并配置IP地址，在接口VLANIF11上创建IP/MAC地址双活网关作为用户侧网关，在接口VLANIF20上创建IP/MAC地址双活网关作为防火墙下行的下一跳，在接口VLANIF30上创建IP/MAC地址双活网关作为防火墙上行的下一跳。同时在DeviceC和DeviceD上配置静态路由，上行下一跳指向防火墙，在DeviceE和DeviceF上配置静态路由，下行下一跳指向防火墙。

   # 配置DeviceC。

   ```
   <DeviceC> system-view
   [~DeviceC] interface vlanif 11
   [*DeviceC-Vlanif11] ip address 10.4.1.1 24
   [*DeviceC-Vlanif11] mac-address 0000-5e00-0102
   [*DeviceC-Vlanif11] quit
   [*DeviceC] interface vlanif 20
   [*DeviceC-Vlanif20] ip address 10.5.1.1 24
   [*DeviceC-Vlanif20] mac-address 0000-5e00-0103
   [*DeviceC-Vlanif20] quit
   [*DeviceC] ip route-static 0.0.0.0 0 10.5.1.3
   [*DeviceC] commit
   [~DeviceC] quit
   ```

   # 配置DeviceD。

   ```
   <DeviceD> system-view
   [~DeviceD] interface vlanif 11
   [*DeviceD-Vlanif11] ip address 10.4.1.1 24
   [*DeviceD-Vlanif11] mac-address 0000-5e00-0102
   [*DeviceD-Vlanif11] quit
   [*DeviceD] interface vlanif 20
   [*DeviceD-Vlanif20] ip address 10.5.1.1 24
   [*DeviceD-Vlanif20] mac-address 0000-5e00-0103
   [*DeviceD-Vlanif20] quit
   [*DeviceD] ip route-static 0.0.0.0 0 10.5.1.3
   [*DeviceD] commit
   [~DeviceD] quit
   ```

   # 配置DeviceE。

   ```
   <DeviceE> system-view
   [~DeviceE] interface vlanif 30
   [*DeviceE-Vlanif30] ip address 10.6.1.1 24
   [*DeviceE-Vlanif30] mac-address 0000-5e00-0104
   [*DeviceE-Vlanif30] quit
   [*DeviceE] ip route-static 10.4.1.0 24 10.6.1.3
   [*DeviceE] commit
   [~DeviceE] quit
   ```

   # 配置DeviceF。

   ```
   <DeviceF> system-view
   [~DeviceF] interface vlanif 30
   [*DeviceF-Vlanif30] ip address 10.6.1.1 24
   [*DeviceF-Vlanif30] mac-address 0000-5e00-0104
   ```



```
[*DeviceF-Vlanif30] quit
[*DeviceF] ip route-static 10.4.1.0 24 10.6.1.3
[*DeviceF] commit
[~DeviceF] quit
```

步骤2 配置SeGWA和SeGWB为路由模式双机热备份。

1. 配置SeGWA和SeGWB的上下行接口。

   # 配置SeGWA。

   ```
   <USG9000> system-view
   [USG9000] sysname SeGWA
   [SeGWA] interface eth-trunk 1
   [SeGWA-Eth-Trunk1] mode lacp-static
   [SeGWA-Eth-Trunk1] trunkport GigabitEthernet 1/0/0 to 1/0/1
   [SeGWA-Eth-Trunk1] ip address 10.5.1.3 24 float master
   [SeGWA-Eth-Trunk1] quit
   [SeGWA] interface eth-trunk 2
   [SeGWA-Eth-Trunk2] mode lacp-static
   [SeGWA-Eth-Trunk2] trunkport GigabitEthernet 2/0/0 to 2/0/1
   [SeGWA-Eth-Trunk2] ip address 10.6.1.3 24 float master
   [SeGWA-Eth-Trunk2] quit
   ```

   # 配置SeGWB。

   ```
   <USG9000> system-view
   [USG9000] sysname SeGWB
   [SeGWB] interface eth-trunk 1
   [SeGWB-Eth-Trunk1] mode lacp-static
   [SeGWB-Eth-Trunk1] trunkport GigabitEthernet 1/0/0 to 1/0/1
   [SeGWB-Eth-Trunk1] ip address 10.5.1.3 24 float slave
   [SeGWB-Eth-Trunk1] quit
   [SeGWB] interface eth-trunk 2
   [SeGWB-Eth-Trunk2] mode lacp-static
   [SeGWB-Eth-Trunk2] trunkport GigabitEthernet 2/0/0 to 2/0/1
   [SeGWB-Eth-Trunk2] ip address 10.6.1.3 24 float slave
   [SeGWB-Eth-Trunk2] quit
   ```

2. 配置SeGWA和SeGWB的心跳接口的IP地址。

   # 配置SeGWA。

   ```
   [SeGWA] interface GigabitEthernet 3/0/0
   [SeGWA-GigabitEthernet3/0/0] ip address 10.10.0.1 24
   [SeGWA-GigabitEthernet3/0/0] quit
   ```

   # 配置SeGWB。

   ```
   [SeGWB] interface GigabitEthernet 3/0/0
   [SeGWB-GigabitEthernet3/0/0] ip address 10.10.0.2 24
   [SeGWB-GigabitEthernet3/0/0] quit
   ```

3. 将SeGWA和SeGWB的上行业务接口加入untrust区域，下行业务接口加入trust区域，心跳口加入dmz区域。

   # 配置SeGWA。

   ```
   [SeGWA] firewall zone untrust
   [SeGWA-zone-untrust] add interface eth-trunk 2
   [SeGWA-zone-untrust] quit
   [SeGWA] firewall zone trust
   [SeGWA-zone-trust] add interface eth-trunk 1
   [SeGWA-zone-trust] quit
   [SeGWA] firewall zone dmz
   [SeGWA-zone-dmz] add interface GigabitEthernet 3/0/0
   [SeGWA-zone-dmz] quit
   ```

   # 配置SeGWB。

```
[SeGWB] firewall zone untrust
[SeGWB-zone-untrust] add interface eth-trunk 2
[SeGWB-zone-untrust] quit
[SeGWB] firewall zone trust
[SeGWB-zone-trust] add interface eth-trunk 1
[SeGWB-zone-trust] quit
[SeGWB] firewall zone dmz
[SeGWB-zone-dmz] add interface GigabitEthernet 3/0/0
[SeGWB-zone-dmz] quit
```

4. 指定心跳接口，启用双机热备。

   # 配置SeGWA。

   ```
   [SeGWA] hrp interface GigabitEthernet 3/0/0 remote 10.10.0.2
   [SeGWA] hrp enable
   ```

   # 配置SeGWB。

   ```
   [SeGWB] hrp interface GigabitEthernet 3/0/0 remote 10.10.0.1
   [SeGWB] hrp enable
   ```

5. 配置静态路由，分别指定防火墙上行流量的下一跳和下行流量的下一跳。

   # 配置SeGWA。

   ```
   [SeGWA] ip route-static 0.0.0.0 0 10.6.1.111
   [SeGWA] ip route-static 10.4.1.0 24 10.5.1.111
   ```

   # 配置SeGWB。

   ```
   [SeGWB] ip route-static 0.0.0.0 0 10.6.1.111
   [SeGWB] ip route-static 10.4.1.0 24 10.5.1.111
   ```

6. 双机热备功能配置完成后，需要在SeGW A上配置安全策略、IPS、攻击防范等安全功能。SeGWA的配置会自动备份到SeGWB。具体配置请参考安全网关设备的相关资料，这里不做具体介绍。

步骤3 在DeviceE、DeviceF、DeviceG和DeviceH上使能OSPF，采用主接口方式建立邻居。

配置DeviceE。

```
<DeviceE> system-view
[~DeviceE] interface 10ge 1/0/29
[*DeviceE-10GE1/0/29] undo portswitch
[*DeviceE-10GE1/0/29] ip address 10.7.1.1 24
[*DeviceE-10GE1/0/29] ospf enable 1 area 0
[*DeviceE-10GE1/0/29] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceE-10GE1/0/29] ospf cost 10     //将接口的OSPF协议的开销改为10
[*DeviceE-10GE1/0/29] quit
[*DeviceE] interface 10ge 1/0/30
[*DeviceE-10GE1/0/30] undo portswitch
[*DeviceE-10GE1/0/30] ip address 10.8.1.1 24
[*DeviceE-10GE1/0/30] ospf enable 1 area 0
[*DeviceE-10GE1/0/30] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceE -10GE1/0/30] quit
[*DeviceE] interface 10ge 1/0/31
[*DeviceE-10GE1/0/31] undo portswitch
[*DeviceE-10GE1/0/31] ip address 10.8.2.1 24
[*DeviceE-10GE1/0/31] ospf enable 1 area 0
[*DeviceE-10GE1/0/31] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceE-10GE1/0/31] quit
[*DeviceE] vlan batch 40
[*DeviceE] interface vlanif 40
[*DeviceE-Vlanif40] ip address 10.7.2.1 24
[*DeviceE-Vlanif40] ospf network-type p2p
[*DeviceE-Vlanif40] ospf cost 100
[*DeviceE-Vlanif40] quit
[*DeviceE] ospf 1
[*DeviceE-ospf-1] area 0
```



```
[*DeviceE-ospf-1-area-0.0.0.0] network 10.7.1.0 0.0.0.255
[*DeviceE-ospf-1-area-0.0.0.0] network 10.7.2.0 0.0.0.255
[*DeviceE-ospf-1-area-0.0.0.0] network 10.8.1.0 0.0.0.255
[*DeviceE-ospf-1-area-0.0.0.0] network 10.8.2.0 0.0.0.255
[*DeviceE-ospf-1-area-0.0.0.0] quit
[*DeviceE-ospf-1] import-route static
[*DeviceE-ospf-1] quit
[*DeviceE] commit
[~DeviceE] quit
```

配置DeviceF。

```
<DeviceF> system-view
[~DeviceF] interface 10ge 1/0/29
[*DeviceF-10GE1/0/29] undo portswitch
[*DeviceF-10GE1/0/29] ip address 10.7.1.2 24
[*DeviceF-10GE1/0/29] ospf enable 1 area 0
[*DeviceF-10GE1/0/29] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceF-10GE1/0/29] ospf cost 10     //将接口的OSPF协议的开销改为10
[*DeviceF-10GE1/0/29] quit
[*DeviceF] interface 10ge 1/0/30
[*DeviceF-10GE1/0/30] undo portswitch
[*DeviceF-10GE1/0/30] ip address 10.9.1.1 24
[*DeviceF-10GE1/0/30] ospf enable 1 area 0
[*DeviceF-10GE1/0/30] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceF-10GE1/0/30] quit
[*DeviceF] interface 10ge 1/0/31
[*DeviceF-10GE1/0/31] undo portswitch
[*DeviceF-10GE1/0/31] ip address 10.9.2.1 24
[*DeviceF-10GE1/0/31] ospf enable 1 area 0
[*DeviceF-10GE1/0/31] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceF-10GE1/0/31] quit
[*DeviceF] vlan batch 40
[*DeviceF] interface vlanif 40
[*DeviceF-Vlanif40] ip address 10.7.2.2 24
[*DeviceF-Vlanif40] ospf network-type p2p
[*DeviceF-Vlanif40] ospf cost 100
[*DeviceF-Vlanif40] quit
[*DeviceF] ospf 1
[*DeviceF-ospf-1] area 0
[*DeviceF-ospf-1-area-0.0.0.0] network 10.7.1.0 0.0.0.255
[*DeviceF-ospf-1-area-0.0.0.0] network 10.7.2.0 0.0.0.255
[*DeviceF-ospf-1-area-0.0.0.0] network 10.9.1.0 0.0.0.255
[*DeviceF-ospf-1-area-0.0.0.0] network 10.9.2.0 0.0.0.255
[*DeviceF-ospf-1-area-0.0.0.0] quit
[*DeviceF-ospf-1] import-route static
[*DeviceF-ospf-1] quit
[*DeviceF] commit
[~DeviceF] quit
```

配置DeviceG。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceG
[*HUAWEI] commit
[~DeviceG] interface 10ge 1/0/1
[~DeviceG-10GE1/0/1] undo portswitch
[*DeviceG-10GE1/0/1] ip address 10.8.1.2 24
[*DeviceG-10GE1/0/1] ospf enable 1 area 0
[*DeviceG-10GE1/0/1] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceG-10GE1/0/1] quit
[*DeviceG] interface 10ge 1/0/2
[*DeviceG-10GE1/0/2] undo portswitch
[*DeviceG-10GE1/0/2] ip address 10.9.2.2 24
[*DeviceG-10GE1/0/2] ospf enable 1 area 0
[*DeviceG-10GE1/0/2] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceG-10GE1/0/2] quit
[~DeviceG] interface 10ge 1/0/9
[~DeviceG-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceG-10GE1/0/9] quit
```



```
[*DeviceG] ospf 1
[*DeviceG-ospf-1] area 0
[*DeviceG-ospf-1-area-0.0.0.0] network 10.8.1.0 0.0.0.255
[*DeviceG-ospf-1-area-0.0.0.0] network 10.9.2.0 0.0.0.255
[*DeviceG-ospf-1-area-0.0.0.0] quit
[*DeviceG-ospf-1] quit
[*DeviceG] commit
```

配置DeviceH。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceH
[*HUAWEI] commit
[~DeviceH] interface 10ge 1/0/1
[~DeviceH-10GE1/0/1] undo portswitch
[*DeviceH-10GE1/0/1] ip address 10.9.1.2 24
[*DeviceH-10GE1/0/1] ospf enable 1 area 0
[*DeviceH-10GE1/0/1] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceH-10GE1/0/1] quit
[*DeviceH] interface 10ge 1/0/2
[*DeviceH-10GE1/0/2] undo portswitch
[*DeviceH-10GE1/0/2] ip address 10.8.2.2 24
[*DeviceH-10GE1/0/2] ospf enable 1 area 0
[*DeviceH-10GE1/0/2] ospf network-type p2p     //将接口的网络类型更改为点到点
[*DeviceH-10GE1/0/2] quit
[~DeviceH] interface 10ge 1/0/9
[~DeviceH-10GE1/0/9] shutdown     //关闭未使用的接口，此处以10GE 1/0/9接口为例
[*DeviceH-10GE1/0/9] quit
[*DeviceH] ospf 1
[*DeviceH-ospf-1] area 0
[*DeviceH-ospf-1-area-0.0.0.0] network 10.8.2.0 0.0.0.255
[*DeviceH-ospf-1-area-0.0.0.0] network 10.9.1.0 0.0.0.255
[*DeviceH-ospf-1-area-0.0.0.0] quit
[*DeviceH-ospf-1] quit
[*DeviceH] commit
```

----结束

检查配置结果

1. 执行命令**display dfs-group**，查看M-LAG的相关信息。

   # 查看DFS Group编号为1的M-LAG信息。（这里以DeviceA和DeviceB组成的M-LAG为例，DeviceC和DeviceD、DeviceE和DeviceF类似。）

   ```
   [~DeviceA] display dfs-group 1 m-lag
   *                : Local node
   Heart beat state     : OK
   Node 1 *
     Dfs-Group ID       : 1
     Priority           : 150
     Dual-active Address : 10.1.1.1
     VPN-Instance       :  public net
     State              : Master
     Causation          : -
     System ID          : 00e0-fc00-1234
     SysName            : DeviceA
     Version            : V300R023C00
     Device Type        :  CE16800
   Node 2
     Dfs-Group ID       : 1
     Priority           : 120
     Dual-active Address : 10.1.1.2
     VPN-Instance       :  public net
     State              : Backup
     Causation          : -
     System ID          : 00e0-fc00-1235
     SysName            : DeviceB
     Version            : V300R023C00
     Device Type        : CE16800
   ```


查看DeviceA上的M-LAG信息。

```
[~DeviceA] display dfs-group 1 node 1 m-lag brief
* - Local node

M-Lag ID    Interface      Port State   Status            Consistency-check
    1       Eth-Trunk 10   Up           active(*)-active  --
    2       Eth-Trunk 20   Up           active(*)-active  --
    3       Eth-Trunk 30   Up           active(*)-active  --
    4       Eth-Trunk 40   Up           active(*)-active  --
```

通过以上显示信息可以看到，“Heart beat state”的状态是“OK”，表明双主检测状态正常；DeviceA作为Node 1，优先级为150，“State”的状态是“Master”；DeviceB作为Node 2，优先级为120，“State”的状态是“Backup”。同时“Causation”的状态是“-”，Node 1的“Port State”状态为“Up”，Node 2的“Port State”状态为“Up”，且Node 1和Node 2的M-LAG状态均为“active”，表明M-LAG的配置正确。

2. 在SeGWA上执行**display hrp state**命令，检查当前HRP的状态，显示以下信息表示HRP建立成功。

```
HRP_M[SeGWA] display hrp state
 Role: active, peer: standby
 Running priority: 51008, peer: 51008
 Core state: normal, peer: normal
 Backup channel usage: 0%
 Stable time: 0 days, 18 hours, 41 minutes
```

配置脚本

- DeviceA的配置脚本

```
#
sysname DeviceA
#
dfs-group 1
 priority 150
 dual-active detection source ip 10.1.1.1 vpn-instance VRF-A peer 10.1.1.2
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 11
#
stp mode rstp
stp v-stp enable
stp bpdu-protection
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0000
lacp m-lag priority 10
#
ip vpn-instance VRF-A
 ipv4-family
  route-distinguisher 100:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface MEth0/0/0
 ip binding vpn-instance VRF-A
 ip address 10.1.1.1 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 port default vlan 11
 stp edged-port enable
```

```
 mode lacp-dynamic
 dfs-group 1 m-lag 1
#
interface Eth-Trunk20
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 2
#
interface Eth-Trunk30
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 3
#
interface Eth-Trunk40
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
 mode lacp-static
 dfs-group 1 m-lag 4
#
interface 10GE1/0/1
 eth-trunk 10
#
interface 10GE1/0/2
 eth-trunk 20
#
interface 10GE1/0/3
 eth-trunk 30
#
interface 10GE1/0/4
 eth-trunk 0
#
interface 10GE1/0/6
 eth-trunk 40
#
interface 10GE1/0/7
 eth-trunk 40
#
interface 10GE1/0/9
 shutdown
#
interface 10GE2/0/4
 eth-trunk 0
#
return
```

- DeviceB的配置脚本

```
#
sysname DeviceB
#
dfs-group 1
 priority 120
 dual-active detection source ip 10.1.1.2 vpn-instance VRF-A peer 10.1.1.1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 11
#
stp mode rstp
stp v-stp enable
stp bpdu-protection
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0000
lacp m-lag priority 10
#
ip vpn-instance VRF-A
 ipv4-family
```



```
 route-distinguisher 100:2
 vpn-target 111:1 export-extcommunity
 vpn-target 111:1 import-extcommunity
#
interface MEth0/0/0
 ip binding vpn-instance VRF-A
 ip address 10.1.1.2 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk10
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 1
#
interface Eth-Trunk20
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 2
#
interface Eth-Trunk30
 port default vlan 11
 stp edged-port enable
 mode lacp-dynamic
 dfs-group 1 m-lag 3
#
interface Eth-Trunk40
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
 mode lacp-static
 dfs-group 1 m-lag 4
#
interface 10GE1/0/1
 eth-trunk 10
#
interface 10GE1/0/2
 eth-trunk 20
#
interface 10GE1/0/3
 eth-trunk 30
#
interface 10GE1/0/4
 eth-trunk 0
#
interface 10GE1/0/6
 eth-trunk 40
#
interface 10GE1/0/7
 eth-trunk 40
#
interface 10GE1/0/9
 shutdown
#
interface 10GE2/0/4
 eth-trunk 0
#
return
```

- DeviceC的配置脚本

```
#
sysname DeviceC
#
dfs-group 1
 priority 150
```



```
 dual-active detection source ip 10.2.1.1 vpn-instance VRF-B peer 10.2.1.2
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 11 20
#
stp bridge-address 00e0-fc00-1234
stp mode rstp
stp v-stp enable
stp instance 0 root primary
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0001
lacp m-lag priority 10
#
ip vpn-instance VRF-B
 ipv4-family
  route-distinguisher 101:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface Vlanif11
 ip address 10.4.1.1 255.255.255.0
 mac-address 0000-5e00-0102
#
interface Vlanif20
 ip address 10.5.1.1 255.255.255.0
 mac-address 0000-5e00-0103
#
interface MEth0/0/0
 ip binding vpn-instance VRF-B
 ip address 10.2.1.1 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk30
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
 stp root-protection
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk40
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 20
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk50
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 20
 mode lacp-static
 dfs-group 1 m-lag 3
#
interface 10GE1/0/1
 eth-trunk 30
#
interface 10GE1/0/2
 eth-trunk 30
#
interface 10GE1/0/3
 eth-trunk 0
#
```

```
interface 10GE1/0/5
 eth-trunk 40
#
interface 10GE1/0/6
 eth-trunk 50
#
interface 10GE1/0/9
 shutdown
#
interface 10GE2/0/3
 eth-trunk 0
#
ip route-static 0.0.0.0 0.0.0.0 10.5.1.3
#
return
```

- DeviceE的配置脚本

```
#
sysname DeviceE
#
dfs-group 1
 priority 150
 dual-active detection source ip 10.3.1.1 vpn-instance VRF-C peer 10.3.1.2
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 30 40
#
stp mode rstp
stp v-stp enable
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0002
lacp m-lag priority 10
#
ip vpn-instance VRF-C
 ipv4-family
  route-distinguisher 102:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface Vlanif30
 ip address 10.6.1.1 255.255.255.0
 mac-address 0000-5e00-0104
#
interface Vlanif40
 ip address 10.7.2.1 255.255.255.0
 ospf cost 100
 ospf network-type p2p
#
interface MEth0/0/0
 ip binding vpn-instance VRF-B
 ip address 10.3.1.1 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk60
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 30
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk70
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 30
```



```
 mode lacp-static
 dfs-group 1 m-lag 3
#
interface 10GE1/0/29
 undo portswitch
 ip address 10.7.1.1 255.255.255.0
 ospf cost 10
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
interface 10GE1/0/30
 undo portswitch
 ip address 10.8.1.1 255.255.255.0
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
interface 10GE1/0/31
 undo portswitch
 ip address 10.8.2.1 255.255.255.0
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
interface 10GE1/0/32
 eth-trunk 0
#
interface 10GE1/0/34
 eth-trunk 60
#
interface 10GE1/0/35
 eth-trunk 70
#
interface 10GE1/0/39
 shutdown
#
interface 10GE2/0/32
 eth-trunk 0
#
ip route-static 10.4.1.0 255.255.255.0 10.6.1.3
#
ospf 1
 import-route static
 area 0.0.0.0
  network 10.7.1.0 0.0.0.255
  network 10.7.2.0 0.0.0.255
  network 10.8.1.0 0.0.0.255
  network 10.8.2.0 0.0.0.255
#
return
```

- DeviceD的配置脚本

```
#
sysname DeviceD
#
dfs-group 1
 priority 120
 dual-active detection source ip 10.2.1.2 vpn-instance VRF-B peer 10.2.1.1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 11 20
#
stp bridge-address 00e0-fc00-1234
stp mode rstp
stp v-stp enable
stp instance 0 root primary
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0001
lacp m-lag priority 10
#
```

```
ip vpn-instance VRF-B
 ipv4-family
  route-distinguisher 101:2
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface Vlanif11
 ip address 10.4.1.1 255.255.255.0
 mac-address 0000-5e00-0102
#
interface Vlanif20
 ip address 10.5.1.1 255.255.255.0
 mac-address 0000-5e00-0103
#
interface MEth0/0/0
 ip binding vpn-instance VRF-B
 ip address 10.2.1.2 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk30
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 11
 stp root-protection
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk40
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 20
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk50
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 20
 mode lacp-static
 dfs-group 1 m-lag 3
#
interface 10GE1/0/1
 eth-trunk 30
#
interface 10GE1/0/2
 eth-trunk 30
#
interface 10GE1/0/3
 eth-trunk 0
#
interface 10GE1/0/5
 eth-trunk 40
#
interface 10GE1/0/6
 eth-trunk 50
#
interface 10GE1/0/9
 shutdown
#
interface 10GE2/0/3
 eth-trunk 0
#
ip route-static 0.0.0.0 0.0.0.0 10.5.1.3
#
return
```

- DeviceF的配置脚本

```
#
sysname DeviceF
#
dfs-group 1
 priority 120
 dual-active detection source ip 10.3.1.2 vpn-instance VRF-A peer 10.3.1.1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 30 40
#
stp mode rstp
stp v-stp enable
stp flush disable
#
lacp m-lag system-id 00e0-fc00-0002
lacp m-lag priority 10
#
ip vpn-instance VRF-C
 ipv4-family
  route-distinguisher 102:2
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface Vlanif30
 ip address 10.6.1.1 255.255.255.0
 mac-address 0000-5e00-0104
#
interface Vlanif40
 ip address 10.7.2.2 255.255.255.0
 ospf cost 100
 ospf network-type p2p
#
interface MEth0/0/0
 ip binding vpn-instance VRF-B
 ip address 10.3.1.2 255.255.255.0
#
interface Eth-Trunk0
 mode lacp-static
 peer-link 1
 port vlan exclude 1
#
interface Eth-Trunk60
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 30
 mode lacp-static
 dfs-group 1 m-lag 2
#
interface Eth-Trunk70
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 30
 mode lacp-static
 dfs-group 1 m-lag 3
#
interface 10GE1/0/29
 undo portswitch
 ip address 10.7.1.2 255.255.255.0
 ospf cost 10
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
interface 10GE1/0/30
 undo portswitch
 ip address 10.9.1.1 255.255.255.0
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
```



```
#
interface 10GE1/0/31
 undo portswitch
 ip address 10.9.2.1 255.255.255.0
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
interface 10GE1/0/32
 eth-trunk 0
#
interface 10GE1/0/34
 eth-trunk 60
#
interface 10GE1/0/35
 eth-trunk 70
#
interface 10GE1/0/39
 shutdown
#
interface 10GE2/0/32
 eth-trunk 0
#
ip route-static 10.4.1.0 255.255.255.0 10.6.1.3
#
ospf 1
 import-route static
 area 0.0.0.0
  network 10.7.1.0 0.0.0.255
  network 10.7.2.0 0.0.0.255
  network 10.9.1.0 0.0.0.255
  network 10.9.2.0 0.0.0.255
#
return
```

- DeviceG的配置脚本

```
#
sysname DeviceG
#
interface 10GE1/0/1
 undo portswitch
 ip address 10.8.1.2 255.255.255.0
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
interface 10GE1/0/2
 undo portswitch
 ip address 10.9.2.2 255.255.255.0
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
interface 10GE1/0/9
 shutdown
#
ospf 1
 area 0.0.0.0
  network 10.8.1.0 0.0.0.255
  network 10.9.2.0 0.0.0.255
#
return
```

- DeviceH的配置脚本

```
#
sysname DeviceH
#
interface 10GE1/0/1
 undo portswitch
 ip address 10.9.1.2 255.255.255.0
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
```



```
interface 10GE1/0/2
 undo portswitch
 ip address 10.8.2.2 255.255.255.0
 ospf network-type p2p
 ospf enable 1 area 0.0.0.0
#
interface 10GE1/0/9
 shutdown
#
ospf 1
 area 0.0.0.0
  network 10.8.2.0 0.0.0.255
  network 10.9.1.0 0.0.0.255
#
return
```

- SeGWA的配置脚本

```
#
sysname SeGWA
#
 hrp enable
 hrp interface GigabitEthernet 3/0/0 remote 10.10.0.2
#
interface Eth-Trunk1
 mode lacp-static
 trunkport GigabitEthernet 1/0/0 to 1/0/1
 ip address 10.5.1.3 24 float master
#
interface Eth-Trunk2
 mode lacp-static
 trunkport GigabitEthernet 2/0/0 to 2/0/1
 ip address 10.6.1.3 24 float master
#
interface GigabitEthernet3/0/0
 ip address 10.10.0.1 24
#
firewall zone trust
 set priority 85
 add interface eth-trunk 1
#
firewall zone dmz
 set priority 50
 add interface GigabitEthernet 3/0/0
#
firewall zone untrust
 set priority 5
 add interface eth-trunk 2
#
 ip route-static 0.0.0.0 0 10.6.1.111
 ip route-static 10.4.1.0 24 10.5.1.111
#
return
```

- SeGWB的配置脚本

```
#
sysname SeGWB
#
 hrp enable
 hrp interface GigabitEthernet 3/0/0 remote 10.10.0.1
#
interface Eth-Trunk1
 mode lacp-static
 trunkport GigabitEthernet 1/0/0 to 1/0/1
 ip address 10.5.1.3 24 float slave
#
interface Eth-Trunk2
 mode lacp-static
 trunkport GigabitEthernet 2/0/0 to 2/0/1
 ip address 10.6.1.3 24 float slave
#
```



```
interface GigabitEthernet3/0/0
 ip address 10.10.0.2 24
#
firewall zone trust
 set priority 85
 add interface eth-trunk 1
#
firewall zone dmz
 set priority 50
 add interface GigabitEthernet 3/0/0
#
firewall zone untrust
 set priority 5
 add interface eth-trunk 2
#
 ip route-static 0.0.0.0 0 10.6.1.111
 ip route-static 10.4.1.0 24 10.5.1.111
#
return
```

1.12 配置动态路由接入 M-LAG 示例

适用产品和版本

CE16800（除X系列单板外）、CE8800、CE6800系列产品V300R020C00或更高版本。

如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

如**图1-12**所示，DeviceB和DeviceC组成M-LAG系统，DeviceB和DeviceC的M-LAG端口支持动态路由协议，用户在DeviceA上配置动态路由通过三层路由方式接入到M-LAG系统。

图 1-12 配置动态路由接入 M-LAG 组网图

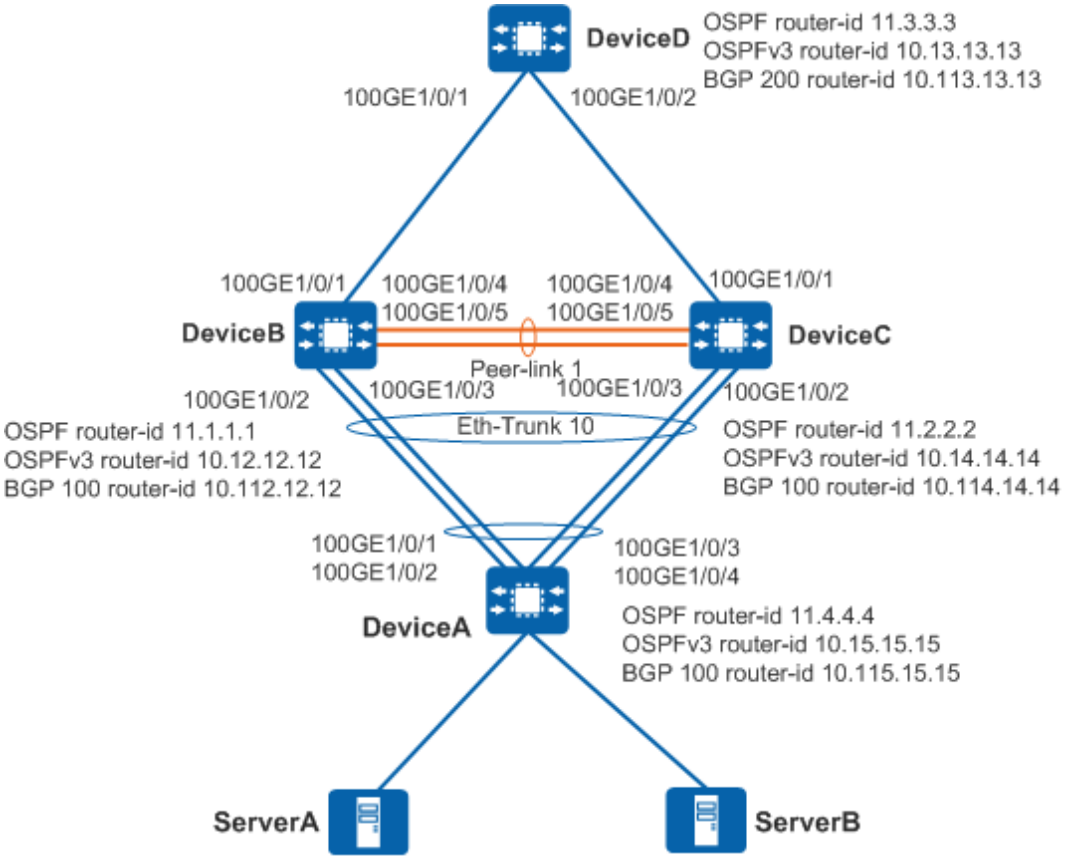


表 1-13 数据准备表

设备名称	接口编号	VLAN及IP地址	对接设备及接口编号
DeviceA	Eth-Trunk 10 • 100GE1/0/1 • 100GE1/0/2 • 100GE1/0/3 • 100GE1/0/4	VLAN 100：IPv4地址 10.100.0.2/24 VLAN 101：IPv6地址 2001:DB8:101::2/64 VLAN 102：IPv4地址 10.102.0.2/24 VLAN 103：IPv6地址 2001:DB8:103::2/64	DeviceB：Eth-Trunk 10 • 100GE1/0/2 • 100GE1/0/3
			DeviceC：Eth-Trunk 10 • 100GE1/0/2 • 100GE1/0/3
DeviceB	管理网口	10.200.1.1/24	-
	100GE1/0/1	IP地址：192.168.1.1/24 IPv6地址：2001:DB8:1::1/64	DeviceD：100GE1/0/1

设备名称	接口编号	VLAN及IP地址	对接设备及接口编号
	Eth-Trunk 1 ● 100GE1/0/4 ● 100GE1/0/5	-	DeviceC：Eth-Trunk 1 ● 100GE1/0/4 ● 100GE1/0/5
	Eth-Trunk 10 ● 100GE1/0/2 ● 100GE1/0/3	VLAN 100：IPv4地址 10.100.0.1/24 M-LAG IPv4地址 10.100.0.3/24 VLAN 101：IPv6地址 2001:DB8:101::1/64 M-LAG Link-local地址 FE80:1000::1 VLAN 102：IPv4地址 10.102.0.1/24 M-LAG IPv4地址 10.102.0.3/24 VLAN 103：IPv6地址 2001:DB8:103::1/64 M-LAG IPv6地址 2001:DB8:103::3/64	DeviceA：Eth-Trunk 10 ● 100GE1/0/1 ● 100GE1/0/2
	LoopBack 1	10.2.2.2/32	-
	LoopBack 2	10.3.3.3/32	-
DeviceC	管理网口	10.200.2.1/24	-
	100GE1/0/1	192.168.2.1/24 2001:DB8:2::1/64	DeviceD：100GE1/0/2
	Eth-Trunk 1 ● 100GE1/0/4 ● 100GE1/0/5	-	DeviceB：Eth-Trunk 1 ● 100GE1/0/4 ● 100GE1/0/5

设备名称	接口编号	VLAN及IP地址	对接设备及接口编号
	Eth-Trunk 10 • 100GE1/0/2 • 100GE1/0/3	VLAN 100： IPv4地址 10.100.0.1/24 M-LAG IPv4地址 10.100.0.4/24 VLAN 101： IPv6地址 2001:DB8:101::1/64 M-LAG Link-local地址 FE80:1000::2 VLAN 102： IPv4地址 10.102.0.1/24 M-LAG IPv4地址 10.102.0.4/24 VLAN 103： IPv6地址 2001:DB8:103::1/64 M-LAG IPv6地址 2001:DB8:103::4/64	DeviceA：Eth-Trunk 10 • 100GE1/0/3 • 100GE1/0/4
	LoopBack 1	10.2.2.2/32	-
	LoopBack 2	10.4.4.4/32	-
DeviceD	100GE1/0/1	192.168.1.2/24 2001:DB8:1::2/64	DeviceB：100GE1/0/1
	100GE1/0/2	192.168.2.2/24 2001:DB8:2::2/64	DeviceC：100GE1/0/1
	LoopBack 1	10.1.1.1/32	-

配置思路

采用如下的思路配置：

说明

- M-LAG侧和网络侧需要进行拓扑隔离，可以按照如下场景配置：
  - 网络侧和M-LAG侧部署不同动态路由协议。
  - 网络侧和M-LAG侧同时部署OSPF/OSPFv3协议，但配置不同OSPF/OSPFv3进程。
  - 网络侧与M-LAG侧同时部署OSPF/OSPFv3协议且属于同一个OSPF/OSPFv3进程，但配置不同的OSPF/OSPFv3区域。
- M-LAG成员设备和DeviceA之间建议配置BFD for M-LAG功能，用于快速检测链路故障。
- 如果配置OSPF/OSPFv3协议时指定了VPN实例，建议在OSPF/OSPFv3视图下执行**vpn-instance-capability simple**命令，禁止路由环路检测，直接进行路由计算。

本示例配置OSPF/OSPFv3协议时，采用网络侧与M-LAG侧同时部署OSPF/OSPFv3协议且属于同一个OSPF/OSPFv3进程，但配置不同的OSPF/OSPFv3区域方式实现。

1. 配置DeviceA、DeviceB、DeviceC建立M-LAG系统。
   - 配置DeviceB和DeviceC的V-STP。
   - 配置DeviceB和DeviceC的DFS Group并绑定管理网口的IP地址。
   - 配置DeviceB和DeviceC之间的peer-link链路。
   - 配置DeviceB和DeviceC的M-LAG成员接口，DeviceA的Eth-Trunk接口。
   - 配置DeviceB和DeviceC的Monitor Link功能。

     说明

     Monitor Link将上行接口和下行接口关联，避免因上行链路故障导致用户侧流量无法转发而丢弃。
2. 配置OSPF接入M-LAG。
   - 配置DeviceB、DeviceC、DeviceD的路由协议OSPF，实现网络三层互通。
   - 配置DeviceB和DeviceC的动态路由OSPF接入M-LAG的IPv4地址。

     说明

     DeviceB和DeviceC上需要配置不一样的M-LAG IPv4地址，否则会导致路由协议邻居建立不起来。
   - 配置DeviceA和M-LAG成员设备（DeviceB、DeviceC）建立动态路由OSPF的IP地址。
3. 配置OSPFv3接入M-LAG。
   - 配置DeviceB、DeviceC、DeviceD的路由协议OSPFv3，实现网络三层互通。
   - 配置DeviceB和DeviceC的动态路由OSPFv3接入M-LAG的Link-local地址。

     说明

     DeviceB和DeviceC上需要配置不一样的M-LAG Link-local地址，否则会导致路由协议邻居建立不起来。
   - 配置DeviceA和M-LAG成员设备（DeviceB、DeviceC）建立动态路由OSPFv3的IPv6地址。
4. 配置BGP接入M-LAG。
   - 配置DeviceB、DeviceC、DeviceD的路由协议BGP，实现网络三层互通。
   - 配置DeviceB和DeviceC的动态路由BGP接入M-LAG的IPv4地址。

     说明

     DeviceB和DeviceC上需要配置不一样的M-LAG IPv4地址，否则会导致路由协议邻居建立不起来。

- 配置DeviceA和M-LAG成员设备（DeviceB、DeviceC）建立动态路由BGP的对等体。

5. 配置BGP4+接入M-LAG。
   - 配置DeviceB、DeviceC、DeviceD的路由协议BGP4+，实现网络三层互通。
   - 配置DeviceB和DeviceC的动态路由BGP4+接入M-LAG的IPv6地址。

     说明

     DeviceB和DeviceC上需要配置不一样的M-LAG IPv6地址，否则会导致路由协议邻居建立不起来。
   - 配置DeviceA和M-LAG成员设备（DeviceB、DeviceC）建立动态路由BGP4+的对等体和引入路由。

操作步骤

步骤1 配置DeviceA、DeviceB、DeviceC建立M-LAG系统。

1. 配置DeviceB和DeviceC的V-STP。

   # 配置DeviceB。DeviceC的配置与DeviceB类似，这里不再赘述。

   ```
   <HUAWEI> system-view
   [~HUAWEI] sysname DeviceB
   [*HUAWEI] commit
   [~DeviceB] stp mode rstp
   [*DeviceB] stp v-stp enable
   [*DeviceB] commit
   ```

2. 配置DeviceB和DeviceC的DFS Group并绑定管理网口的IP地址。

   DeviceB和DeviceC管理网口需要保证能够三层互通。

   # 配置DeviceB。DeviceC的配置与DeviceB类似，这里不再赘述。

   ```
   [~DeviceB] dfs-group 1
   [*DeviceB-dfs-group-1] dual-active detection source ip 10.200.1.1 peer 10.200.2.1
   [*DeviceB-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
   [*DeviceB-dfs-group-1] quit
   [*DeviceB] commit
   ```

3. 配置DeviceB和DeviceC之间的peer-link链路。

   # 配置DeviceB。DeviceC的配置与DeviceB类似，这里不再赘述。

   ```
   [~DeviceB] interface eth-trunk 1
   [*DeviceB-Eth-Trunk1] mode lacp-static
   [*DeviceB-Eth-Trunk1] trunkport 100ge 1/0/4
   [*DeviceB-Eth-Trunk1] trunkport 100ge 1/0/5
   [*DeviceB-Eth-Trunk1] peer-link 1
   [*DeviceB-Eth-Trunk1] quit
   [*DeviceB] commit
   ```

4. 配置DeviceB和DeviceC的M-LAG成员接口，DeviceA的Eth-Trunk接口。

   # 配置DeviceB。DeviceC的配置与DeviceB类似，这里不再赘述。

   ```
   [~DeviceB] interface eth-trunk 10
   [*DeviceB-Eth-Trunk10] mode lacp-static
   [*DeviceB-Eth-Trunk10] port link-type trunk
   [*DeviceB-Eth-Trunk10] lacp mixed-rate link enable
   [*DeviceB-Eth-Trunk10] trunkport 100ge 1/0/2
   [*DeviceB-Eth-Trunk10] trunkport 100ge 1/0/3
   [*DeviceB-Eth-Trunk10] dfs-group 1 m-lag 1
   [*DeviceB-Eth-Trunk10] quit
   [*DeviceB] commit
   ```


配置DeviceA。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] interface eth-trunk 10
[*DeviceA-Eth-Trunk10] mode lacp-static
[*DeviceA-Eth-Trunk10] port link-type trunk
[*DeviceA-Eth-Trunk10] trunkport 100ge 1/0/1 to 1/0/4
[*DeviceA-Eth-Trunk10] quit
[*DeviceA] commit
```

5. 配置DeviceB和DeviceC的Monitor Link功能。

   # 配置DeviceB。DeviceC的配置与DeviceB类似，这里不再赘述。

   ```
   [~DeviceB] monitor-link group 1
   [*DeviceB-mtlk-group1] port 100ge 1/0/1 uplink
   [*DeviceB-mtlk-group1] port eth-trunk 10 downlink 1
   [*DeviceB-mtlk-group1] quit
   [*DeviceB] commit
   ```

步骤2 配置OSPF接入M-LAG。

1. 配置DeviceB、DeviceC、DeviceD的路由协议OSPF，实现网络三层互通。

   # 配置DeviceB。DeviceC和DeviceD的配置与DeviceB类似，这里不再赘述。配置OSPF时，注意需要发布32位Loopback接口地址。

   ```
   [~DeviceB] interface loopback 1
   [*DeviceB-LoopBack1] ip address 10.2.2.2 32
   [*DeviceB-LoopBack1] quit
   [*DeviceB] interface loopback 2
   [*DeviceB-LoopBack2] ip address 10.3.3.3 32
   [*DeviceB-LoopBack2] quit
   [*DeviceB] interface 100ge 1/0/1
   [*DeviceB-100GE1/0/1] undo portswitch
   [*DeviceB-100GE1/0/1] ip address 192.168.1.1 24
   [*DeviceB-100GE1/0/1] quit
   [*DeviceB] ospf 1 router-id 11.1.1.1
   [*DeviceB-ospf-1] area 0
   [*DeviceB-ospf-1-area-0.0.0.0] network 10.2.2.2 0.0.0.0
   [*DeviceB-ospf-1-area-0.0.0.0] network 10.3.3.3 0.0.0.0
   [*DeviceB-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
   [*DeviceB-ospf-1-area-0.0.0.0] quit
   [*DeviceB-ospf-1] quit
   [*DeviceB] commit
   ```

   OSPF成功配置后，DeviceB、DeviceC、DeviceD之间可通过OSPF协议发现对方的Loopback接口的IP地址，并能互相ping通。

2. 配置DeviceB和DeviceC的动态路由OSPF接入M-LAG的IPv4地址。

   # 配置DeviceB。

   ```
   [~DeviceB] vlan 100
   [*DeviceB-vlan100] quit
   [*DeviceB] interface vlanif 100
   [*DeviceB-Vlanif100] ip address 10.100.0.1 255.255.255.0
   [*DeviceB-Vlanif100] m-lag ip address 10.100.0.3 255.255.255.0
   [*DeviceB-Vlanif100] ospf source sub-address 10.100.0.3
   [*DeviceB-Vlanif100] mac-address 0000-5e00-0101
   [*DeviceB-Vlanif100] arp proxy enable
   [*DeviceB-Vlanif100] quit
   [*DeviceB] commit
   [~DeviceB] interface eth-trunk 10
   [*DeviceB-Eth-Trunk10] port trunk allow-pass vlan 100
   [*DeviceB-Eth-Trunk10] quit
   [*DeviceB] commit
   [~DeviceB] ospf
   ```

```
[~DeviceB-ospf-1] area 1
[~DeviceB-ospf-1-area-0.0.0.1] network 10.100.0.0 0.0.0.255
[*DeviceB-ospf-1-area-0.0.0.1] quit
[*DeviceB-ospf-1] quit
[*DeviceB] commit
```

配置DeviceC。

```
[~DeviceC] vlan 100
[*DeviceC-vlan100] quit
[*DeviceC] interface vlanif 100
[*DeviceC-Vlanif100] ip address 10.100.0.1 255.255.255.0
[*DeviceC-Vlanif100] m-lag ip address 10.100.0.4 255.255.255.0
[*DeviceC-Vlanif100] ospf source sub-address 10.100.0.4
[*DeviceC-Vlanif100] mac-address 0000-5e00-0101
[*DeviceC-Vlanif100] arp proxy enable
[*DeviceC-Vlanif100] quit
[*DeviceC] commit
[~DeviceC] interface eth-trunk 10
[*DeviceC-Eth-Trunk10] port trunk allow-pass vlan 100
[*DeviceC-Eth-Trunk10] quit
[~DeviceC] ospf
[~DeviceC-ospf-1] area 1
[~DeviceC-ospf-1-area-0.0.0.1] network 10.100.0.0 0.0.0.255
[*DeviceC-ospf-1-area-0.0.0.1] quit
[*DeviceC-ospf-1] quit
[*DeviceC] commit
```

3. 配置DeviceA和M-LAG成员设备（DeviceB、DeviceC）建立动态路由OSPF的IP地址。

   # 配置DeviceA。

```
[~DeviceA] vlan 100
[*DeviceA-vlan100] quit
[*DeviceA] interface vlanif 100
[*DeviceA-Vlanif100] ip address 10.100.0.2 255.255.255.0
[*DeviceA-Vlanif100] quit
[*DeviceA] commit
[~DeviceA] interface eth-trunk 10
[*DeviceA-Eth-Trunk10] port trunk allow-pass vlan 100
[*DeviceA-Eth-Trunk10] quit
[*DeviceA] ospf 1 router-id 11.4.4.4
[*DeviceA-ospf-1] area 1
[*DeviceA-ospf-1-area-0.0.0.1] network 10.100.0.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.1] quit
[*DeviceA-ospf-1] quit
[*DeviceA] commit
```

步骤3 配置OSPFv3接入M-LAG。

1. 配置DeviceB、DeviceC、DeviceD的路由协议OSPFv3，实现网络三层互通。

   # 配置DeviceB。DeviceC和DeviceD的配置与DeviceB类似，这里不再赘述。

```
[~DeviceB] interface 100ge 1/0/1
[*DeviceB-100GE1/0/1] undo portswitch
[*DeviceB-100GE1/0/1] ipv6 enable
[*DeviceB-100GE1/0/1] ipv6 address 2001:db8:1::1/64
[*DeviceB-100GE1/0/1] quit
[*DeviceB] ospfv3 1
[*DeviceB-ospfv3-1] router-id 10.12.12.12
[*DeviceB-ospfv3-1] area 0
[*DeviceB-ospfv3-1-area-0.0.0.0] quit
[*DeviceB-ospfv3-1] quit
[*DeviceB] commit
[~DeviceB] interface 100ge 1/0/1
[*DeviceB-100GE1/0/1] ospfv3 1 area 0
[*DeviceB-100GE1/0/1] quit
[*DeviceB] commit
```

OSPFv3成功配置后，DeviceB、DeviceC、DeviceD之间可以建立OSPFv3邻居，且状态为Full。

```
[~DeviceD] display ospfv3 peer
OSPFv3 Process (1)
Total number of peer(s): 2
 Peer(s) in full state: 2
OSPFv3 Area (0.0.0.0)
Neighbor ID     Pri State           Dead Time  Interface         Instance ID
10.12.12.12       1 Full/Backup     00:00:37   100GE1/0/1              0
10.14.14.14       1 Full/DROther    00:00:38   100GE1/0/2              0
```

2. 配置DeviceB和DeviceC的动态路由OSPFv3接入M-LAG的Link-local地址。

   # 配置DeviceB。

```
[~DeviceB] vlan 101
[*DeviceB-vlan101] quit
[*DeviceB] interface vlanif 101
[*DeviceB-Vlanif101] ipv6 enable
[*DeviceB-Vlanif101] ipv6 address 2001:db8:101::1/64
[*DeviceB-Vlanif101] ospfv3 1 area 0.0.0.1
[*DeviceB-Vlanif101] mac-address 0000-5e00-0102
[*DeviceB-Vlanif101] m-lag ipv6 address FE80:1000::1 link-local
[*DeviceB-Vlanif101] ipv6 nd proxy route enable
[*DeviceB-Vlanif101] ipv6 nd na glean
[*DeviceB-Vlanif101] quit
[*DeviceB] commit
[~DeviceB] interface eth-trunk 10
[*DeviceB-Eth-Trunk10] port trunk allow-pass vlan 101
[*DeviceB-Eth-Trunk10] quit
[*DeviceB] commit
```

   # 配置DeviceC。

```
[~DeviceC] vlan 101
[*DeviceC-vlan101] quit
[*DeviceC] interface vlanif 101
[*DeviceC-Vlanif101] ipv6 enable
[*DeviceC-Vlanif101] ipv6 address 2001:db8:101::1/64
[*DeviceC-Vlanif101] ospfv3 1 area 0.0.0.1
[*DeviceC-Vlanif101] mac-address 0000-5e00-0102
[*DeviceC-Vlanif101] m-lag ipv6 address FE80:1000::2 link-local
[*DeviceC-Vlanif101] ipv6 nd proxy route enable
[*DeviceC-Vlanif101] ipv6 nd na glean
[*DeviceC-Vlanif101] quit
[*DeviceC] commit
[~DeviceC] interface eth-trunk 10
[*DeviceC-Eth-Trunk10] port trunk allow-pass vlan 101
[*DeviceC-Eth-Trunk10] quit
[*DeviceC] commit
```

3. 配置DeviceA和M-LAG成员设备（DeviceB、DeviceC）建立动态路由OSPFv3的IP地址。

   # 配置DeviceA。

```
[~DeviceA] ospfv3 1
[*DeviceA-ospfv3-1] router-id 10.15.15.15
[*DeviceA-ospfv3-1] area 1
[*DeviceA-ospfv3-1-area-0.0.0.1] quit
[*DeviceA-ospfv3-1] quit
[*DeviceA] commit
[~DeviceA] vlan 101
[*DeviceA-vlan101] quit
[*DeviceA] interface vlanif 101
[*DeviceA-Vlanif101] ipv6 enable
[*DeviceA-Vlanif101] ipv6 address 2001:db8:101::2/64
[*DeviceA-Vlanif101] ospfv3 1 area 0.0.0.1
[*DeviceA-Vlanif101] quit
```

```
[*DeviceA] commit
[~DeviceA] interface eth-trunk 10
[*DeviceA-Eth-Trunk10] port trunk allow-pass vlan 101
[*DeviceA-Eth-Trunk10] quit
[*DeviceA] commit
```

步骤4 配置BGP接入M-LAG。

1. 配置DeviceB、DeviceC、DeviceD的路由协议BGP，实现网络三层互通。

   # 配置DeviceB的BGP对等体。

   ```
   [~DeviceB] bgp 100
   [*DeviceB-bgp] router-id 10.111.12.12
   [*DeviceB-bgp] peer 192.168.1.2 as-number 200
   [*DeviceB-bgp] quit
   [*DeviceB] commit
   ```

   # 配置DeviceC的BGP对等体。

   ```
   [~DeviceC] bgp 100
   [*DeviceC-bgp] router-id 10.114.14.14
   [*DeviceC-bgp] peer 192.168.2.2 as-number 200
   [*DeviceC-bgp] quit
   [*DeviceC] commit
   ```

   # 配置DeviceD的BGP对等体。

   ```
   [~DeviceD] bgp 200
   [*DeviceD-bgp] router-id 10.113.13.13
   [*DeviceD-bgp] peer 192.168.1.1 as-number 100
   [*DeviceD-bgp] peer 192.168.2.1 as-number 100
   [*DeviceD-bgp] quit
   [*DeviceD] commit
   ```

   BGP成功配置后，DeviceB、DeviceC、DeviceD之间可以建立BGP邻居，且状态为Established。

   ```
   [~DeviceD] display bgp peer
    Status codes: * - Dynamic
    BGP local router ID        : 10.113.13.13
    Local AS number            : 200
    Total number of peers      : 2
    Peers in established state : 2
    Total number of dynamic peers : 0

     Peer          V        AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
     192.168.1.1   4        100       3        3     0 00:00:45 Established    0
     192.168.2.1   4        100       7        7     0 00:03:52 Established    0
   ```

2. 配置DeviceA、DeviceB和DeviceC的动态路由BGP接入M-LAG的IPv4地址。

   # 配置DeviceB。

   ```
   [~DeviceB] vlan 102
   [*DeviceB-vlan102] quit
   [*DeviceB] interface vlanif 102
   [*DeviceB-Vlanif102] ip address 10.102.0.1 255.255.255.0
   [*DeviceB-Vlanif102] m-lag ip address 10.102.0.3 255.255.255.0
   [*DeviceB-Vlanif102] mac-address 0000-5e00-0103
   [*DeviceB-Vlanif102] arp proxy enable
   [*DeviceB-Vlanif102] quit
   [*DeviceB] commit
   [~DeviceB] interface eth-trunk 10
   [*DeviceB-Eth-Trunk10] port trunk allow-pass vlan 102
   [*DeviceB-Eth-Trunk10] quit
   [*DeviceB] commit
   ```

   # 配置DeviceC。

```
[~DeviceC] vlan 102
[*DeviceC-vlan102] quit
[*DeviceC] interface vlanif 102
[*DeviceC-Vlanif102] ip address 10.102.0.1 255.255.255.0
[*DeviceC-Vlanif102] m-lag ip address 10.102.0.4 255.255.255.0
[*DeviceC-Vlanif102] mac-address 0000-5e00-0103
[*DeviceC-Vlanif102] arp proxy enable
[*DeviceC-Vlanif102] quit
[*DeviceC] commit
[~DeviceC] interface eth-trunk 10
[*DeviceC-Eth-Trunk10] port trunk allow-pass vlan 102
[*DeviceC-Eth-Trunk10] quit
[*DeviceC] commit
```

配置DeviceA。

```
[~DeviceA] vlan 102
[*DeviceA-vlan102] quit
[*DeviceA] interface vlanif 102
[*DeviceA-Vlanif102] ip address 10.102.0.2 255.255.255.0
[*DeviceA-Vlanif102] quit
[*DeviceA] commit
[~DeviceA] interface eth-trunk 10
[*DeviceA-Eth-Trunk10] port trunk allow-pass vlan 102
[*DeviceA-Eth-Trunk10] quit
[*DeviceA] commit
```

3. 配置DeviceA和M-LAG成员设备（DeviceB、DeviceC）建立动态路由BGP的对等体。

   # 配置DeviceA。

   ```
   [~DeviceA] bgp 100
   [*DeviceA-bgp] router-id 10.115.15.15
   [*DeviceA-bgp] peer 10.102.0.3 as-number 100
   [*DeviceA-bgp] peer 10.102.0.4 as-number 100
   ```

   # 配置DeviceB。

   ```
   [~DeviceB] bgp 100
   [~DeviceB-bgp] peer 10.102.0.2 as-number 100
   [*DeviceB-bgp] peer 10.102.0.2 connect-interface Vlanif102 10.102.0.3
   ```

   # 配置DeviceC。

   ```
   [~DeviceC] bgp 100
   [~DeviceC-bgp] peer 10.102.0.2 as-number 100
   [*DeviceC-bgp] peer 10.102.0.2 connect-interface Vlanif102 10.102.0.4
   ```

步骤5 配置BGP4+接入M-LAG。

1. 配置DeviceB、DeviceC、DeviceD的路由协议BGP4+，实现网络三层互通。

   # 配置DeviceB的BGP4+对等体和引入路由。

   ```
   [~DeviceB] bgp 100
   [~DeviceB-bgp] peer 2001:db8:1::2 as-number 200
   [*DeviceB-bgp] ipv6-family unicast
   [*DeviceB-bgp-af-ipv6] peer 2001:db8:1::2 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]:y
   [*DeviceB-bgp-af-ipv6] network 2001:db8:1::2 64
   [*DeviceB-bgp-af-ipv6] quit
   [*DeviceB-bgp] quit
   [*DeviceB] commit
   ```

   # 配置DeviceC的BGP4+对等体和引入路由。

   ```
   [~DeviceC] bgp 100
   [~DeviceC-bgp] peer 2001:db8:2::2 as-number 200
   [*DeviceC-bgp] ipv6-family unicast
   ```



```
[*DeviceC-bgp-af-ipv6] peer 2001:db8:2::2 enable
Warning: This operation will reset the peer session. Continue? [Y/N]:y
[*DeviceC-bgp-af-ipv6] network 2001:db8:2::2 64
[*DeviceC-bgp-af-ipv6] quit
[*DeviceC-bgp] quit
[*DeviceC] commit
```

配置DeviceD的BGP4+对等体和引入路由。

```
[~DeviceD] bgp 200
[~DeviceD-bgp] peer 2001:db8:1::1 as-number 100
[*DeviceD-bgp] peer 2001:db8:2::1 as-number 100
[*DeviceD-bgp] ipv6-family unicast
[*DeviceD-bgp-af-ipv6] peer 2001:db8:1::1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]:y
[*DeviceD-bgp-af-ipv6] peer 2001:db8:2::1 enable
Warning: This operation will reset the peer session. Continue? [Y/N]:y
[*DeviceD-bgp-af-ipv6] network 2001:db8:1::1 64
[*DeviceD-bgp-af-ipv6] network 2001:db8:2::1 64
[*DeviceD-bgp-af-ipv6] quit
[*DeviceD-bgp] quit
[*DeviceD] commit
```

BGP4+成功配置后，DeviceB、DeviceC、DeviceD之间可以建立BGP4+邻居，且状态为Established。

```
[~DeviceD] display bgp ipv6 peer
 Status codes: * - Dynamic
 BGP local router ID        : 10.113.13.13
 Local AS number            : 200
 Total number of peers      : 2
 Peers in established state
 Total number of dynamic peers : 0
  Peer            V         AS  MsgRcvd  MsgSent  OutQ  Up/Down       State
PrefRcv
  2001:db8:1::1   4         100        4        4     0 00:00:19 Established        1
  2001:db8:2::1   4         100        4        4     0 00:00:18 Established        1
```

2. 配置DeviceB和DeviceC的动态路由BGP4+接入M-LAG的IPv6地址。

   # 配置DeviceB。

   ```
   [~DeviceB] vlan 103
   [*DeviceB-vlan103] quit
   [*DeviceB] interface vlanif 103
   [*DeviceB-Vlanif103] ipv6 enable
   [*DeviceB-Vlanif103] ipv6 address 2001:db8:103::1/64
   [*DeviceB-Vlanif103] mac-address 0000-5e00-0104
   [*DeviceB-Vlanif103] m-lag ipv6 address 2001:db8:103::3
   [*DeviceB-Vlanif103] ipv6 nd proxy route enable
   [*DeviceB-Vlanif103] ipv6 nd na glean
   [*DeviceB-Vlanif103] quit
   [*DeviceB] commit
   [~DeviceB] interface eth-trunk 10
   [*DeviceB-Eth-Trunk10] port trunk allow-pass vlan 103
   [*DeviceB-Eth-Trunk10] quit
   [*DeviceB] commit
   ```

   # 配置DeviceC。

   ```
   [~DeviceC] vlan 103
   [*DeviceC-vlan103] quit
   [*DeviceC] interface vlanif 103
   [*DeviceC-Vlanif103] ipv6 enable
   [*DeviceC-Vlanif103] ipv6 address 2001:db8:103::1/64
   [*DeviceC-Vlanif103] mac-address 0000-5e00-0104
   [*DeviceC-Vlanif103] m-lag ipv6 address 2001:db8:103::4
   [*DeviceC-Vlanif103] ipv6 nd proxy route enable
   [*DeviceC-Vlanif103] ipv6 nd na glean
   [*DeviceC-Vlanif103] quit
   ```

```
[*DeviceC] commit
[~DeviceC] interface eth-trunk 10
[*DeviceC-Eth-Trunk10] port trunk allow-pass vlan 103
[*DeviceC-Eth-Trunk10] quit
[*DeviceC] commit
```

配置DeviceA。

```
[~DeviceA] vlan 103
[*DeviceA-vlan103] quit
[*DeviceA] interface vlanif 103
[*DeviceA-Vlanif103] ipv6 enable
[*DeviceA-Vlanif103] ipv6 address 2001:db8:103::2/64
[*DeviceA-Vlanif103] quit
[*DeviceA] commit
[~DeviceA] interface eth-trunk 10
[*DeviceA-Eth-Trunk10] port trunk allow-pass vlan 103
[*DeviceA-Eth-Trunk10] quit
[*DeviceA] commit
```

3. 配置DeviceA和M-LAG成员设备（DeviceB、DeviceC）建立动态路由BGP4+的对等体和引入路由。

配置DeviceA。

```
[~DeviceA] bgp 100
[~DeviceA-bgp] peer 2001:db8:103::3 as-number 100
[*DeviceA-bgp] peer 2001:db8:103::4 as-number 100
[*DeviceA-bgp] ipv6-family unicast
[*DeviceA-bgp-af-ipv6] peer 2001:db8:103::3 enable
Warning: This operation will reset the peer session. Continue? [Y/N]:y
[*DeviceA-bgp-af-ipv6] peer 2001:db8:103::4 enable
Warning: This operation will reset the peer session. Continue? [Y/N]:y
[*DeviceA-bgp-af-ipv6] quit
[*DeviceA-bgp] quit
[*DeviceA] commit
```

配置DeviceB。

```
[~DeviceB] bgp 100
[~DeviceB-bgp] peer 2001:db8:103::2 as-number 100
[*DeviceB-bgp] peer 2001:db8:103::2 connect-interface Vlanif 103 2001:db8:103::3
[*DeviceB-bgp] ipv6-family unicast
[*DeviceB-bgp-af-ipv6] peer 2001:db8:103::2 enable
Warning: This operation will reset the peer session. Continue? [Y/N]:y
[*DeviceB-bgp-af-ipv6] quit
[*DeviceB-bgp] quit
[*DeviceB] commit
```

配置DeviceC。

```
[~DeviceC] bgp 100
[~DeviceC-bgp] peer 2001:db8:103::2 as-number 100
[*DeviceC-bgp] peer 2001:db8:103::2 connect-interface Vlanif 103 2001:db8:103::4
[*DeviceC-bgp] ipv6-family unicast
[*DeviceC-bgp-af-ipv6] peer 2001:db8:103::2 enable
Warning: This operation will reset the peer session. Continue? [Y/N]:y
[*DeviceC-bgp-af-ipv6] quit
[*DeviceC-bgp] quit
[*DeviceC] commit
```

步骤6 验证配置结果

- 执行命令**display dfs-group 1 m-lag**，查看M-LAG的相关信息。

  # 查看DFS Group编号为1的M-LAG信息。

```
[~DeviceB] display dfs-group 1 m-lag
*             : Local node
Heart beat state : OK
Node 1 *
```

```
  Dfs-Group ID       : 1
  Priority           : 100
  Dual-active Address : 10.200.1.1
  VPN-Instance       :  public net
  State              : Master
  Causation          : -
  System ID          : 0000-5e95-7c11
  SysName            : DeviceB
  Version            : V300R020C00SPC100
  Device Series      : CE6800      //从V300R022C00版本开始，此条回显信息修改为Device Type    :
CE6800
Node 2
  Dfs-Group ID       : 1
  Priority           : 100
  Dual-active Address : 10.200.2.1
  VPN-Instance       :  public net
  State              : Backup
  Causation          : -
  System ID          : 0000-5e95-7c31
  SysName            : DeviceC
  Version            : V300R020C00SPC100
  Device Series      : CE6800      //从V300R022C00版本开始，此条回显信息修改为Device Type    :
CE6800
```

查看DeviceB上的M-LAG信息。

```
[~DeviceB] display dfs-group 1 node 1 m-lag brief
* - Local node

M-Lag ID    Interface      Port State   Status              Consistency-check
      1     Eth-Trunk 10   Up           active(*)-active    --

Failed reason:
   1 -- Relationship between vlan and port is inconsistent
   2 -- STP configuration under the port is inconsistent
   3 -- STP port priority configuration is inconsistent
   4 -- LACP mode of M-LAG is inconsistent
   5 -- M-LAG configuration is inconsistent
   6 -- The number of M-LAG members is inconsistent
   7 -- LACP system-id of M-LAG is inconsistent
   8 -- LACP priority of M-LAG is inconsistent
   9 -- STP port edged configuration is inconsistent
   10 -- M-LAG mode configuration is inconsistent
```

说明

从V300R022C00版本开始设备支持7、8、9、10项M-LAG一致性检查。

查看DeviceC上的M-LAG信息。

```
[~DeviceC] display dfs-group 1 node 2 m-lag brief
* - Local node

M-Lag ID    Interface      Port State   Status              Consistency-check
      1     Eth-Trunk 10   Up           active(*)-active    --

Failed reason:
   1 -- Relationship between vlan and port is inconsistent
   2 -- STP configuration under the port is inconsistent
   3 -- STP port priority configuration is inconsistent
   4 -- LACP mode of M-LAG is inconsistent
   5 -- M-LAG configuration is inconsistent
   6 -- The number of M-LAG members is inconsistent
   7 -- LACP system-id of M-LAG is inconsistent
   8 -- LACP priority of M-LAG is inconsistent
   9 -- STP port edged configuration is inconsistent
   10 -- M-LAG mode configuration is inconsistent
```

说明

从V300R022C00版本开始设备支持7、8、9、10项M-LAG一致性检查。

- 在DeviceB、DeviceC、DeviceA上执行**display ospf peer brief**命令，可查看到OSPF中各邻居的信息。

  # 查看DeviceB上的OSPF邻居信息。

```
[~DeviceB] display ospf peer brief
      OSPF Process 1 with Router ID 11.1.1.1
                Peer Statistic Information
 Total number of peer(s): 3
 Peer(s) in full state: 3
 --------------------------------------------------------------------------------
 Area Id         Interface                Neighbor id      State
 0.0.0.0         100GE1/0/1               11.3.3.3         Full
 0.0.0.0         Vlanif100                11.2.2.2         Full
 0.0.0.1         Vlanif100                11.4.4.4         Full
```

  # 查看DeviceC上的OSPF邻居信息。

```
[~DeviceC] display ospf peer brief
      OSPF Process 1 with Router ID 11.2.2.2
                Peer Statistic Information
 Total number of peer(s): 3
 Peer(s) in full state: 3
 --------------------------------------------------------------------------------
 Area Id         Interface                Neighbor id      State
 0.0.0.0         100GE1/0/1               11.3.3.3         Full
 0.0.0.0         Vlanif100                11.1.1.1         Full
 0.0.0.1         Vlanif100                11.4.4.4         Full
```

  # 查看DeviceA上的OSPF邻居信息。

```
[~DeviceA] display ospf peer brief
      OSPF Process 1 with Router ID 11.4.4.4
                Peer Statistic Information
 Total number of peer(s): 2
 Peer(s) in full state: 2
 --------------------------------------------------------------------------------
 Area Id         Interface                Neighbor id      State
 0.0.0.1         Vlanif100                11.1.1.1         Full
 0.0.0.1         Vlanif100                11.2.2.2         Full
```

- 在DeviceB、DeviceC、DeviceA上执行**display ospfv3 peer**命令，可查看到OSPFv3中各邻居的信息。

  # 查看DeviceB上的OSPFv3邻居信息。

```
[~DeviceB] display ospfv3 peer
OSPFv3 Process (1)
Total number of peer(s): 3
 Peer(s) in full state: 3
OSPFv3 Area (0.0.0.0)
Neighbor ID      Pri State           Dead Time  Interface         Instance ID
10.13.13.13        1 Full/Backup     00:00:37   100GE1/0/1          0
10.14.14.14        1 Full/DROther    00:00:38   Vlanif101           0

OSPFv3 Area (0.0.0.1)
Neighbor ID      Pri State           Dead Time  Interface         Instance ID
10.15.15.15        1 Full/DR         00:00:40   Vlanif101           0
```

  # 查看DeviceC上的OSPFv3邻居信息。

```
[~DeviceC] display ospfv3 peer
OSPFv3 Process (1)
Total number of peer(s): 3
 Peer(s) in full state: 3
OSPFv3 Area (0.0.0.0)
Neighbor ID      Pri State           Dead Time  Interface         Instance ID
10.13.13.13        1 Full/DR         00:00:37   100GE1/0/1          0
10.12.12.12        1 Full/DROther    00:00:38   Vlanif101           0

OSPFv3 Area (0.0.0.1)
Neighbor ID      Pri State           Dead Time  Interface         Instance ID
10.15.15.15        1 Full/DR         00:00:40   Vlanif101           0
```

  # 查看DeviceA上的OSPFv3邻居信息。

```
[~DeviceA]display ospfv3 peer
OSPFv3 Process (1)
Total number of peer(s): 2
 Peer(s) in full state: 2
OSPFv3 Area (0.0.0.1)
Neighbor ID     Pri State           Dead Time  Interface         Instance ID
10.14.14.14       1 Full/DROther    00:00:38   Vlanif101           0
10.12.12.12       1 Full/Backup     00:00:40   Vlanif101           0
```

- 在DeviceB、DeviceC、DeviceA上执行**display bgp peer**命令，可查看到BGP中各邻居的信息。

  # 查看DeviceB上的BGP邻居信息。

```
[~DeviceB] display bgp peer
 Status codes: * - Dynamic
 BGP local router ID        : 10.111.12.12
 Local AS number            : 100
 Total number of peers      : 2
 Peers in established state : 2
 Total number of dynamic peers : 0

 Peer                    V         AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
 10.102.0.2                4        100       30       30     0 00:24:37 Established        0
 192.168.1.2               4        200       64       66     0 00:54:13 Established        0
```

  # 查看DeviceC上的BGP邻居信息。

```
[~DeviceC] display bgp peer
 Status codes: * - Dynamic
 BGP local router ID        : 10.114.14.14
 Local AS number            : 100
 Total number of peers      : 59
 Peers in established state : 2
 Total number of dynamic peers : 0

 Peer                    V         AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
 10.102.0.2                4        100       34       33     0 00:26:56 Established        0
 192.168.2.2               4        200       71       72     0 00:59:44 Established        0
```

  # 查看DeviceA上的BGP邻居信息。

```
[~DeviceA] display bgp peer
2020-07-07 21:06:02.443
 Status codes: * - Dynamic
 BGP local router ID        : 10.115.15.15
 Local AS number            : 100
 Total number of peers      : 2
 Peers in established state : 2
 Total number of dynamic peers : 0

 Peer                    V         AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
 10.102.0.3                4        100       29       29     0 00:23:22 Established        0
 10.102.0.4                4        100       29       30     0 00:23:18 Established        0
```

- 在DeviceB、DeviceC、DeviceA上执行**display bgp ipv6 peer**命令，可查看到BGP4+中各邻居的信息。

  # 查看DeviceB上的BGP4+邻居信息。

```
[~DeviceB] display bgp ipv6 peer
 Status codes: * - Dynamic
 BGP local router ID        : 10.111.12.12
 Local AS number            : 100
 Total number of peers      : 2
 Peers in established state : 2
 Total number of dynamic peers : 0

 Peer                    V         AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
 2001:DB8:1::2               4        200       61       61     0 00:50:07 Established        2
 2001:DB8:103::2             4        100        5        6     0 00:02:18 Established        0
```

  # 查看DeviceC上的BGP4+邻居信息。

```
[~DeviceC] display bgp ipv6 peer
 Status codes: * - Dynamic
 BGP local router ID        : 10.114.14.14
 Local AS number            : 100
 Total number of peers      : 2
 Peers in established state : 2
 Total number of dynamic peers : 0

 Peer                    V         AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
 2001:DB8:2::2               4        200       76       75     0 01:03:13 Established        2
 2001:DB8:103::2             4        100       21       22     0 00:15:55 Established        0
```

查看DeviceA上的BGP4+邻居信息。

```
[~DeviceA] display bgp ipv6 peer
 Status codes: * - Dynamic
 BGP local router ID        : 10.115.15.15
 Local AS number            : 100
 Total number of peers      : 2
 Peers in established state : 2
 Total number of dynamic peers : 0

 Peer                    V         AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
 2001:DB8:103::3             4        100        7        7     0 00:03:02 Established        1
 2001:DB8:103::4             4        100        7        6     0 00:02:33 Established        1
```

----结束

配置脚本

- DeviceA的配置脚本

```
#
sysname DeviceA
#
vlan batch 100 to 103
#
ospfv3 1
 router-id 10.15.15.15
 area 0.0.0.1
#
interface Vlanif100
 ip address 10.100.0.2 255.255.255.0
#
interface Vlanif101
 ipv6 enable
 ipv6 address 2001:DB8:101::2/64
 ospfv3 1 area 0.0.0.1
#
interface Vlanif102
 ip address 10.102.0.2 255.255.255.0
#
interface Vlanif103
 ipv6 enable
 ipv6 address 2001:DB8:103::2/64
#
interface Eth-Trunk10
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 100 101 102 103
 mode lacp-static
 lacp mixed-rate link enable
#
interface 100GE1/0/1
 eth-trunk 10
#
interface 100GE1/0/2
 eth-trunk 10
#
interface 100GE1/0/3
```



```
 eth-trunk 10
#
interface 100GE1/0/4
 eth-trunk 10
#
bgp 100
 router-id 10.115.15.15
 peer 10.102.0.3 as-number 100
 peer 10.102.0.4 as-number 100
 peer 2001:DB8:103::3 as-number 100
 peer 2001:DB8:103::4 as-number 100
 #
 ipv4-family unicast
  peer 10.102.0.3 enable
  peer 10.102.0.4 enable
 #
 ipv6-family unicast
  peer 2001:DB8:103::3 enable
  peer 2001:DB8:103::4 enable
#
ospf 1 router-id 11.4.4.4
 area 0.0.0.1
  network 10.100.0.0 0.0.0.255
#
return
```

- DeviceB的配置文件

```
#
sysname Leaf
#
dfs-group 1
 dual-active detection source ip 10.200.1.1 peer 10.200.2.1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgw-h
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
#
vlan batch 100 to 103
#
stp mode rstp
stp v-stp enable
#
ospfv3 1
 router-id 10.12.12.12
 area 0.0.0.0
 area 0.0.0.1
#
interface Vlanif100
 ip address 10.100.0.1 255.255.255.0
 ospf source sub-address 10.100.0.3
 mac-address 0000-5e00-0101
 m-lag ip address 10.100.0.3 255.255.255.0
 arp proxy enable
#
interface Vlanif101
 ipv6 enable
 ipv6 address 2001:DB8:101::1/64
 ospfv3 1 area 0.0.0.1
 mac-address 0000-5e00-0102
 m-lag ipv6 address FE80:1000::1 link-local
 ipv6 nd proxy route enable
 ipv6 nd na glean
#
interface Vlanif102
 ip address 10.102.0.1 255.255.255.0
 mac-address 0000-5e00-0103
 m-lag ip address 10.102.0.3 255.255.255.0
 arp proxy enable
#
interface Vlanif103
 ipv6 enable
 ipv6 address 2001:DB8:103::1/64
```



```
 mac-address 0000-5e00-0104
 m-lag ipv6 address 2001:DB8:103::3
 ipv6 nd proxy route enable
 ipv6 nd na glean
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk10
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 100 101 102 103
 mode lacp-static
 lacp mixed-rate link enable
 dfs-group 1 m-lag 1
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ip address 192.168.1.1 255.255.255.0
 ipv6 address 2001:DB8:1::1/64
 ospfv3 1 area 0.0.0.0
#
interface 100GE1/0/2
 eth-trunk 10
#
interface 100GE1/0/3
 eth-trunk 10
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 1
#
interface LoopBack1
 ip address 10.2.2.2 255.255.255.255
#
interface LoopBack2
 ip address 10.3.3.3 255.255.255.255
#
monitor-link group 1
 port 100ge 1/0/1 uplink
 port eth-trunk 10 downlink 1
#
bgp 100
 router-id 10.111.12.12
 peer 10.102.0.2 as-number 100
 peer 10.102.0.2 connect-interface Vlanif102 10.102.0.3
 peer 192.168.1.2 as-number 200
 peer 2001:DB8:1::2 as-number 200
 peer 2001:DB8:103::2 as-number 100
 peer 2001:DB8:103::2 connect-interface Vlanif103 2001:DB8:103::3
 #
 ipv4-family unicast
  peer 10.102.0.2 enable
  peer 192.168.1.2 enable
#
 ipv6-family unicast
  network 2001:DB8:1:: 64
  peer 2001:DB8:1::2 enable
  peer 2001:DB8:103::2 enable
#
ospf 1 router-id 11.1.1.1
 area 0.0.0.0
  network 10.2.2.2 0.0.0.0
  network 10.3.3.3 0.0.0.0
  network 192.168.1.0 0.0.0.255
```



```
 area 0.0.0.1
  network 10.100.0.0 0.0.0.255
#
return
```

- DeviceC的配置脚本

```
#
sysname DeviceC
#
dfs-group 1
 dual-active detection source ip 10.200.2.1 peer 10.200.1.1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!=I9f8>C{!P_bhB31@7r-=jrS8c|
_"(Bn~#=!!!!!!!!!!!!!!!9!!!!kx-6@.tGA(wAt/IQXl6>[g{6YlOi9$!!!!!!!!!!%+%#
#
vlan batch 100 to 103
#
stp mode rstp
stp v-stp enable
#
ospfv3 1
 router-id 10.14.14.14
 area 0.0.0.0
 area 0.0.0.1
#
interface Vlanif100
 ip address 10.100.0.1 255.255.255.0
 ospf source sub-address 10.100.0.4
 mac-address 0000-5e00-0101
 m-lag ip address 10.100.0.4 255.255.255.0
 arp proxy enable
#
interface Vlanif101
 ipv6 enable
 ipv6 address 2001:db8:101::1/64
 ospfv3 1 area 0.0.0.1
 mac-address 0000-5e00-0102
 m-lag ipv6 address FE80:1000::2 link-local
 ipv6 nd proxy route enable
 ipv6 nd na glean
#
interface Vlanif102
 ip address 10.102.0.1 255.255.255.0
 mac-address 0000-5e00-0103
 m-lag ip address 10.102.0.4 255.255.255.0
 arp proxy enable
#
interface Vlanif103
 ipv6 enable
 ipv6 address 2001:DB8:103::1/64
 mac-address 0000-5e00-0104
 m-lag ipv6 address 2001:DB8:103::4
 ipv6 nd proxy route enable
 ipv6 nd na glean
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk10
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 100 101 102 103
 mode lacp-static
 lacp mixed-rate link enable
 dfs-group 1 m-lag 1
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ip address 192.168.2.1 255.255.255.0
```



```
 ipv6 address 2001:db8:2::1/64
 ospfv3 1 area 0.0.0.0
#
interface 100GE1/0/2
 eth-trunk 10
#
interface 100GE1/0/3
 eth-trunk 10
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 1
#
interface LoopBack1
 ip address 10.2.2.2 255.255.255.255
#
interface LoopBack2
 ip address 10.4.4.4 255.255.255.255
#
monitor-link group 1
 port 100ge 1/0/1 uplink
 port eth-trunk 10 downlink 1
#
bgp 100
 router-id 10.114.14.14
 peer 10.102.0.2 as-number 100
 peer 10.102.0.2 connect-interface Vlanif102 10.102.0.4
 peer 192.168.2.2 as-number 200
 peer 2001:DB8:2::2 as-number 200
 peer 2001:DB8:103::2 as-number 100
 peer 2001:DB8:103::2 connect-interface Vlanif103 2001:DB8:103::4
 #
 ipv4-family unicast
  peer 10.102.0.2 enable
  peer 192.168.2.2 enable
 #
 ipv6-family unicast
  network 2001:DB8:2:: 64
  peer 2001:DB8:2::2 enable
  peer 2001:DB8:103::2 enable
#
ospf 1 router-id 11.2.2.2
 area 0.0.0.0
  network 10.2.2.2 0.0.0.0
  network 10.4.4.4 0.0.0.0
  network 192.168.2.0 0.0.0.255
 area 0.0.0.1
  network 10.100.0.0 0.0.0.255
#
return
```

- DeviceD的配置脚本

```
#
sysname DeviceD
#
ospfv3 1
 router-id 10.13.13.13
 area 0.0.0.0
#
interface 100GE1/0/1
 undo portswitch
 ipv6 enable
 ip address 192.168.1.2 255.255.255.0
 ipv6 address 2001:db8:1::2/64
 ospfv3 1 area 0.0.0.0
#
interface 100GE1/0/2
 undo portswitch
```

```
 ipv6 enable
 ip address 192.168.2.2 255.255.255.0
 ipv6 address 2001:db8:2::2/64
 ospfv3 1 area 0.0.0.0
#
interface LoopBack1
 ip address 10.1.1.1 255.255.255.255
#
bgp 200
 router-id 10.113.13.13
 peer 192.168.1.1 as-number 100
 peer 192.168.2.1 as-number 100
 peer 2001:db8:1::1 as-number 100
 peer 2001:db8:2::1 as-number 100
 #
 ipv4-family unicast
  peer 192.168.1.1 enable
  peer 192.168.2.1 enable
 #
 ipv6-family unicast
  network 2001:db8:1:: 64
  network 2001:db8:2:: 64
  peer 2001:db8:1::1 enable
  peer 2001:db8:2::1 enable
#
ospf 1 router-id 11.3.3.3
 area 0.0.0.0
  network 10.1.1.1 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.2.0 0.0.0.255
#
return
```

1.13 配置组播源在外部网络的 IPv4 三层组播 over VXLAN 示例

适用产品和版本

- CE16800（仅P系列单板和SAN系列单板）、CE8800、CE6800（除CE6820H、CE6820H-K、CE6820S外）系列产品V300R021C10或更高版本。
- 如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

如**配置组播源在外部网络的IPv4三层组播 over VXLAN组网图**所示，在IPv4网络中，Borderleaf1、Borderleaf2、Leaf1和Leaf2均部署VXLAN分布式网关，为了保证高可靠性，Borderleaf和Leaf均采用M-LAG部署方式。组播源Source处于外部网络，组播接收者Receiver处于VXLAN Overlay网络。Source与CE直连，CE通过双活网关Borderleaf1和Borderleaf2接入VXLAN网络；Receiver通过双活网关Leaf1和Leaf2上的二层子接口接入VXLAN网络，属于BD10。Receiver通过不指定组播源的方式进行点播，点播的组地址为225.1.1.1。

为了满足上述用户的需求，需要在部署分布式网关的VXLAN网络中配置IPv4三层组播。

图 1-13 配置组播源在外部网络的 IPv4 三层组播 over VXLAN 组网图

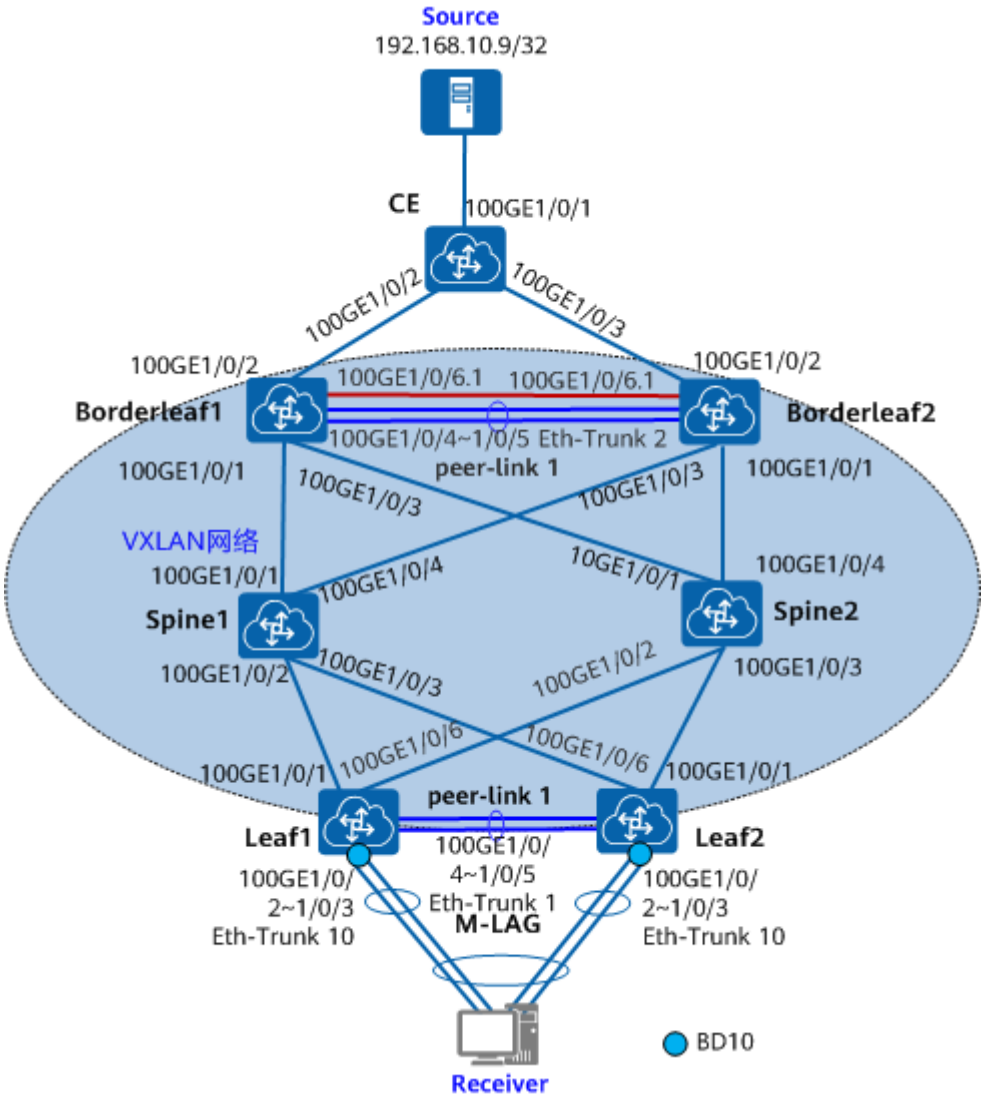


表 1-14 数据准备表

设备	接口	IP地址	对接设备
CE	100GE1/0/1	192.168.10.2/24	Source
	100GE1/0/2	192.168.20.2/24	Borderleaf1
	100GE1/0/3	192.168.30.2/24	Borderleaf2
Borderleaf1	100GE1/0/1	10.1.1.1/24	Spine1
	100GE1/0/2	192.168.20.1/24	CE
	VLANIF 3000	10.10.21.1/24	Borderleaf2
	100GE1/0/3	10.1.11.1/24	Spine2
	100GE1/0/6.1	10.6.1.1/24	Borderleaf2

设备	接口	IP地址	对接设备
	LoopBack1	1.1.1.1/32	NA
	LoopBack2	1.1.1.110/32	NA
	LoopBack3	1.1.1.10/32（**私网RP**）	NA
	LoopBack4	10.10.10.10/32	Borderleaf2
	MEth0/0/0	10.10.11.1/24	NA
Borderleaf2	100GE1/0/1	10.1.44.1/24	Spine2
	100GE1/0/2	192.168.30.1/24	CE
	VLANIF 3000	10.10.21.2/24	Borderleaf1
	100GE1/0/3	10.1.4.1/24	Spine1
	100GE1/0/6.1	10.6.1.2/24	Borderleaf1
	LoopBack1	2.2.2.2/32	NA
	LoopBack2	1.1.1.110/32	NA
	LoopBack3	10.10.20.10/32	Borderleaf1
	MEth0/0/0	10.10.11.2/24	NA
Spine1	100GE1/0/1	10.1.1.2/24	Borderleaf1
	100GE1/0/2	10.1.2.2/24	Leaf1
	100GE1/0/3	10.1.3.2/24	Leaf2
	100GE1/0/4	10.1.4.2/24	Borderleaf2
	LoopBack1	5.5.5.5/32（**公网RP**）	NA
	LoopBack2	5.5.5.1/32	NA
Spine2	100GE1/0/1	10.1.11.2/24	Borderleaf1
	100GE1/0/2	10.1.22.2/24	Leaf1
	100GE1/0/3	10.1.33.2/24	Leaf2
	100GE1/0/4	10.1.44.2/24	Borderleaf2
	LoopBack1	5.5.5.5/32（**公网RP**）	NA
	LoopBack2	5.5.5.2/32	NA
Leaf1	100GE1/0/1	10.1.2.1/24	Spine1
	VBDIF 10	192.168.10.1/24	Receiver

设备	接口	IP地址	对接设备
	VLANIF 4000	10.10.20.1/24	Leaf2
	100GE1/0/6	10.1.22.1/24	Spine2
	LoopBack1	3.3.3.3/32	NA
	LoopBack2	2.2.2.210/32	NA
	LoopBack3	10.10.30.10/32	Leaf2
	MEth0/0/0	10.10.10.1/24	NA
Leaf2	100GE1/0/1	10.1.33.1/24	Spine1
	VBDIF 10	192.168.10.1/24	Receiver
	VLANIF 4000	10.10.20.2/24	Leaf1
	100GE1/0/6	10.1.3.1/24	Spine1
	LoopBack1	4.4.4.4/32	NA
	LoopBack2	2.2.2.210/32	NA
	LoopBack3	10.10.40.10/32	Leaf1
	MEth0/0/0	10.10.10.2/24	NA

配置思路

说明

要实现以上需求，需要保证Borderleaf1和Borderleaf2组成的M-LAG主备设备之间除了peer-link链路外，还有直连的三层链路，并且该三层链路需要使能PIM协议，同时要求M-LAG主备设备上的组播配置保持一致。

可按照以下思路进行配置：

1. 在Borderleaf1和Borderleaf2、Leaf1和Leaf2之间配置M-LAG。
2. 在各Leaf上配置BGP EVPN方式建立VXLAN隧道，并部署分布式网关，通过私网路由实现Leaf侧的主机三层互通。
3. 在各Leaf上针对L3VPN实例下的三层VNI配置BUM组播复制。
4. 在Borderleaf1、Borderleaf2、Leaf1和Leaf2之间配置BGP MVPN邻居。
5. 在各Leaf上配置VXLAN类型的I-PMSI隧道。
6. 在各Leaf上绑定L3VPN实例的接口，使能PIM SM，建立私网组播路由表。
7. 在各Leaf上配置Monitor Link关联上行接口和下行接口。
8. 在各VBDIF接口，以及与用户网段相连的组播物理接口上配置IGMP协议。
9. 在Borderleaf1、Borderleaf2和CE之间配置私网三层组播。

数据准备

为完成此配置例，如**表1 数据准备表**所示，需准备如下的数据：

- 处于VXLAN Overlay网络的接收者所属的VLAN ID为VLAN 10。
- 广播域BD ID为BD 10。
- BD下VXLAN网络标识VNI ID为VNI 10。
- L3VPN实例名称为mcast1，L3VPN实例下VXLAN网络标识VNI ID是VNI 5010。
- EVPN实例的RD值11:1，11:2，RT值为12:1，12:2，13:1。
- L3VPN实例的RD值是1:1，RT值为1:1，与EVPN实例交互的RT值为13:1。
- 针对L3VPN实例下的三层VNI进行BUM组播复制的组播组地址为225.0.0.1。
- 各Leaf节点的MVPN ID为各自Loopback2接口IP地址
- 各Leaf节点的本地VPN实例标识为1。
- 公网实例下的静态RP为5.5.5.5，私网实例下的静态RP为1.1.1.10。

操作步骤

步骤1 配置各交换机接口的IP地址及单播路由协议。

按照表1-14配置各交换机接口的IP地址和掩码，并配置各交换机之间采用OSPF进行互连，确保网络中各交换机间能够在网络层互通。CE、Borderleaf2、Leaf1、Leaf2、Spine1、Spine2上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
<HUAWEI> system-view
[HUAWEI] sysname Borderleaf1
[*HUAWEI] commit
[~Borderleaf1] interface meth 0/0/0
[~Borderleaf1-MEth0/0/0] ip address 10.10.11.1 24
[*Borderleaf1-MEth0/0/0] quit
[~Borderleaf1] interface loopback 1
[*Borderleaf1-LoopBack1] ip address 1.1.1.1 32
[*Borderleaf1-LoopBack1] quit
[~Borderleaf1] interface loopback 2
[*Borderleaf1-LoopBack2] ip address 1.1.1.110 32
[*Borderleaf1-LoopBack2] quit
[*Borderleaf1] interface 100GE 1/0/1
[*Borderleaf1-100GE1/0/1] undo portswitch
[*Borderleaf1-100GE1/0/1] ip address 10.1.1.1 24
[*Borderleaf1-100GE1/0/1] quit
[*Borderleaf1] interface 100GE 1/0/3
[*Borderleaf1-100GE1/0/3] undo portswitch
[*Borderleaf1-100GE1/0/3] ip address 10.1.11.1 24
[*Borderleaf1-100GE1/0/3] quit
[*Borderleaf1] ospf
[*Borderleaf1-ospf-1] area 0
[*Borderleaf1-ospf-1-area-0.0.0.0] network 1.1.1.1 0.0.0.0
[*Borderleaf1-ospf-1-area-0.0.0.0] network 1.1.1.110 0.0.0.0
[*Borderleaf1-ospf-1-area-0.0.0.0] network 10.1.1.0 0.0.0.255
[*Borderleaf1-ospf-1-area-0.0.0.0] network 10.1.11.0 0.0.0.255
[*Borderleaf1-ospf-1-area-0.0.0.0] network 10.6.1.0 0.0.0.255
[*Borderleaf1-ospf-1-area-0.0.0.0] network 10.10.11.0 0.0.0.255
[*Borderleaf1-ospf-1-area-0.0.0.0] quit
[*Borderleaf1-ospf-1] quit
[*Borderleaf1] commit
```

OSPF成功配置后，交换机之间可通过OSPF协议发现对方接口的IP地址，并能互相ping通。例如，Borderleaf1 ping Borderleaf2上的Loopback 1接口地址的显示如下：

```
[~Borderleaf1] ping 2.2.2.2
  PING 2.2.2.2: 56  data bytes, press CTRL_C to break
    Reply from 2.2.2.2: bytes=56 Sequence=1 ttl=254 time=2 ms
    Reply from 2.2.2.2: bytes=56 Sequence=2 ttl=254 time=1 ms
    Reply from 2.2.2.2: bytes=56 Sequence=3 ttl=254 time=1 ms
```

```
  Reply from 2.2.2.2: bytes=56 Sequence=4 ttl=254 time=1 ms
  Reply from 2.2.2.2: bytes=56 Sequence=5 ttl=254 time=1 ms

--- 2.2.2.2 ping statistics ---
  5 packet(s) transmitted
  5 packet(s) received
  0.00% packet loss
  round-trip min/avg/max = 1/1/1 ms
```

步骤2 在Borderleaf1与Borderleaf2上配置基于V-STP的M-LAG，并配置M-LAG主备设备之间通过三层子接口直连的三层链路。

配置Borderleaf1。Borderleaf2上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] stp mode rstp
[*Borderleaf1] stp v-stp enable
[*Borderleaf1] dfs-group 1
[*Borderleaf1-dfs-group-1] dual-active detection source ip 10.10.11.1 peer 10.10.11.2
[*Borderleaf1-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Borderleaf1-dfs-group-1] quit
[*Borderleaf1] interface eth-trunk 2
[*Borderleaf1-Eth-Trunk2] trunkport 100GE 1/0/4
[*Borderleaf1-Eth-Trunk2] trunkport 100GE 1/0/5
[*Borderleaf1-Eth-Trunk2] mode lacp-static
[*Borderleaf1-Eth-Trunk2] peer-link 1
[*Borderleaf1-Eth-Trunk2] quit
[*Borderleaf1] interface 100GE 1/0/6
[*Borderleaf1-100GE1/0/6] undo portswitch
[*Borderleaf1-100GE1/0/6] quit
[*Borderleaf1] commit
[~Borderleaf1] interface 100GE 1/0/6.1
[*Borderleaf1-100GE1/0/6.1] ip address 10.6.1.1 24
[*Borderleaf1-100GE1/0/6.1] quit
[*Borderleaf1] commit
```

步骤3 在Leaf1与Leaf2上配置基于V-STP的M-LAG。

配置Leaf1。Leaf2上的配置过程与Leaf1上的配置类似，配置过程略，详见配置脚本。

说明

如果Leaf1上行接入VXLAN网络的链路出现故障，当用户流量到达Leaf1时，由于没有可用的上行出接口，Leaf1会将用户流量全部丢弃。此时，可以配置monitor-link关联Leaf1的上行接口和下行接口，当Leaf1的上行出接口DOWN时，下行接口状态也会变为DOWN，这样用户侧流量就不会通过Leaf1进行转发，从而防止流量被丢弃。

```
[~Leaf1] stp mode rstp
[*Leaf1] stp v-stp enable
[*Leaf1] dfs-group 1
[*Leaf1-dfs-group-1] authentication-mode hmac-sha256 password YsHsjx_202206
[*Leaf1-dfs-group-1] dual-active detection source ip 10.10.10.1 peer 10.10.10.2
[*Leaf1-dfs-group-1] quit
[*Leaf1] interface eth-trunk 1
[*Leaf1-Eth-Trunk1] trunkport 100GE 1/0/4
[*Leaf1-Eth-Trunk1] trunkport 100GE 1/0/5
[*Leaf1-Eth-Trunk1] mode lacp-static
[*Leaf1-Eth-Trunk1] peer-link 1
[*Leaf1-Eth-Trunk1] quit
[*Leaf1] interface eth-trunk 10
[*Leaf1-Eth-Trunk10] trunkport 100GE 1/0/2 to 1/0/3
[*Leaf1-Eth-Trunk10] mode lacp-static
[*Leaf1-Eth-Trunk10] dfs-group 1 m-lag 1
[*Leaf1-Eth-Trunk10] stp edged-port enable
[*Leaf1-Eth-Trunk10] quit
[*Leaf1] commit
```

步骤4 配置VXLAN。

在Leaf1与Leaf2上配置业务接入点。

配置Leaf1。Leaf2上的配置过程与Leaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Leaf1] bridge-domain 10
[*Leaf1-bd10] quit
[*Leaf1] interface eth-trunk 10.1 mode l2
[*Leaf1-Eth-Trunk10.1] encapsulation dot1q vid 10
[*Leaf1-Eth-Trunk10.1] bridge-domain 10
[*Leaf1-Eth-Trunk10.1] quit
[*Leaf1] commit
```

在各Leaf上配置BGP EVPN作为VXLAN控制面协议并配置BGP EVPN对等体。其他Leaf上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] evpn-overlay enable
[*Borderleaf1] bgp 100
[*Borderleaf1] router-id 1.1.1.1
[*Borderleaf1-bgp] peer 2.2.2.2 as-number 100
[*Borderleaf1-bgp] peer 2.2.2.2 connect-interface LoopBack1
[*Borderleaf1-bgp] peer 3.3.3.3 as-number 100
[*Borderleaf1-bgp] peer 3.3.3.3 connect-interface LoopBack1
[*Borderleaf1-bgp] peer 4.4.4.4 as-number 100
[*Borderleaf1-bgp] peer 4.4.4.4 connect-interface LoopBack1
[*Borderleaf1-bgp] l2vpn-family evpn
[*Borderleaf1-bgp-af-evpn] peer 2.2.2.2 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Borderleaf1-bgp-af-evpn] peer 3.3.3.3 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Borderleaf1-bgp-af-evpn] peer 4.4.4.4 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Borderleaf1-bgp-af-evpn] quit
[*Borderleaf1-bgp] quit
[*Borderleaf1] commit
```

在Borderleaf1和Borderleaf2上配置L3VPN实例。Borderleaf2上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] ip vpn-instance mcast1
[*Borderleaf1-vpn-instance-mcast1] vxlan vni 5010
[*Borderleaf1-vpn-instance-mcast1] ipv4-family
[*Borderleaf1-vpn-instance-mcast1-af-ipv4] route-distinguisher 1:1
[*Borderleaf1-vpn-instance-mcast1-af-ipv4] vpn-target 1:1
[*Borderleaf1-vpn-instance-mcast1-af-ipv4] vpn-target 13:1 evpn
[*Borderleaf1-vpn-instance-mcast1-af-ipv4] quit
[*Borderleaf1-vpn-instance-mcast1] quit
[*Borderleaf1] commit
```

在Leaf1和Leaf2上配置L3VPN实例和EVPN实例。Leaf2上的配置过程与Leaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Leaf1] ip vpn-instance mcast1
[*Leaf1-vpn-instance-mcast1] vxlan vni 5010
[*Leaf1-vpn-instance-mcast1] ipv4-family
[*Leaf1-vpn-instance-mcast1-af-ipv4] route-distinguisher 1:1
[*Leaf1-vpn-instance-mcast1-af-ipv4] vpn-target 1:1
[*Leaf1-vpn-instance-mcast1-af-ipv4] vpn-target 13:1 evpn
[*Leaf1-vpn-instance-mcast1-af-ipv4] quit
[*Leaf1-vpn-instance-mcast1] quit
[*Leaf1] bridge-domain 10
[*Leaf1-bd10] vxlan vni 10
[*Leaf1-bd10] evpn
[*Leaf1-bd10-evpn] route-distinguisher 11:1
[*Leaf1-bd10-evpn] vpn-target 12:1
[*Leaf1-bd10-evpn] vpn-target 13:1
[*Leaf1-bd10-evpn] quit
[*Leaf1-bd10] quit
[*Leaf1] commit
```

在各Leaf上配置头端复制功能。

- 配置Borderleaf1。
  ```
  [~Borderleaf1] interface nve 1
  [*Borderleaf1-Nve1] source 1.1.1.110
  [*Borderleaf1-Nve1] vni 5010 head-end peer-list protocol bgp
  [*Borderleaf1-Nve1] mac-address 0000-5e00-0101
  [*Borderleaf1-Nve1] quit
  [*Borderleaf1] commit
  ```
- 配置Borderleaf2。由于Borderleaf1和Borderleaf2为双活网关，请确保这两台设备上配置的NVE接口的IP地址和MAC地址相同。
  ```
  [~Borderleaf2] interface nve 1
  [*Borderleaf2-Nve1] source 1.1.1.110
  [*Borderleaf2-Nve1] vni 5010 head-end peer-list protocol bgp
  [*Borderleaf2-Nve1] mac-address 0000-5e00-0101
  [*Borderleaf2-Nve1] quit
  [*Borderleaf2] commit
  ```
- 配置Leaf1。
  ```
  [~Leaf1] interface nve 1
  [*Leaf1-Nve1] source 2.2.2.210
  [*Leaf1-Nve1] mac-address 0000-5e00-0102
  [*Leaf1-Nve1] vni 5010 head-end peer-list protocol bgp
  [*Leaf1-Nve1] quit
  [*Leaf1] commit
  ```
- 配置Leaf2。由于Leaf1和Leaf2为双活网关，请确保这两台设备上配置的NVE接口的IP地址和MAC地址相同。
  ```
  [~Leaf2] interface nve 1
  [*Leaf2-Nve1] source 2.2.2.210
  [*Leaf2-Nve1] mac-address 0000-5e00-0102
  [*Leaf2-Nve1] vni 5010 head-end peer-list protocol bgp
  [*Leaf2-Nve1] quit
  [*Leaf2] commit
  ```

在Leaf1和Leaf2上配置VXLAN三层网关。

- 在Leaf1上配置VXLAN三层网关。
  ```
  [~Leaf1] interface vbdif 10
  [*Leaf1-Vbdif10] ip binding vpn-instance mcast1
  [*Leaf1-Vbdif10] ip address 192.168.10.1 24
  [*Leaf1-Vbdif10] mac-address 0000-5e00-0104
  [*Leaf1-Vbdif10] vxlan anycast-gateway enable
  [*Leaf1-Vbdif10] arp collect host enable
  [*Leaf1-Vbdif10] quit
  [*Leaf1] commit
  ```
- 在Leaf2上配置VXLAN三层网关。由于Leaf1和Leaf2为双活网关，请确保这两台设备上配置的VBDIF接口IP地址和MAC地址都相同。
  ```
  [~Leaf2] interface vbdif 10
  [*Leaf2-Vbdif10] ip binding vpn-instance mcast1
  [*Leaf2-Vbdif10] ip address 192.168.10.1 24
  [*Leaf2-Vbdif10] mac-address 0000-5e00-0104
  [*Leaf2-Vbdif10] vxlan anycast-gateway enable
  [*Leaf2-Vbdif10] arp collect host enable
  [*Leaf2-Vbdif10] quit
  [*Leaf2] commit
  ```

在M-LAG设备中配置静态Bypass VXLAN隧道。

在M-LAG双归接入VXLAN的场景中，当下行一条链路发生故障时，业务流量需绕行M-LAG设备之间的Peer-link链路。因此，在该场景下M-LAG设备之间必须配置静态Bypass VXLAN隧道，将绕行的业务流量引导至Peer-link链路上。

下面以Leaf1和Leaf2配置为例，Bordleaf1和Borderleaf2的配置与之类似，这里不再赘述，具体配置请参考配置脚本。

- 配置Leaf1。

```
[~Leaf1] interface loopback3
[*Leaf1-LoopBack3] ip address 10.10.30.10 32
[*Leaf1-LoopBack3] quit
[*Leaf1] vlan 100  //本VLAN不能划分给其他业务使用，本例中以100举例
[*Leaf1-vlan100] m-lag peer-link reserved  //仅允许peer-link加入到该VLAN
[*Leaf1-vlan100] quit
[*Leaf1] interface vlanif 100
[*Leaf1-Vlanif100] reserved for vxlan bypass  //指定peer-link接口上VLANIF的IPv4地址只给Bypass VXLAN隧道使用
[*Leaf1-Vlanif100] ip address 10.9.1.1 24  //配置静态Bypass VXLAN隧道的源端IPv4地址
[*Leaf1-Vlanif100] quit
[*Leaf1] ip route-static 10.10.40.10 32 10.9.1.2 preference 1  //配置静态路由，打通Bypass VXLAN隧道
[*Leaf1] interface nve 1
[*Leaf1-Nve1] pip-source 10.10.30.10 peer 10.10.40.10 bypass  //创建静态Bypass VXLAN隧道，指定源端地址和对端地址
[*Leaf1-Nve1] quit
[*Leaf1] commit
```

- 配置Leaf2。

```
[~Leaf2] interface loopback3
[*Leaf2-LoopBack3] ip address 10.10.40.10 32
[*Leaf2-LoopBack3] quit
[*Leaf2] vlan 100
[*Leaf2-vlan100] m-lag peer-link reserved
[*Leaf2-vlan100] quit
[*Leaf2] interface vlanif 100
[*Leaf2-Vlanif100] reserved for vxlan bypass
[*Leaf2-Vlanif100] ip address 10.9.1.2 24
[*Leaf2-Vlanif100] quit
[*Leaf2] ip route-static 10.10.30.10 32 10.9.1.1 preference 1
[*Leaf2] interface nve 1
[*Leaf2-Nve1] pip-source 10.10.40.10 peer 10.10.30.10 bypass
[*Leaf2-Nve1] quit
[*Leaf2] commit
```

在各Leaf上配置BGP对邻居发布IRB类型的路由。其他Leaf上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] bgp 100
[~Borderleaf1-bgp] l2vpn-family evpn
[~Borderleaf1-bgp-af-evpn] peer 2.2.2.2 advertise irb
[*Borderleaf1-bgp-af-evpn] peer 3.3.3.3 advertise irb
[*Borderleaf1-bgp-af-evpn] peer 4.4.4.4 advertise irb
[*Borderleaf1-bgp-af-evpn] quit
[*Borderleaf-bgp] quit
[*Borderleaf] commit
```

在Borderleaf1上配置BGP对邻居发布IP前缀类型的路由。

```
[~Borderleaf1] bgp 100
[~Borderleaf1-bgp] ipv4-family vpn-instance mcast1
[*Borderleaf1-bgp-mcast1] advertise l2vpn evpn
[*Borderleaf1-bgp-mcast1] network 1.1.1.10 32
[*Borderleaf1-bgp-mcast1] quit
[*Borderleaf1-bgp] quit
[*Borderleaf1] commit
```

在各Leaf和Spine上配置公网三层组播。Borderleaf1、Borderleaf2、Leaf1、Leaf2、Spine2上的配置过程与Spine1上的配置类似，配置过程略，详见配置脚本。

```
[~Spine1] multicast routing-enable
[*Spine1] interface loopback 1
[*Spine1-LoopBack1] pim sm
[*Spine1-LoopBack1] quit
[*Spine1] interface 100GE 1/0/1
[*Spine1-100GE1/0/1] pim sm
[*Spine1-100GE1/0/1] quit
```

```
[*Spine1] interface 100GE 1/0/2
[*Spine1-100GE1/0/2] pim sm
[*Spine1-100GE1/0/2] quit
[*Spine1] interface 100GE 1/0/3
[*Spine1-100GE1/0/3] pim sm
[*Spine1-100GE1/0/3] quit
[*Spine1] interface 100GE 1/0/4
[*Spine1-100GE1/0/4] pim sm
[*Spine1-100GE1/0/4] quit
[*Spine1] pim
[*Spine1-pim] static-rp 5.5.5.5
[*Spine1-pim] quit
[*Spine1] commit
```

配置Spine1和Spine2的Loopback1接口为Anycast RP，并配置Spine1和Spine2的Loopback2接口为各自的Anycast RP本地地址，Spine2上的配置过程与Spine1上的配置类似，配置过程略，详见配置脚本。

```
[~Spine1] pim
[~Spine1-pim] anycast-rp 5.5.5.5
[*Spine1-pim-anycast-rp-5.5.5.5] local-address 5.5.5.1
[*Spine1-pim-anycast-rp-5.5.5.5] commit
[~Spine1-pim-anycast-rp-5.5.5.5] peer 5.5.5.2
[*Spine1-pim-anycast-rp-5.5.5.5] quit
[*Spine1-pim] quit
[*Spine1] commit
```

对Borderleaf1和Borderleaf2上的三层链路使能PIM协议。Borderleaf2上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] interface 100GE 1/0/6
[~Borderleaf1-100GE1/0/6] pim sm
[~Borderleaf1-100GE1/0/6] quit
[*Borderleaf1] commit
[~Borderleaf1] interface 100GE 1/0/6.1
[~Borderleaf1-100GE1/0/6.1] dot1q termination vid 1  //配置三层子接口对Dot1q报文的终结功能，请根据报文携带的VLAN Tag值进行配置
[~Borderleaf1-100GE1/0/6.1] pim sm
[~Borderleaf1-100GE1/0/6.1] quit
[*Borderleaf1] commit
```

在各Leaf上针对L3VPN实例下的三层VNI配置BUM组播复制。其他Leaf上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] interface nve 1
[~Borderleaf1-Nve1] vni 5010 mcast-group 225.0.0.1
[*Borderleaf1-Nve1] quit
[*Borderleaf1] commit
```

在Borderleaf1和Borderleaf2上配置VLANIF接口加入组播组，使得M-LAG主备设备之间通过Peer-link同步VXLAN封装的组播报文。Borderleaf2上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] vlan 3000
[*Borderleaf1-vlan3000] quit
[*Borderleaf1] interface vlanif 3000
[*Borderleaf1-Vlanif3000] ip address 10.10.21.1 24
[*Borderleaf1-Vlanif3000] vxlan multicast-group member enable
[*Borderleaf1-Vlanif3000] pim sm
[*Borderleaf1-Vlanif3000] quit
[*Borderleaf1] commit
```

在Leaf1和Leaf2上配置VLANIF接口加入组播组，使得M-LAG主备设备之间通过Peer-link同步VXLAN封装的组播报文。Leaf2上的配置过程与Leaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Leaf1] vlan 4000
[*Leaf1-vlan4000] quit
[*Leaf1] interface vlanif 4000
```

```
[*Leaf1-Vlanif4000] ip address 10.10.20.1 24
[*Leaf1-Vlanif4000] vxlan multicast-group member enable
[*Leaf1-Vlanif4000] pim sm
[*Leaf1-Vlanif4000] quit
[*Leaf1] commit
```

上述配置完成后，各Leaf之间可以成功建立VXLAN隧道。执行命令**display vxlan tunnel**可查看到VXLAN隧道的信息。以Borderleaf1上的显示为例，可以看到“State”字段显示为**up**，表示VXLAN隧道可达；“Type”字段显示为**static**表示目的IP地址是通过静态方式配置的，显示为**dynamic**表示目的IP地址是通过协议动态学习到的。

```
[~Borderleaf1] display vxlan tunnel
Number of vxlan tunnel : 2
Tunnel ID   Source            Destination         State  Type     Uptime
--------------------------------------------------------------------------------
4026531841  1.1.1.110         225.0.0.1             up     static   04:44:52
4026531845  1.1.1.110         2.2.2.210             up     dynamic  04:40:52
```

步骤5 在各Leaf上配置BGP MVPN邻居。

配置Borderleaf1。其他Leaf上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] bgp 100
[~Borderleaf1-bgp] ipv4-family mvpn
[*Borderleaf1-bgp-af-mvpn] peer 2.2.2.2 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Borderleaf1-bgp-af-mvpn] peer 3.3.3.3 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Borderleaf1-bgp-af-mvpn] peer 4.4.4.4 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Borderleaf1-bgp-af-mvpn] quit
[*Borderleaf1-bgp] quit
[*Borderleaf1] commit
```

步骤6 在各Leaf上配置VXLAN类型的I-PMSI隧道。

配置Borderleaf1。在VXLAN分布式网关上配置的MVPN ID必须是当前网关设备上的VTEP IP地址。

```
[~Borderleaf1] multicast mvpn 1.1.1.110
[*Borderleaf1] ip vpn-instance mcast1
[*Borderleaf1-vpn-instance-mcast1] ipv4-family
[*Borderleaf1-vpn-instance-mcast1-af-ipv4] multicast routing-enable
[*Borderleaf1-vpn-instance-mcast1-af-ipv4] multicast mvpn route-import local-admin-id 1
[*Borderleaf1-vpn-instance-mcast1-af-ipv4] mvpn
[*Borderleaf1-vpn-instance-mcast1-af-ipv4-mvpn] c-multicast signaling bgp
[*Borderleaf1-vpn-instance-mcast1-af-ipv4-mvpn] spt-only mode
[*Borderleaf1-vpn-instance-mcast1-af-ipv4-mvpn] ipmsi-tunnel
[*Borderleaf1-vpn-instance-mcast1-af-ipv4-mvpn-ipmsi] vxlan static
[*Borderleaf1-vpn-instance-mcast1-af-ipv4-mvpn-ipmsi] quit
[*Borderleaf1-vpn-instance-mcast1-af-ipv4-mvpn] quit
[*Borderleaf1-vpn-instance-mcast1-af-ipv4] quit
[*Borderleaf1-vpn-instance-mcast1] quit
[*Borderleaf1] commit
```

配置Borderleaf2。由于Borderleaf1和Borderleaf2为双活网关，请确保这两台设备上配置的MVPN ID相同，且均为设备的VTEP IP地址。

```
[~Borderleaf2] multicast mvpn 1.1.1.110
[*Borderleaf2] ip vpn-instance mcast1
[*Borderleaf2-vpn-instance-mcast1] ipv4-family
[*Borderleaf2-vpn-instance-mcast1-af-ipv4] multicast routing-enable
[*Borderleaf2-vpn-instance-mcast1-af-ipv4] multicast mvpn route-import local-admin-id 1
[*Borderleaf2-vpn-instance-mcast1-af-ipv4] mvpn
[*Borderleaf2-vpn-instance-mcast1-af-ipv4-mvpn] c-multicast signaling bgp
[*Borderleaf2-vpn-instance-mcast1-af-ipv4-mvpn] spt-only mode
```

```
[*Borderleaf2-vpn-instance-mcast1-af-ipv4-mvpn] ipmsi-tunnel
[*Borderleaf2-vpn-instance-mcast1-af-ipv4-mvpn-ipmsi] vxlan static
[*Borderleaf2-vpn-instance-mcast1-af-ipv4-mvpn-ipmsi] quit
[*Borderleaf2-vpn-instance-mcast1-af-ipv4-mvpn] quit
[*Borderleaf2-vpn-instance-mcast1-af-ipv4] quit
[*Borderleaf2-vpn-instance-mcast1] quit
[*Borderleaf2] commit
```

配置Leaf1。

```
[~Leaf1] multicast mvpn 2.2.2.210
[*Leaf1] ip vpn-instance mcast1
[*Leaf1-vpn-instance-mcast1] ipv4-family
[*Leaf1-vpn-instance-mcast1-af-ipv4] multicast routing-enable
[*Leaf1-vpn-instance-mcast1-af-ipv4] multicast mvpn route-import local-admin-id 1
[*Leaf1-vpn-instance-mcast1-af-ipv4] mvpn
[*Leaf1-vpn-instance-mcast1-af-ipv4-mvpn] c-multicast signaling bgp
[*Leaf1-vpn-instance-mcast1-af-ipv4-mvpn] spt-only mode
[*Leaf1-vpn-instance-mcast1-af-ipv4-mvpn] ipmsi-tunnel
[*Leaf1-vpn-instance-mcast1-af-ipv4-mvpn-ipmsi] vxlan static
[*Leaf1-vpn-instance-mcast1-af-ipv4-mvpn-ipmsi] quit
[*Leaf1-vpn-instance-mcast1-af-ipv4-mvpn] quit
[*Leaf1-vpn-instance-mcast1-af-ipv4] quit
[*Leaf1-vpn-instance-mcast1] quit
[*Leaf1] commit
```

配置Leaf2。由于Leaf1和Leaf2为双活网关，请确保这两台设备上配置的MVPN ID相同，且均为设备的VTEP IP地址。

```
[~Leaf2] multicast mvpn 2.2.2.210
[*Leaf2] ip vpn-instance mcast1
[*Leaf2-vpn-instance-mcast1] ipv4-family
[*Leaf2-vpn-instance-mcast1-af-ipv4] multicast routing-enable
[*Leaf2-vpn-instance-mcast1-af-ipv4] multicast mvpn route-import local-admin-id 1
[*Leaf2-vpn-instance-mcast1-af-ipv4] mvpn
[*Leaf2-vpn-instance-mcast1-af-ipv4-mvpn] c-multicast signaling bgp
[*Leaf2-vpn-instance-mcast1-af-ipv4-mvpn] spt-only mode
[*Leaf2-vpn-instance-mcast1-af-ipv4-mvpn] ipmsi-tunnel
[*Leaf2-vpn-instance-mcast1-af-ipv4-mvpn-ipmsi] vxlan static
[*Leaf2-vpn-instance-mcast1-af-ipv4-mvpn-ipmsi] quit
[*Leaf2-vpn-instance-mcast1-af-ipv4-mvpn] quit
[*Leaf2-vpn-instance-mcast1-af-ipv4] quit
[*Leaf2-vpn-instance-mcast1] quit
[*Leaf2] commit
```

步骤7 在VBDIF接口下配置PIM SM和IGMP。

配置Leaf1。

```
[~Leaf1] interface vbdif 10
[~Leaf1-Vbdif10] pim sm
[*Leaf1-Vbdif10] igmp enable
[*Leaf1-Vbdif10] quit
[*Leaf1] commit
```

配置Leaf2。

```
[~Leaf2] interface vbdif 10
[~Leaf2-Vbdif10] pim sm
[*Leaf2-Vbdif10] igmp enable
[*Leaf2-Vbdif10] quit
[*Leaf2] commit
```

步骤8 在各Leaf上配置私网静态RP。

配置Borderleaf1。

```
[~Borderleaf1] interface loopback 3
[~Borderleaf1-LoopBack2] ip binding vpn-instance mcast1
```



```
[*Borderleaf1-LoopBack2] ip address 1.1.1.10 32
[*Borderleaf1-LoopBack2] pim sm
[*Borderleaf1-LoopBack2] quit
[*Borderleaf1] pim vpn-instance mcast1
[*Borderleaf1-pim-mcast1] static-rp 1.1.1.10
[*Borderleaf1-pim-mcast1] quit
[*Borderleaf1] commit
```

配置Borderleaf2。

```
[*Borderleaf2] pim vpn-instance mcast1
[*Borderleaf2-pim-mcast1] static-rp 1.1.1.10
[*Borderleaf2-pim-mcast1] quit
[*Borderleaf2] commit
```

配置Leaf1。

```
[*Leaf1] pim vpn-instance mcast1
[*Leaf1-pim-mcast1] static-rp 1.1.1.10
[*Leaf1-pim-mcast1] quit
[*Leaf1] commit
```

配置Leaf2。

```
[*Leaf2] pim vpn-instance mcast1
[*Leaf2-pim-mcast1] static-rp 1.1.1.10
[*Leaf2-pim-mcast1] quit
[*Leaf2] commit
```

步骤9 在各Leaf上配置Monitor Link关联上行接口和下行接口。

配置Leaf1。

```
[~Leaf1] monitor-link group 1
[*Leaf1-mtlk-group1] port 100GE 1/0/1 uplink
[*Leaf1-mtlk-group1] port 100GE 1/0/6 uplink
[*Leaf1-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf1-mtlk-group1] quit
[*Leaf1] commit
```

配置Leaf2。

```
[~Leaf2] monitor-link group 1
[*Leaf2-mtlk-group1] port 100GE 1/0/1 uplink
[*Leaf2-mtlk-group1] port 100GE 1/0/6 uplink
[*Leaf2-mtlk-group1] port eth-trunk 10 downlink 1
[*Leaf2-mtlk-group1] quit
[*Leaf2] commit
```

步骤10 在Borderleaf1、Borderleaf2和CE之间配置私网三层组播。

配置Borderleaf1，Borderleaf2上的配置过程与Borderleaf1上的配置类似，配置过程略，详见配置脚本。

```
[~Borderleaf1] interface 100GE 1/0/2
[*Borderleaf1-100GE1/0/2] undo portswitch
[*Borderleaf1-100GE1/0/2] ip binding vpn-instance mcast1
[*Borderleaf1-100GE1/0/2] ip address 192.168.20.1 24
[*Borderleaf1-100GE1/0/2] pim sm
[*Borderleaf1-100GE1/0/2] igmp enable
[*Borderleaf1-100GE1/0/2] quit
[*Borderleaf1] ip route-static vpn-instance mcast1 192.168.10.2 24 100GE 1/0/2
[*Borderleaf1] bgp 100
[*Borderleaf1-bgp] ipv4-family vpn-instance mcast1
[*Borderleaf1-bgp-mcast1] import-route static
[*Borderleaf1-bgp-mcast1] quit
[*Borderleaf1-bgp] quit
[*Borderleaf1] commit
```

配置CE。


```
[~CE] multicast routing-enable
[*CE] interface 100GE 1/0/1
[*CE-100GE1/0/1] pim sm
[*CE-100GE1/0/1] igmp enable
[*CE-100GE1/0/1] quit
[*CE] interface 100GE 1/0/2
[*CE-100GE1/0/2] pim sm
[*CE-100GE1/0/2] igmp enable
[*CE-100GE1/0/2] quit
[*CE] interface 100GE 1/0/3
[*CE-100GE1/0/3] pim sm
[*CE-100GE1/0/3] igmp enable
[*CE-100GE1/0/3] quit
[*CE] ip route-static 192.168.10.1 24 100GE 1/0/2
[*CE] commit
```

----结束

验证配置结果

在各Leaf上通过命令**display bgp mvpn all peer**，可以查看到BGP MVPN邻居信息。例如Borderleaf1上的显示信息如下：

```
[~Borderleaf1] display bgp mvpn all peer
 BGP local router ID        : 1.1.1.1
 Local AS number            : 100
 Total number of peers      : 3
 Peers in established state : 3

 Peer          V         AS  MsgRcvd  MsgSent  OutQ  Up/Down       State  PrefRcv
 2.2.2.2       4        100     1860     1859     0 04:43:30 Established        1
 3.3.3.3       4        100     3219     3221     0 04:43:39 Established        1
 4.4.4.4       4        100     4559     4561     0 04:44:39 Established        1
```

在接收者发起加入组播组之后，在各Leaf上通过命令**display pim vpn-instance mcast1 routing-table**，可以查看到私网PIM路由表信息。

- 在组播源侧Borderleaf1上可以看到，Receiver通过BGP加入。

  ```
  [~Borderleaf1] display pim vpn-instance mcast1 routing-table
   VPN-Instance: mcast1
   Total 1 (*, G) entry; 1 (S, G) entry

   (*, 225.1.1.1)
      RP: 1.1.1.10 (local)
      Protocol: pim-sm, Flag: WC
      UpTime: 04:43:57
      Upstream interface: 100GE1/0/2
          Upstream neighbor: 192.168.20.2
          RPF prime neighbor: 192.168.20.2
      Downstream interface(s) information:
      Total number of downstreams: 1
        1: pseudo
            Protocol: BGP, UpTime: 04:43:57, Expires: -

   (192.168.20.9, 225.1.1.1)
      RP: 1.1.1.10 (local)
      Protocol: pim-sm, Flag: SPT LOC ACT
      UpTime: 04:43:57
      Upstream interface: 100GE1/0/2
          Upstream neighbor: 192.168.20.2
          RPF prime neighbor: 192.168.20.2
      Downstream interface(s) information:
      Total number of downstreams: 1
        1: pseudo
            Protocol: pim-sm, UpTime: 04:43:57, Expires: -
  ```

- 在Leaf1上可以看到，接收者Receiver对应的PIM私网路由表项上游为Through-BGP，下游出接口为VBDIF10。


```
[~Leaf1] display pim vpn-instance mcast1 routing-table
 VPN-Instance: mcast1
 Total 1 (S, G) entry; 1 (S, G) entry

 (*, 225.1.1.1)
     RP: 1.1.1.10
     Protocol: pim-sm, Flag: WC
     UpTime: 04:44:18
     Upstream interface: through-BGP
         Upstream neighbor: 1.1.1.1
         RPF prime neighbor: 1.1.1.1
     Downstream interface(s) information:
     Total number of downstreams: 1
       1: Vbdif10
            Protocol: igmp, UpTime: 04:44:18, Expires: -

 (192.168.20.9, 225.1.1.1)
     RP: 1.1.1.10
     Protocol: pim-sm, Flag: SPT ACT
     UpTime: 04:44:18
     Upstream interface: through-BGP
         Upstream neighbor: 1.1.1.1
         RPF prime neighbor: 1.1.1.1
     Downstream interface(s) information:
     Total number of downstreams: 1
       1: Vbdif10
            Protocol: pim-sm, UpTime: 04:44:18, Expires: -
```

配置脚本

- CE的配置脚本

```
#
sysname CE
#
multicast routing-enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.2 255.255.255.0
 pim sm
 igmp enable
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.20.2 255.255.255.0
 pim sm
 igmp enable
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.30.2 255.255.255.0
 pim sm
 igmp enable
#
ip route-static 192.168.10.0 255.255.255.0 100GE 1/0/2
#
return
```

- Borderleaf1的配置脚本

```
#
sysname Borderleaf1
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 10.10.11.1 peer 10.10.11.2
#
vlan batch 3000
```

```
#
VLAN 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
multicast routing-enable
#
multicast mvpn 1.1.1.110
#
ip vpn-instance mcast1
 ipv4-family
  route-distinguisher 1:1
  vpn-target 1:1 export-extcommunity
  vpn-target 13:1 export-extcommunity evpn
  vpn-target 1:1 import-extcommunity
  vpn-target 13:1 import-extcommunity evpn
  multicast routing-enable
  multicast mvpn route-import local-admin-id 1
  mvpn
   c-multicast signaling bgp
   spt-only mode
   ipmsi-tunnel
    vxlan static
 vxlan vni 5010
#
interface Vlanif100
 ip address 10.8.1.1 255.255.255.0
 reserved for vxlan bypass
#
interface Vlanif3000
 ip address 10.10.21.1 255.255.255.0
 pim sm
 vxlan multicast-group member enable
#
interface MEth0/0/0
 ip address 10.10.11.1 255.255.255.0
#
interface Eth-Trunk2
 mode lacp-static
 peer-link 1
#
ip route-static vpn-instance mcast1 192.168.10.2 24 100GE1/0/2
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.1 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip binding vpn-instance mcast1
 ip address 192.168.20.1 24
 pim sm
 igmp enable
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.1.11.1 255.255.255.0
 pim sm
#
interface 100GE1/0/4
 eth-trunk 2
#
interface 100GE1/0/5
 eth-trunk 2
```

```
#
interface 100GE1/0/6
 undo portswitch
 pim sm
#
interface 100GE1/0/6.1
 ip address 10.6.1.1 255.255.255.0
 dot1q termination vid 1
 pim sm
#
interface LoopBack1
 ip address 1.1.1.1 255.255.255.255
 pim sm
#
interface LoopBack2
 ip address 1.1.1.110 255.255.255.255
 pim sm
#
interface LoopBack3
 ip binding vpn-instance mcast1
 ip address 1.1.1.10 255.255.255.255
 pim sm
#
interface LoopBack4
 ip address 10.10.10.10 255.255.255.255
#
interface Nve1
 source 1.1.1.110
 pip-source 10.10.10.10 peer 10.10.20.10 bypass
 vni 5010 head-end peer-list protocol bgp
 vni 5010 mcast-group 225.0.0.1
 mac-address 0000-5e00-0101
#
bgp 100
 router-id 1.1.1.1
 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
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack1
 #
 ipv4-family unicast
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
  peer 4.4.4.4 enable
 #
 ipv4-family mvpn
  policy vpn-target
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
  peer 4.4.4.4 enable
 #
 ipv4-family vpn-instance mcast1
  network 1.1.1.10 255.255.255.255
  import-route static
  advertise l2vpn evpn
 #
 l2vpn-family evpn
  policy vpn-target
  peer 2.2.2.2 enable
  peer 2.2.2.2 advertise irb
  peer 3.3.3.3 enable
  peer 3.3.3.3 advertise irb
  peer 4.4.4.4 enable
  peer 4.4.4.4 advertise irb
#
ospf 1
 area 0.0.0.0
```

```
 network 1.1.1.1 0.0.0.0
 network 1.1.1.110 0.0.0.0
 network 10.1.1.0 0.0.0.255
 network 10.1.11.0 0.0.0.255
 network 10.6.1.0 0.0.0.255
 network 10.10.11.0 0.0.0.255
#
ip route-static 10.10.20.10 32 10.8.1.2 preference 1
#
pim
 static-rp 5.5.5.5
#
pim vpn-instance mcast1
 static-rp 1.1.1.10
#
return
```

- Borderleaf2的配置脚本

```
#
sysname Borderleaf2
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 10.10.11.2 peer 10.10.11.1
#
vlan batch 3000
#
VLAN 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
multicast routing-enable
#
multicast mvpn 1.1.1.110
#
ip vpn-instance mcast1
 ipv4-family
  route-distinguisher 1:1
  vpn-target 1:1 export-extcommunity
  vpn-target 13:1 export-extcommunity evpn
  vpn-target 1:1 import-extcommunity
  vpn-target 13:1 import-extcommunity evpn
  multicast routing-enable
  multicast mvpn route-import local-admin-id 1
  mvpn
   c-multicast signaling bgp
   spt-only mode
   ipmsi-tunnel
    vxlan static
 vxlan vni 5010
#
interface Vlanif100
 ip address 10.8.1.2 255.255.255.0
 reserved for vxlan bypass
#
interface Vlanif3000
 ip address 10.10.21.2 255.255.255.0
 pim sm
 vxlan multicast-group member enable
#
interface MEth0/0/0
 ip address 10.10.11.2 255.255.255.0
#
interface Eth-Trunk2
 mode lacp-static
```



```
 peer-link 1
#
ip route-static vpn-instance mcast1 192.168.10.2 24 100GE1/0/2
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.44.1 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip binding vpn-instance mcast1
 ip address 192.168.30.1 24
 pim sm
 igmp enable
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.1.4.1 255.255.255.0
 pim sm
#
interface 100GE1/0/4
 eth-trunk 2
#
interface 100GE1/0/5
 eth-trunk 2
#
interface 100GE1/0/6
 undo portswitch
 pim sm
#
interface 100GE1/0/6.1
 ip address 10.6.1.2 255.255.255.0
 dot1q termination vid 1
 pim sm
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
 pim sm
#
interface LoopBack2
 ip address 1.1.1.110 255.255.255.255
 pim sm
#
interface LoopBack3
 ip address 10.10.20.10 255.255.255.255
#
interface Nve1
 source 1.1.1.110
 pip-source 10.10.20.10 peer 10.10.10.10 bypass
 vni 5010 head-end peer-list protocol bgp
 vni 5010 mcast-group 225.0.0.1
 mac-address 0000-5e00-0101
#
bgp 100
 router-id 2.2.2.2
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack1
 peer 3.3.3.3 as-number 100
 peer 3.3.3.3 connect-interface LoopBack1
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack1
 #
 ipv4-family unicast
  peer 1.1.1.1 enable
  peer 3.3.3.3 enable
  peer 4.4.4.4 enable
 #
 ipv4-family mvpn
```

```
 policy vpn-target
 peer 1.1.1.1 enable
 peer 3.3.3.3 enable
 peer 4.4.4.4 enable
#
ipv4-family vpn-instance mcast1
 import-route static
#
l2vpn-family evpn
 policy vpn-target
 peer 1.1.1.1 enable
 peer 1.1.1.1 advertise irb
 peer 3.3.3.3 enable
 peer 3.3.3.3 advertise irb
 peer 4.4.4.4 enable
 peer 4.4.4.4 advertise irb
#
ospf 1
 area 0.0.0.0
  network 2.2.2.2 0.0.0.0
  network 1.1.1.110 0.0.0.0
  network 10.1.4.0 0.0.0.255
  network 10.1.44.0 0.0.0.255
  network 10.6.1.0 0.0.0.255
  network 10.10.11.0 0.0.0.255
#
ip route-static 10.10.10.10 32 10.8.1.1 preference 1
#
pim
 static-rp 5.5.5.5
#
pim vpn-instance mcast1
 static-rp 1.1.1.10
#
return
```

- Leaf1的配置文件

```
#
sysname Leaf1
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!%+%#
 dual-active detection source ip 10.10.10.1 peer 10.10.10.2
#
vlan batch 4000
#
VLAN 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
multicast routing-enable
#
multicast mvpn 2.2.2.210
#
ip vpn-instance mcast1
 ipv4-family
  route-distinguisher 1:1
  vpn-target 1:1 export-extcommunity
  vpn-target 13:1 export-extcommunity evpn
  vpn-target 1:1 import-extcommunity
  vpn-target 13:1 import-extcommunity evpn
  multicast routing-enable
  multicast mvpn route-import local-admin-id 1
  mvpn
   c-multicast signaling bgp
```



```
  spt-only mode
  ipmsi-tunnel
   vxlan static
 vxlan vni 5010
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 11:1
  vpn-target 12:1 export-extcommunity
  vpn-target 13:1 export-extcommunity
  vpn-target 12:1 import-extcommunity
  vpn-target 13:1 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance mcast1
 ip address 192.168.10.1 255.255.255.0
 mac-address 0000-5e00-0104
 pim sm
 igmp enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif100
 ip address 10.9.1.1 255.255.255.0
 reserved for vxlan bypass
#
interface Vlanif4000
 ip address 10.10.20.1 255.255.255.0
 pim sm
 vxlan multicast-group member enable
#
interface MEth0/0/0
 ip address 10.10.10.1 255.255.255.0
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk10.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.2.1 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 eth-trunk 10
#
interface 100GE1/0/3
 eth-trunk 10
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 1
#
interface 100GE1/0/6
 undo portswitch
 ip address 10.1.22.1 255.255.255.0
 pim sm
```

```
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.255
 pim sm
#
interface LoopBack2
 ip address 2.2.2.210 255.255.255.255
 pim sm
#
interface LoopBack3
 ip address 10.10.30.10 255.255.255.255
#
interface Nve1
 source 2.2.2.210
 pip-source 10.10.30.10 peer 10.10.40.10 bypass
 vni 5010 head-end peer-list protocol bgp
 vni 5010 mcast-group 225.0.0.1
 mac-address 0000-5e00-0102
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/6 uplink
 port Eth-Trunk10 downlink 1
#
bgp 100
 router-id 3.3.3.3
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack1
 peer 2.2.2.2 as-number 100
 peer 2.2.2.2 connect-interface LoopBack1
 peer 4.4.4.4 as-number 100
 peer 4.4.4.4 connect-interface LoopBack1
 #
 ipv4-family unicast
  peer 1.1.1.1 enable
  peer 2.2.2.2 enable
  peer 4.4.4.4 enable
 #
 ipv4-family mvpn
  policy vpn-target
  peer 1.1.1.1 enable
  peer 2.2.2.2 enable
  peer 4.4.4.4 enable
 #
 l2vpn-family evpn
  policy vpn-target
  peer 1.1.1.1 enable
  peer 1.1.1.1 advertise irb
  peer 2.2.2.2 enable
  peer 2.2.2.2 advertise irb
  peer 4.4.4.4 enable
  peer 4.4.4.4 advertise irb
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 2.2.2.210 0.0.0.0
  network 10.1.2.0 0.0.0.255
  network 10.1.22.0 0.0.0.255
  network 10.10.10.0 0.0.0.255
#
ip route-static 10.10.40.10 32 10.9.1.2 preference 1
#
pim
 static-rp 5.5.5.5
#
pim vpn-instance mcast1
 static-rp 1.1.1.10
```

```
#
return
```

- Leaf2的配置文件

```
#
sysname Leaf2
#
dfs-group 1
 authentication-mode hmac-sha256 password %+%##!!!!!!!!!"!!!!"!!!!*!!!!C+tR0CW9x*eB&pWp`t),Azgwh
\o8#4LZPD!!!!!!!!!!!!!!!9!!!!>fwJ)I0E{=:%,*,XRhbH&t0MCy_8=7!!!!!!!!!!%+%#
 dual-active detection source ip 10.10.10.2 peer 10.10.10.1
#
vlan batch 4000
#
VLAN 100
 m-lag peer-link reserved
#
stp mode rstp
stp v-stp enable
#
evpn-overlay enable
#
multicast routing-enable
#
multicast mvpn 2.2.2.210
#
ip vpn-instance mcast1
 ipv4-family
  route-distinguisher 1:1
  vpn-target 1:1 export-extcommunity
  vpn-target 13:1 export-extcommunity evpn
  vpn-target 1:1 import-extcommunity
  vpn-target 13:1 import-extcommunity evpn
  multicast routing-enable
  multicast mvpn route-import local-admin-id 1
  mvpn
   c-multicast signaling bgp
   spt-only mode
   ipmsi-tunnel
    vxlan static
 vxlan vni 5010
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 11:1
  vpn-target 12:1 export-extcommunity
  vpn-target 13:1 export-extcommunity
  vpn-target 12:1 import-extcommunity
  vpn-target 13:1 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance mcast1
 ip address 192.168.10.1 255.255.255.0
 mac-address 0000-5e00-0104
 pim sm
 igmp enable
 vxlan anycast-gateway enable
 arp collect host enable
#
interface Vlanif4000
 ip address 10.10.20.2 255.255.255.0
 pim sm
 vxlan multicast-group member enable
#
interface Vlanif100
 ip address 10.9.1.2 255.255.255.0
 reserved for vxlan bypass
#
interface MEth0/0/0
```



```
 ip address 10.10.10.2 255.255.255.0
#
interface Eth-Trunk1
 mode lacp-static
 peer-link 1
#
interface Eth-Trunk10
 stp edged-port enable
 mode lacp-static
 dfs-group 1 m-lag 1
#
interface Eth-Trunk10.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.33.1 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 eth-trunk 10
#
interface 100GE1/0/3
 eth-trunk 10
#
interface 100GE1/0/4
 eth-trunk 1
#
interface 100GE1/0/5
 eth-trunk 1
#
interface 100GE1/0/6
 undo portswitch
 ip address 10.1.3.1 255.255.255.0
 pim sm
#
interface LoopBack1
 ip address 4.4.4.4 255.255.255.255
 pim sm
#
interface LoopBack2
 ip address 2.2.2.210 255.255.255.255
 pim sm
#
interface LoopBack3
 ip address 10.10.40.10 255.255.255.255
#
interface Nve1
 source 2.2.2.210
 pip-source 10.10.40.10 peer 10.10.30.10 bypass
 vni 5010 head-end peer-list protocol bgp
 vni 5010 mcast-group 225.0.0.1
 mac-address 0000-5e00-0102
#
monitor-link group 1
 port 100GE1/0/1 uplink
 port 100GE1/0/6 uplink
 port Eth-Trunk10 downlink 1
#
bgp 100
 router-id 4.4.4.4
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack1
 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
  peer 1.1.1.1 enable
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
 #
 ipv4-family mvpn
  policy vpn-target
  peer 1.1.1.1 enable
  peer 2.2.2.2 enable
  peer 3.3.3.3 enable
 #
 l2vpn-family evpn
  policy vpn-target
  peer 1.1.1.1 enable
  peer 1.1.1.1 advertise irb
  peer 2.2.2.2 enable
  peer 2.2.2.2 advertise irb
  peer 3.3.3.3 enable
  peer 3.3.3.3 advertise irb
#
ospf 1
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 2.2.2.210 0.0.0.0
  network 10.1.3.0 0.0.0.255
  network 10.1.33.0 0.0.0.255
  network 10.10.10.0 0.0.0.255
#
ip route-static 10.10.30.10 32 10.9.1.1 preference 1
#
pim
 static-rp 5.5.5.5
#
pim vpn-instance mcast1
 static-rp 1.1.1.10
#
return
```

- Spine1的配置脚本

```
#
sysname Spine1
#
multicast routing-enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.2 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.2.2 255.255.255.0
 pim sm
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.1.3.2 255.255.255.0
 pim sm
#
interface 100GE1/0/4
 undo portswitch
 ip address 10.1.4.2 255.255.255.0
 pim sm
#
interface LoopBack1
 ip address 5.5.5.5 255.255.255.255
 pim sm
#
interface LoopBack2
 ip address 5.5.5.1 255.255.255.255
```

```
 pim sm
#
ospf 1
 area 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 5.5.5.1 0.0.0.0
  network 10.1.1.0 0.0.0.255
  network 10.1.2.0 0.0.0.255
  network 10.1.3.0 0.0.0.255
  network 10.1.4.0 0.0.0.255
#
pim
 static-rp 5.5.5.5
 anycast-rp 5.5.5.5
  local-address 5.5.5.1
  peer 5.5.5.2
#
return
```

- Spine2的配置脚本

```
#
sysname Spine2
#
multicast routing-enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.11.2 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.22.2 255.255.255.0
 pim sm
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.1.33.2 255.255.255.0
 pim sm
#
interface 100GE1/0/4
 undo portswitch
 ip address 10.1.44.2 255.255.255.0
 pim sm
#
interface LoopBack1
 ip address 5.5.5.5 255.255.255.255
 pim sm
#
interface LoopBack2
 ip address 5.5.5.2 255.255.255.255
 pim sm
#
ospf 1
 area 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 5.5.5.2 0.0.0.0
  network 10.1.11.0 0.0.0.255
  network 10.1.22.0 0.0.0.255
  network 10.1.33.0 0.0.0.255
  network 10.1.44.0 0.0.0.255
#
pim
 static-rp 5.5.5.5
 anycast-rp 5.5.5.5
  local-address 5.5.5.2
  peer 5.5.5.1
#
return
```


1.14 配置智能无损网络综合示例

适用产品和版本

安装了P系列单板的CE16800、CE6866、CE6866K、CE8851、CE8851K系列交换机V300R020C00或更高版本。

安装了SAN系列单板的CE16800、CE6860-SAN、CE8850-SAN系列交换机V300R020C10或更高版本。

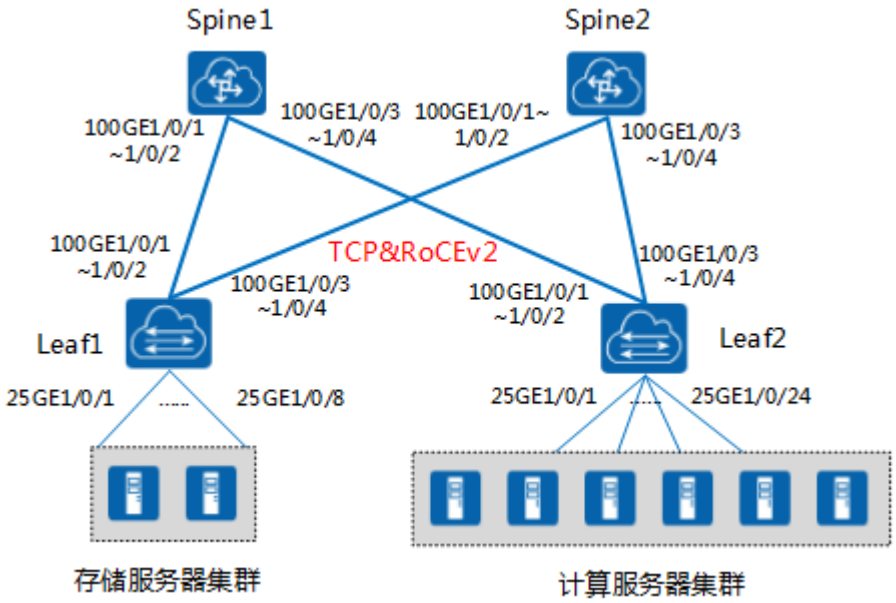
CE6860-HAM、CE8850-HAM系列交换机V300R022C00或更高版本。

如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

组网需求

如**图1-14**所示，为某RoCEv2高性能应用组网，网络中同时存在TCP和RoCEv2流量，服务器均支持RoCEv2协议并开启DCQCN功能。计算和存储服务器均部署在一个PoD中，节点数量按3：1比例部署。Leaf和Spine之间采用100GE链路全互联，服务器使用25GE链路接入Leaf交换机，收敛比为1:1。本示例的设备选型为：Leaf交换机使用CE6866-48S8CQ-P，Spine设备使用CloudEngine 16800（安装CE-MPUE系列主控板）。

图 1-14 智能无损网络组网图



配置思路

说明

- 本示例中配置的参数取值仅为参考，更多内容请参见《CloudFabric数据中心网络解决方案智能无损场景最佳实践》，用户请根据实际组网中的流量模型对每台设备进行配置。
- 本举例默认组网中Spine-Leaf-Server之间的互联IP及路由配置已完成，Server之间路由可达。

采用如下的思路配置：

- 配置Leaf交换机。
  a. 配置PFC优先级流量控制，需要先配置优先级映射。
  b. 配置PFC死锁检测。
  c. 配置PFC死锁预防。
  d. 配置嵌入式AI功能，加载AI ECN组件需要使用的模型文件。
  e. 配置无损队列的AI ECN功能。
- 配置Spine交换机。
  a. 配置PFC优先级流量控制，需要先配置优先级映射。
  b. 配置PFC死锁检测。
  c. 配置嵌入式AI功能，加载AI ECN组件需要使用的模型文件。
  d. 配置无损队列的AI ECN功能。
- 配置服务器网卡（操作步骤略）。
  a. 配置网卡工作在RoCEv2模式。
  b. 配置RoCEv2的建链方式。
  c. 配置网卡信任DSCP模式，并配置RoCEv2报文和CNP报文的DSCP值。
  d. 在网卡上为RoCEv2的优先级使能PFC。
  e. 在网卡上为RoCEv2的优先级使能DCQCN。

操作步骤

配置Leaf1，Leaf2上的配置与Leaf1上的配置类似，配置过程略

步骤1 配置PFC优先级流量控制

1. 配置优先级映射

   # 根据规划，本次示例中设置RoCEv2的DSCP值为24，CNP报文的DSCP值为25，则配置设备中的Diffserv Domain优先级映射模板如下，将RoCEv2的优先级映射为优先级4（走队列4），CNP报文的优先级映射为优先级6（走队列6）。

   ```
   <HUAWEI> system-view
   [~HUAWEI] sysname Leaf1
   [*HUAWEI] commit
   [~Leaf1] diffserv domain ds1
   [*Leaf1-dsdomain-ds1] ip-dscp-inbound 24 phb af4 green
   [*Leaf1-dsdomain-ds1] ip-dscp-inbound 25 phb cs6 green
   [*Leaf1-dsdomain-ds1] quit
   [*Leaf1] port-group server_using
   [*Leaf1-port-group-server_using] group-member 25ge 1/0/1 to 25ge 1/0/8
   [*Leaf1-port-group-server_using] quit
   [*Leaf1] commit
   [~Leaf1] port-group server_using
   [*Leaf1-port-group-server_using] trust dscp
   [*Leaf1-port-group-server_using] trust upstream ds1
   [*Leaf1-port-group-server_using] quit
   [*Leaf1] commit
   ```

2. 为承载RoCEv2流量的优先级配置PFC功能

   #规划使用优先级4来承载网络中的RoCEv2流量，则需要在各个接口下针对优先级4使能PFC，并使能PFC功能基于DSCP映射后的优先级进行反压。

   ```
   [~Leaf1] dcb pfc
   [~Leaf1-dcb-pfc-default] priority 4
   [*Leaf1-dcb-pfc-default] quit
   ```



```
[*Leaf1] port-group spine_using
[*Leaf1-port-group-spine_using] group-member 100ge 1/0/1 to 100ge 1/0/4
[*Leaf1-port-group-spine_using] quit
[*Leaf1] commit
[~Leaf1] port-group spine_using
[*Leaf1-port-spine_using] dcb pfc enable mode manual
[*Leaf1-port-spine_using] qos phb marking dscp enable
[*Leaf1-port-spine_using] quit
[*Leaf1] port-group server_using
[*Leaf1-port-group-server_using] dcb pfc enable mode manual
[*Leaf1-port-group-server_using] quit
[*Leaf1] dcb pfc dscp-mapping enable slot 1  //仅安装了P系列单板、SAN系列单板的CE16800支持本命令，如果用户入方向映射选择信任报文的DSCP映射内部优先级，还需要使用此命令，使能PFC功能基于DSCP映射后的优先级进行反压。
[*Leaf1] commit
```

说明

上述配置完成后，承载网络中的RoCEv2流量的优先级为4的队列即为无损队列。

步骤2 配置PFC死锁检测

配置无损队列的PFC死锁检测周期和恢复周期为100毫秒，并配置设备20s内出现5次PFC死锁时，去使能PFC功能。

```
[~Leaf1] dcb pfc
[*Leaf1-dcb-pfc-default] priority 4 turn-off threshold 5
[*Leaf1-dcb-pfc-default] quit
[*Leaf1] dcb pfc deadlock-detect timer 100
[*Leaf1] dcb pfc deadlock-recovery timer 100
[*Leaf1] commit
```

说明

配置完成后，若需要修改PFC死锁检测的配置，为了保障配置成功，防止设备处于死锁恢复期间，需要执行**shutdown**命令，关闭应用了PFC功能的端口。

步骤3 配置PFC死锁预防

在Leaf1上创建名称为myuplink的PFC上联端口组，并将Leaf1与Spine1、Spine2相连的端口都加入该PFC上联端口组。Leaf2上的配置与Leaf1上的配置类似，配置过程略。

```
[~Leaf1] dcb pfc uplink group myuplink
[*Leaf1-dcb-pfc-uplink-group-myuplink] group-member interface 100ge 1/0/1 to 100ge 1/0/4
[*Leaf1-dcb-pfc-uplink-group-myuplink] quit
[*Leaf1] commit
```

配置PFC上联端口组myuplink，为DSCP值为24的无损队列设置一个无损备份队列5，DSCP值设置为32，和一个有损备份队列2，DSCP值设置为16。

```
[~Leaf1] dcb pfc uplink group myuplink
[~Leaf1-dcb-pfc-uplink-group-myuplink] adjust original-dscp 24 to priority 5 dscp 32
[*Leaf1-dcb-pfc-uplink-group-myuplink] adjust original-dscp 32 to priority 2 dscp 16
[*Leaf1-dcb-pfc-uplink-group-myuplink] quit
[*Leaf1] commit
```

步骤4 配置嵌入式AI功能

在Leaf1上配置嵌入式AI功能，加载AI ECN组件需要使用的模型文件，本示例中模型文件已上传到设备上，完整路径为：flash:/AI_ECN-1.0.0-1.0.2.zip。缺省情况下，设备上已预加载一个模型文件。

- 对于V300R022C00之前版本。

  ```
  [~Leaf1] ai-service
  [~Leaf1-ai-service] model load file-path flash:/AI_ECN-1.0.0-1.0.2.zip all
  [~Leaf1-ai-service] quit
  ```

- 对于V300R022C00及之后版本。

```
[~Leaf1] quit
<Leaf1> load ai-service model-file flash:/AI_ECN-1.0.0-1.0.2.zip all
<Leaf1> system-view
```

步骤5 配置无损队列的AI ECN功能

配置Leaf1，在无损队列4上使能AI ECN功能。

```
[~Leaf1] ai-service
[~Leaf1-ai-service] ai-ecn
[*Leaf1-ai-service-ai-ecn] assign queue 4
[*Leaf1-ai-service-ai-ecn] ai-ecn enable
[*Leaf1-ai-service-ai-ecn] quit
[*Leaf1-ai-service] quit
[*Leaf1] commit
```

----结束

配置Spine1，Spine2上的配置与Spine1上的配置类似，配置过程略

步骤1 配置PFC优先级流量控制

1. 配置优先级映射

   # 根据规划，本次示例中设置RoCEv2的DSCP值为24，CNP报文的DSCP值为25，则配置设备中的Diffserv Domain优先级映射模板如下，将RoCEv2的优先级映射为优先级4（走队列4），CNP报文的优先级映射为优先级6（走队列6）。

   ```
   <HUAWEI> system-view
   [~HUAWEI] sysname Spine1
   [*HUAWEI] commit
   [~Spine1] diffserv domain ds1
   [*Spine1-dsdomain-ds1] ip-dscp-inbound 24 phb af4 green
   [*Spine1-dsdomain-ds1] ip-dscp-inbound 25 phb cs6 green
   [*Spine1-dsdomain-ds1] quit
   [*Spine1] commit
   ```

2. 为承载RoCEv2流量的优先级配置PFC功能

   #规划使用优先级4来承载网络中的RoCEv2流量，则需要在各个接口下针对优先级4使能PFC，并使能PFC功能基于DSCP映射后的优先级进行反压。

   ```
   [~Spine1] dcb pfc
   [~Spine1-dcb-pfc-default] priority 4
   [*Spine1-dcb-pfc-default] quit
   [*Spine1] port-group all_using
   [*Spine1-port-group-all_using] group-member 100ge 1/0/1 to 100ge 1/0/4
   [*Spine1-port-group-all_using] quit
   [*Spine1] commit
   [~Spine1] port-group all_using
   [*Spine1-port-all_using] dcb pfc enable mode manual
   [*Spine1-port-all_using] qos phb marking dscp enable
   [*Spine1-port-all_using] quit
   [*Spine1] dcb pfc dscp-mapping enable slot 1  //仅安装了P系列单板、SAN系列单板的CE16800支持本命令，如果用户入方向映射选择信任报文的DSCP映射内部优先级，还需要使用此命令，使能PFC功能基于DSCP映射后的优先级进行反压。
   [*Spine1] commit
   ```

   说明

   上述配置完成后，承载网络中的RoCEv2流量的优先级为4的队列即为无损队列。

步骤2 配置PFC死锁检测

配置无损队列的PFC死锁检测周期和恢复周期为100毫秒，并配置设备20s内出现5次PFC死锁时，去使能PFC功能。

```
[~Spine1] dcb pfc
[*Spine1-dcb-pfc-default] priority 4 turn-off threshold 5
```



```
[*Spine1-dcb-pfc-default] quit
[*Spine1] dcb pfc deadlock-detect timer 100
[*Spine1] dcb pfc deadlock-recovery timer 100
[*Spine1] commit
```

说明

配置完成后，若需要修改PFC死锁检测的配置，为了保障配置成功，防止设备处于死锁恢复期间，需要执行**shutdown**命令，关闭应用了PFC功能的端口。

步骤3 配置嵌入式AI功能

在Spine1上配置嵌入式AI功能，加载AI ECN组件需要使用的模型文件，本示例中模型文件已上传到设备上，完整路径为：flash:/AI_ECN-1.0.0-1.0.2.zip。缺省情况下，设备上已预加载一个模型文件。

- 对于V300R022C00之前版本。

```
[~Spine1] ai-service
[~Spine1-ai-service] model load file-path flash:/AI_ECN-1.0.0-1.0.2.zip all
[~Spine1-ai-service] quit
```

- 对于V300R022C00及之后版本。

```
[~Spine1] quit
<Spine1> load ai-service model-file flash:/AI_ECN-1.0.0-1.0.2.zip all
<Spine1> system-view
```

步骤4 配置无损队列的AI ECN功能

配置Spine1，在无损队列4上使能AI ECN功能。

```
[~Spine1] ai-service
[~Spine1-ai-service] ai-ecn
[*Spine1-ai-service-ai-ecn] assign queue 4
[*Spine1-ai-service-ai-ecn] ai-ecn enable
[*Spine1-ai-service-ai-ecn] quit
[*Spine1-ai-service] quit
[*Spine1] commit
```

----结束

检查配置结果

- 查看设备上所有的模型信息。可以看到，AI ECN功能已订阅新加载的模型。

```
<HUAWEI> display ai-service model
-------------------------------------------------------------------------------
File Name
        Model Name       Model Version    Model Type       Service
-------------------------------------------------------------------------------
AI_ECN-1.0.0-1.0.0.zip
        AI_ECN           1.0.0            Default          --
AI_ECN-1.0.0-1.0.2.zip
        AI_ECN           1.0.2            User Define      AI ECN
-------------------------------------------------------------------------------
```

- 查看PFC门限值。

```
[~Leaf1] display dcb pfc buffer interface 100ge1/0/1
Xon:        PFC backpressure stop threshold
Xoff:       PFC backpressure threshold
K:kilobytes   D:dynamic alpha
-----------------------------------------------------
Interface      Queue          Xon          Xoff
-----------------------------------------------------
100GE1/0/1        4        100(K)        125(K)
-----------------------------------------------------
```

- 查看PFC死锁触发和恢复的次数，**DeadlockNum**和**RecoveryNum**为0表示未触发死锁。

```
[~Leaf1] display dcb pfc interface 100ge 1/0/1
-------------------------------------------------------------------------------
Interface         Queue         Received(Frames)        ReceivedRate(pps)     DeadlockNum
                        Transmitted(Frames)     TransmittedRate(pps)     RecoveryNum
-------------------------------------------------------------------------------
100GE1/0/1          4                   0                   0           0
                                    0                   0           0
-------------------------------------------------------------------------------
```

- 查看无损队列的AI ECN功能计算出的ECN门限值，可以看到，AI ECN功能通过NN模式（模型推理模式）计算出了设备的ECN门限值。

```
[~Leaf1] display ai-ecn calculated state
AI-ECN Model Version : 1.0.1
Mode : NN - Model inference    BBR - Heuristic inference    STATIC - Static threshold
-------------------------------------------------------------------------------------------------------------------------
------
Interface      Queue   Low-Threshold   High-Threshold   Probability   Mode              Active
model       Actived time
                          (Byte)          (Byte)          (%)
-------------------------------------------------------------------------------------------------------------------------
------
100GE1/0/1          4          33024          150016          20     NN     AI_ECN_DistributedStorage
2022-01-10 09:09:23
100GE1/0/2          4          33024          150016          20     NN     AI_ECN_DistributedStorage
2022-01-10 09:09:23
-------------------------------------------------------------------------------------------------------------------------
------
```

配置脚本

Leaf1的配置脚本

```
#
sysname Leaf1
#
dcb pfc
 priority 4
 priority 4 turn-off threshold 5
#
dcb pfc deadlock-detect timer 100
dcb pfc deadlock-recover timer 100
dcb pfc dscp-mapping enable slot 1
#
diffserv domain ds1
 ip-dscp-inbound 24 phb af4 green
 ip-dscp-inbound 25 phb cs6 green
#
interface 25GE1/0/1
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/2
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/3
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/4
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/5
 trust dscp
 trust upstream ds1
```

```
 dcb pfc enable mode manual
#
interface 25GE1/0/6
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/7
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/8
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 100GE1/0/1
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/2
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/3
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/4
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
ai-service
 #
 ai-ecn
  ai-ecn enable
  assign queue 4
#
dcb pfc uplink group myuplink
 adjust original-dscp 24 to priority 5 dscp 32
 adjust original-dscp 32 to priority 2 dscp 16
 group-member interface 100GE1/0/1
 group-member interface 100GE1/0/2
 group-member interface 100GE1/0/3
 group-member interface 100GE1/0/4
#
port-group server_using
 group-member 25GE1/0/1
 group-member 25GE1/0/2
 group-member 25GE1/0/3
 group-member 25GE1/0/4
 group-member 25GE1/0/5
 group-member 25GE1/0/6
 group-member 25GE1/0/7
 group-member 25GE1/0/8
#
port-group spine_using
 group-member 100GE1/0/1
 group-member 100GE1/0/2
 group-member 100GE1/0/3
 group-member 100GE1/0/4
#
return
```

Leaf2的配置脚本

```
#
sysname Leaf2
#
dcb pfc
```

```
 priority 4
 priority 4 turn-off threshold 5
#
dcb pfc deadlock-detect timer 100
dcb pfc deadlock-recover timer 100
dcb pfc dscp-mapping enable slot 1
#
diffserv domain ds1
 ip-dscp-inbound 24 phb af4 green
 ip-dscp-inbound 25 phb cs6 green
#
interface 25GE1/0/1
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/2
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/3
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/4
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/5
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/6
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/7
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/8
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
......
#
interface 25GE1/0/23
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 25GE1/0/24
 trust dscp
 trust upstream ds1
 dcb pfc enable mode manual
#
interface 100GE1/0/1
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/2
 dcb pfc enable mode manual
```

```
 qos phb marking dscp enable
#
interface 100GE1/0/3
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/4
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
ai-service
 #
 ai-ecn
  ai-ecn enable
  assign queue 4
#
dcb pfc uplink group myuplink
 adjust original-dscp 24 to priority 5 dscp 32
 adjust original-dscp 32 to priority 2 dscp 16
 group-member interface 100GE1/0/1
 group-member interface 100GE1/0/2
 group-member interface 100GE1/0/3
 group-member interface 100GE1/0/4
#
port-group server_using
 group-member 25GE1/0/1
 group-member 25GE1/0/2
 group-member 25GE1/0/3
 group-member 25GE1/0/4
 group-member 25GE1/0/5
 group-member 25GE1/0/6
 group-member 25GE1/0/7
 group-member 25GE1/0/8
……
 group-member 25GE1/0/23
 group-member 25GE1/0/24
#
port-group spine_using
 group-member 100GE1/0/1
 group-member 100GE1/0/2
 group-member 100GE1/0/3
 group-member 100GE1/0/4
#
return
```

Spine1的配置脚本

```
#
sysname Spine1
#
dcb pfc
 priority 4
 priority 4 turn-off threshold 5
#
dcb pfc deadlock-detect timer 100
dcb pfc deadlock-recover timer 100
dcb pfc dscp-mapping enable slot 1
#
diffserv domain ds1
 ip-dscp-inbound 24 phb af4 green
 ip-dscp-inbound 25 phb cs6 green
#
interface 100GE1/0/1
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/2
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/3
```

```
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/4
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
ai-service
 #
 ai-ecn
  ai-ecn enable
  assign queue 4
#
port-group all_using
 group-member 100GE1/0/1
 group-member 100GE1/0/2
 group-member 100GE1/0/3
 group-member 100GE1/0/4
#
return
```

Spine2的配置脚本

```
#
sysname Spine2
#
dcb pfc
 priority 4
 priority 4 turn-off threshold 5
#
dcb pfc deadlock-detect timer 100
dcb pfc deadlock-recover timer 100
dcb pfc dscp-mapping enable slot 1
#
diffserv domain ds1
 ip-dscp-inbound 24 phb af4 green
 ip-dscp-inbound 25 phb cs6 green
#
interface 100GE1/0/1
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/2
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/3
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
interface 100GE1/0/4
 dcb pfc enable mode manual
 qos phb marking dscp enable
#
ai-service
 #
 ai-ecn
  ai-ecn enable
  assign queue 4
#
port-group all_using
 group-member 100GE1/0/1
 group-member 100GE1/0/2
 group-member 100GE1/0/3
 group-member 100GE1/0/4
#
return
```

1.15 配置带反射器的 iNOF 功能示例

适用产品和版本

安装了SAN系列单板的CE16800系列交换机V300R020C10或更高版本。

安装了P系列单板的CE16800系列交换机V300R021C00或更高版本。

CE6866、CE6866K、CE8851、CE8851K系列交换机V300R020C00或更高版本。

CE6860-SAN、CE8850-SAN系列交换机V300R020C10或更高版本。

CE6860-HAM、CE8850-HAM系列交换机V300R022C00或更高版本。

如果需要了解软件版本与交换机具体型号的配套信息，请查看**硬件查询工具**。

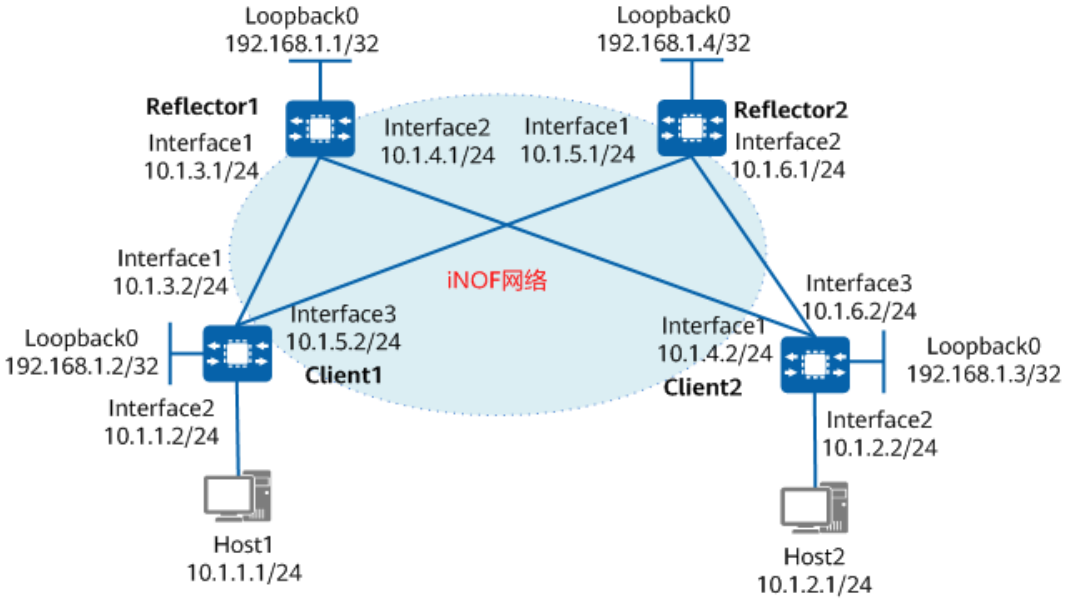
组网需求

如**图1-15**所示，Host1、Host2通过Client1、Client2接入网络，为了能更好的管理接入的主机，现在Reflector1、Reflector2、Client1和Client2上面配置iNOF功能，并指定Reflector1和Reflector2为iNOF反射器，Client1和Client2为iNOF客户端。

图 1-15 iNOF 功能双反射器组网图

说明

本例中interface1、interface2和interface3代表100GE1/0/1、100GE1/0/2和100GE1/0/3。



配置注意事项

主机和存储阵列（**图1-15**中的Host）须启用SNSD功能。

操作步骤

步骤1 配置各接口的IP地址，并配置路由协议，保证网络三层互通。

在Reflector1上配置接口的IP地址和OSPF路由协议。Reflector2、Client1和Client2上的配置与Reflector1上的配置类似，配置过程略。

```
<HUAWEI> system-view
[~HUAWEI] sysname Reflector1
[*HUAWEI] commit
[~Reflector1] interface 100GE 1/0/1
[~Reflector1-100GE1/0/1] undo portswitch
[*Reflector1-100GE1/0/1] ip address 10.1.3.1 24
[*Reflector1-100GE1/0/1] quit
[*Reflector1] interface 100GE 1/0/2
[*Reflector1-100GE1/0/2] undo portswitch
[*Reflector1-100GE1/0/2] ip address 10.1.4.1 24
[*Reflector1-100GE1/0/2] quit
[*Reflector1] interface loopback 0
[*Reflector1-LoopBack0] ip address 192.168.1.1 32
[*Reflector1-LoopBack0] quit
[*Reflector1] ospf 1
[*Reflector1-ospf-1] area 0
[*Reflector1-ospf-1-area-0.0.0.0] network 10.1.3.0 0.0.0.255
[*Reflector1-ospf-1-area-0.0.0.0] network 10.1.4.0 0.0.0.255
[*Reflector1-ospf-1-area-0.0.0.0] network 192.168.1.1 0.0.0.0
[*Reflector1-ospf-1-area-0.0.0.0] quit
[*Reflector1-ospf-1] quit
[*Reflector1] commit
```

步骤2 启用设备的LLDP功能。

在Reflector1上启用设备的LLDP功能。Reflector2、Client1和Client2上的配置与Reflector1上的配置一致，配置过程略。

```
[~Reflector1] lldp enable
[*Reflector1] commit
```

步骤3 配置iNOF报文的认证模式和认证密码。

在Reflector1上配置iNOF报文的认证模式为hmac-sha256，认证密码为Huawei@5678。Reflector2、Client1和Client2上的配置与Reflector1上的配置一致，配置过程略。

说明

用户配置了iNOF报文的认证模式和认证密码后，要确保对端设备也配置了相同的认证模式和认证密码，否则不能通过认证，只有通过认证后iNOF报文才能正常传输。

```
[~Reflector1] ai-service
[*Reflector1-ai-service] inof
[*Reflector1-ai-service-inof] authentication-mode hmac-sha256 password YsHsjx_202206
[*Reflector1-ai-service-inof] quit
[*Reflector1-ai-service] quit
[*Reflector1] commit
```

步骤4 配置iNOF反射器和客户端。

配置Reflector1为iNOF反射器。并在Reflector1上配置设备的本端地址为192.168.1.1，传输iNOF报文的端口号为10002。

说明

对于所有iNOF内的设备，传输iNOF报文的端口号需要配置成一致。

```
[~Reflector1] ai-service
[~Reflector1-ai-service] inof
```

```
[~Reflector1-ai-service-inof] role reflector
[*Reflector1-ai-service-inof] service-address 192.168.1.1 port-id 10002
[*Reflector1-ai-service-inof] quit
[*Reflector1-ai-service] quit
[*Reflector1] commit
```

配置Reflector2为iNOF反射器。并在Reflector2上配置设备的本端地址为192.168.1.4，传输iNOF报文的端口号为10002。

```
[~Reflector2] ai-service
[~Reflector2-ai-service] inof
[~Reflector2-ai-service-inof] role reflector
[*Reflector2-ai-service-inof] service-address 192.168.1.4 port-id 10002
[*Reflector2-ai-service-inof] quit
[*Reflector2-ai-service] quit
[*Reflector2] commit
```

配置Client1为iNOF客户端。并在Client1上配置设备的本端地址为192.168.1.2，传输iNOF报文的端口号为10002。

```
[~Client1] ai-service
[~Client1-ai-service] inof
[~Client1-ai-service-inof] role reflect-client
[*Client1-ai-service-inof] service-address 192.168.1.2 port-id 10002
[*Client1-ai-service-inof] quit
[*Client1-ai-service] quit
[*Client1] commit
```

配置Client2为iNOF客户端。并在Client2上配置设备的本端地址为192.168.1.3，传输iNOF报文的端口号为10002。

```
[~Client2] ai-service
[~Client2-ai-service] inof
[~Client2-ai-service-inof] role reflect-client
[*Client2-ai-service-inof] service-address 192.168.1.3 port-id 10002
[*Client2-ai-service-inof] quit
[*Client2-ai-service] quit
[*Client2] commit
```

步骤5 在iNOF反射器上指定客户端地址。

在Reflector1上指定iNOF客户端的地址为192.168.1.2和192.168.1.3，并指定Reflector2也为本端设备的客户端。

```
[~Reflector1] ai-service
[~Reflector1-ai-service] inof
[~Reflector1-ai-service-inof] peer 192.168.1.2 reflect-client
[*Reflector1-ai-service-inof] peer 192.168.1.3 reflect-client
[*Reflector1-ai-service-inof] peer 192.168.1.4 reflect-client
[*Reflector1-ai-service-inof] quit
[*Reflector1-ai-service] quit
[*Reflector1] commit
```

在Reflector2上指定iNOF客户端的地址为192.168.1.2和192.168.1.3，并指定Reflector1也为本端设备的客户端。

```
[~Reflector2] ai-service
[~Reflector2-ai-service] inof
[~Reflector2-ai-service-inof] peer 192.168.1.2 reflect-client
[*Reflector2-ai-service-inof] peer 192.168.1.3 reflect-client
[*Reflector2-ai-service-inof] peer 192.168.1.1 reflect-client
[*Reflector2-ai-service-inof] quit
[*Reflector1-ai-service] quit
[*Reflector2] commit
```

步骤6 在iNOF反射器上创建iNOF自定义域。

在Reflector1上去使能iNOF默认域自动加入功能，节省网络资源。Reflector2上的配置与Reflector1上的配置一致，配置过程略。


```
[~Reflector1] ai-service
[~Reflector1-ai-service] inof
[~Reflector1-ai-service-inof] undo default-zone enable
[*Reflector1-ai-service-inof] quit
[*Reflector1-ai-service] quit
[*Reflector1] commit
```

在Reflector1上创建名为zone1的iNOF自定义域，并将Host1和Host2的IP地址加入该域。Reflector2上的配置与Reflector1上的配置一致，配置过程略。

```
[~Reflector1] ai-service
[~Reflector1-ai-service] inof
[~Reflector1-ai-service-inof] zone zone1
[*Reflector1-ai-service-inof-zone-zone1] host 10.1.1.1
[*Reflector1-ai-service-inof-zone-zone1] host 10.1.2.1
[*Reflector1-ai-service-inof] quit
[*Reflector1-ai-service] quit
[*Reflector1] commit
```

说明

为了确保Reflector1和Reflector2互为备份，两个反射器上的域配置需要保持一致。

步骤7 配置BFD for iNOF功能。

在Reflector1上配置BFD for iNOF功能。Reflector2上的配置与Reflector1上的配置一致，配置过程略。

```
[~Reflector1] bfd
[*Reflector1] commit
[~Reflector1] ai-service
[~Reflector1-ai-service] inof
[~Reflector1-ai-service-inof] inof bfd enable
[*Reflector1-ai-service-inof] quit
[*Reflector1-ai-service] quit
[*Reflector1] commit
```

步骤8 启用iNOF域隔离功能。

在Reflector1上配置iNOF域隔离功能。Reflector2上的配置与Reflector1上的配置一致，配置过程略。

```
[~Reflector1] ai-service
[~Reflector1-ai-service] inof
[~Reflector1-ai-service-inof] hard-zoning enable
[*Reflector1-ai-service-inof] quit
[*Reflector1-ai-service] quit
[*Reflector1] commit
```

----结束

检查配置结果

在Reflector1上查看配置iNOF反射器和客户端后，本端设备与对端设备建立的iNOF连接信息。

```
[~Reflector1] display inof peer connection
Role: Reflector
IPv4 Info:
Port: 10002
ServiceAddress: 192.168.1.1
------------------------------------------------------------
Index  PeerIP          ConnectStatus  EstablishTime
------------------------------------------------------------
    1  192.168.1.2     Established    2020-09-03 23:31:12
    1  192.168.1.3     Established    2020-09-03 23:31:12
```



```
  2  192.168.1.4      Established    2020-09-03 23:32:12
------------------------------------------------------------
```

查看Reflector1上iNOF域成员的配置信息。

```
[~Reflector1] display inof configuration host
IPv4 Info:
-------------------------------------------------------------------------------------------------------------
Host              Learned-From        ZoneName
-------------------------------------------------------------------------------------------------------------
10.1.1.1          Local               zone1
10.1.1.1          192.168.1.4           zone1
10.1.2.1          Local               zone1
10.1.2.1          192.168.1.4           zone1
-------------------------------------------------------------------------------------------------------------
```

配置脚本

- Reflector1的配置脚本

```
#
sysname Reflector1
#
bfd
lldp enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.3.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.4.1 255.255.255.0
#
interface LoopBack0
  ip address 192.168.1.1 255.255.255.255
#
ai-service
 #
 inof
  authentication-mode hmac-sha256 password %+%##!<x@k!01K6<Di#Xie66M:rx~U7=>Ws*I
$1!!!!!!!!!!!!!!;!!!!%7&7
  hard-zoning enable
  inof bfd enable
  peer 192.168.1.2 reflect-client
  peer 192.168.1.3 reflect-client
  peer 192.168.1.4 reflect-client
  role reflector
  service-address 192.168.1.1 port-id 10002
  undo default-zone enable
  #
  zone zone1
   host 10.1.1.1
   host 10.1.2.1
#
ospf 1
 area 0.0.0.0
  network 10.1.3.0 0.0.0.255
  network 10.1.4.0 0.0.0.255
  network 192.168.1.1 0.0.0.0
#
return
```

- Reflector2的配置脚本

```
#
sysname Reflector2
#
bfd
lldp enable
```



```
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.5.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.6.1 255.255.255.0
#
interface LoopBack0
  ip address 192.168.1.4 255.255.255.255
#
ai-service
 #
 inof
  authentication-mode hmac-sha256 password %+%##!<x@k!01Q_/8W\@B'Bq$FrD66M:rx~U7=>Ws*I
$1!!!!!!!!!!!;!!!!%7&7
  hard-zoning enable
  inof bfd enable
  peer 192.168.1.2 reflect-client
  peer 192.168.1.3 reflect-client
  peer 192.168.1.1 reflect-client
  role reflector
  service-address 192.168.1.4 port-id 10002
  undo default-zone enable
  #
  zone zone1
   host 10.1.1.1
   host 10.1.2.1
#
ospf 1
 area 0.0.0.0
  network 10.1.5.0 0.0.0.255
  network 10.1.6.0 0.0.0.255
  network 192.168.1.4 0.0.0.0
#
return
```

- Client1的配置脚本

```
#
sysname Client1
#
lldp enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.3.2 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.1.2 255.255.255.0
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.1.5.2 255.255.255.0
#
interface LoopBack0
  ip address 192.168.1.2 255.255.255.255
#
ai-service
 #
 inof
  authentication-mode hmac-sha256 password %+%##!!!<x@k!01Kv>7iLc"ypITVI3>d<2PQ
\g)Y(!!!!!!!!!!!!!!:!!!!'JVz
  role reflect-client
  service-address 192.168.1.2 port-id 10002
#
ospf 1
 area 0.0.0.0
  network 10.1.1.0 0.0.0.255
```

```
  network 10.1.3.0 0.0.0.255
  network 10.1.5.0 0.0.0.255
  network 192.168.1.2 0.0.0.0
#
return
```

- Client2的配置脚本

```
#
sysname Client2
#
lldp enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.4.2 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.2.2 255.255.255.0
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.1.6.2 255.255.255.0
#
interface LoopBack0
  ip address 192.168.1.3 255.255.255.255
#
ai-service
 #
 inof
  authentication-mode hmac-sha256 password %+%#!!!!"!!x@k!01Km:Q_/8W\@B'Bq$FrD\i;tK-\
(!!!!!!!!!!!!!!;!!!!&.lER
  role reflect-client
  service-address 192.168.1.3 port-id 10002
#
ospf 1
 area 0.0.0.0
  network 10.1.2.0 0.0.0.255
  network 10.1.4.0 0.0.0.255
  network 10.1.6.0 0.0.0.255
  network 192.168.1.3 0.0.0.0
#
return
```


2 特性典型配置案例

2.1 基础配置

2.1.1 登录设备命令行界面

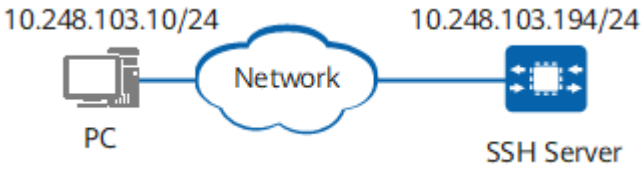
2.1.1.1 举例：配置用户通过 STelnet 登录设备

组网需求

如**图1**所示，在作为SSH服务器的设备上使能STelnet服务器功能后，SSH客户端PC可以通过不同的认证方式登录SSH服务器，这里以RSA认证方式为例介绍客户端通过STelnet登录服务器的配置过程。

为了提升系统安全性，防止非法用户登录到SSH服务器，用户可以在SSH服务器上配置ACL规则。

图 2-1 配置用户通过 STelnet 登录设备组网图



配置思路

采用如下的思路配置SSH用户通过STelnet登录设备：

1. 配置SSH服务器的管理网口IP地址。
2. 在SSH服务器端生成本地密钥对。
3. 配置SSH服务器的VTY用户界面。
4. 创建本地用户，并配置服务类型。
5. 创建SSH用户，并配置认证方式。
6. SSH客户端根据配置的SSH用户认证类型创建相应的密钥对，并将公钥拷贝至SSH服务器。
7. SSH服务器端编辑公钥，并将编辑好的公钥分配给用户。
8. 使能SSH服务器的STelnet功能，配置SSH用户的服务类型为STelnet。
9. 在SSH服务器上配置允许STelnet客户端登录的ACL规则。
10. 配置客户端登录软件的参数，STelnet至服务器。

数据准备

为完成此配置示例，需准备如下数据：

说明

为了保证更好的安全性，建议使用3072位及以上的RSA密钥对。

- SSH客户端已安装OpenSSH软件。
- SSH服务器管理网口的IP地址为10.248.103.194/24。
- 本地用户的认证方式为password，用户名为“admin123”，密码为“YsHsjx_202206”。
- SSH用户的认证方式为RSA。
- 配置基本的ACL 2000，允许10.248.103.0/24网段的客户端合法接入SSH服务器。

操作步骤

步骤1 配置SSH服务器的管理网口IP地址。

```
<HUAWEI> system-view
[~HUAWEI] sysname SSH Server
[*HUAWEI] commit
[~SSH Server] interface meth 0/0/0
[~SSH Server-MEth0/0/0] ip address 10.248.103.194 255.255.255.0
```

```
[*SSH Server-MEth0/0/0] quit
[*SSH Server] commit
```

步骤2 在SSH服务器端生成本地密钥对。

```
[~SSH Server] rsa local-key-pair create
The key name will be:Host
The range of public key size is (2048, 4096).
NOTE: Key pair generation will take a short while.
Please input the modulus [default = 3072]:3072
[*SSH Server] commit
```

步骤3 配置SSH服务器的VTY用户界面。

```
[~SSH Server] user-interface vty 0 4
[~SSH Server-ui-vty0-4] authentication-mode aaa
[*SSH Server-ui-vty0-4] protocol inbound ssh
[*SSH Server-ui-vty0-4] quit
[*SSH Server] commit
```

说明

若配置登录协议为SSH，则设备将自动禁止Telnet功能。

步骤4 在服务器端创建本地用户，并配置用户服务方式。

```
[~SSH Server] aaa
[~SSH Server-aaa] local-user admin123 password
Please configure the login password (8-128)
It is recommended that the password consist of at least 2 types of characters, i
ncluding lowercase letters, uppercase letters, numerals and special characters.
Please enter password:
Please confirm password:
Info: Add a new user.
[*SSH Server-aaa] local-user admin123 service-type ssh
[*SSH Server-aaa] local-user admin123 privilege level 3
[*SSH Server-aaa] quit
[*SSH Server] commit
```

步骤5 在服务器端创建SSH用户，并配置认证方式。

```
[~SSH Server] ssh user admin123
[*SSH Server] ssh user admin123 authentication-type rsa
[*SSH Server] commit
```

步骤6 配置SSH服务器的公钥算法、加密算法、密钥交换算法列表、HMAC认证算法和最小密钥长度。

```
[~SSH Server] ssh server cipher aes128_ctr aes256_ctr aes192_ctr aes128_gcm aes256_gcm
[~SSH Server] ssh server hmac sha2_256 sha2_512
[~SSH Server] ssh server key-exchange dh_group_exchange_sha256 dh_group16_sha512
[~SSH Server] ssh server publickey rsa_sha2_256 rsa_sha2_512
[~SSH Server] ssh server dh-exchange min-len 3072
[*SSH Server] commit
```

步骤7 SSH客户端使用OpenSSH创建RSA密钥对，并将密钥对中的公钥拷贝至SSH服务器。

进入Windows的命令行提示符，创建RSA密钥对，并保存到本地id_rsa.pub文件中（以下内容仅为示例）。

```
C:\Users\User1>ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\User1/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\User1/.ssh/id_rsa.
Your public key has been saved in C:\Users\User1/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:c43yubJjCUjY3JqH0aVZwJFM3gWJcH4YI5+4HUDAIqo
The key's randomart image is:
+---[RSA 3072]----+
| ..o==B=.o.      |
```

```
|o .  O=*+.       |
|o. +.oB=o        |
|. . =o=o   o     |
|.  ..*. S o .    |
|E   = o  = .     |
|     . . .o      |
|        =  .     |
|       ..+.      |
+----[SHA256]-----+
```

步骤8 SSH服务器编辑SSH客户端OpenSSH生成的公钥，并将编辑后的公钥分配给SSH用户。

```
[~SSH Server] rsa peer-public-key rsa01 encoding-type openssh
[*SSH Server-rsa-public-key] public-key-code begin
[*SSH Server-rsa-public-key-rsa-key-code] ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQCg5Ag490i6ilB7QuCVb35B8RJEh1DIYB88h2p1qjdh7qdMQv8rp
JaVAgQWxwzKZO0XdFuz4ReGQzTCSf7Det7Ajicddw3qi+6P8hRqZj6MPdLg/o3RN4aPCfr/
LFWCwqJ3gWGHlOC7qqjRk+6pySVoiWcSk5/elBkU7WVk/
cSWrt4qFXJV373OCesKcEVeDvAa1Tvx6L3LQroBqUO0EXzDgOthPCmOqiqvS5h3JipzqVsesdSKjeInooCQzS
Ov5eePpBcFcIvU6wFiLIZ5vnf6YtypgTVzHuje/sh4xM7Iuuon7AYXKHT8NpO9jd9zA/lKaRPXyDtei1O1Bt/
5lxnn
[*SSH Server-rsa-public-key-rsa-key-code] public-key-code end
[*SSH Server-key-code] peer-public-key end
[*SSH Server] ssh user admin123 assign rsa-key rsa01
[*SSH Server] commit
```

步骤9 使能STelnet功能，并配置用户的服务类型为STelnet。

```
[~SSH Server] stelnet server enable
[*SSH Server] ssh server-source all-interface
[*SSH Server] ssh user admin123 service-type stelnet
[*SSH Server] commit
```

步骤10 配置ACL规则。

```
[~SSH Server] acl 2000
[*SSH Server-acl4-basic-2000] rule permit source 10.248.103.0 8
[*SSH Server-acl4-basic-2000] quit
[*SSH Server] ssh server acl 2000
[*SSH Server] commit
```

----结束

检查配置结果

客户端通过OpenSSH软件登录SSH服务器。进入Windows的命令行提示符，执行OpenSSH命令，通过STelnet方式访问设备。

```
C:\Users\User1>ssh admin123@10.248.103.194
Enter passphrase for key 'C:\Users\User/.ssh/id_rsa':
Info: The max number of VTY users is 21, the number of current VTY users online is 4, and total number of
terminal users online is 4.
      The current login time is 2020-12-15 15:58:03.
<SSH Server>
```

配置脚本

```
#
sysname SSH Server
#
acl number 2000
 rule 5 permit source 10.248.103.0 0.0.0.255
#
rsa peer-public-key rsa01 encoding-type openssh
 public-key-code begin
  ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQCg5Ag490i6ilB7QuCVb35B8RJEh1DIYB88h2p1qjdh7qdMQv8rpJaVA
gQWxwzKZO0XdFuz4ReGQzTCSf7Det7Ajicddw3qi+6P8hRqZj6MPdLg/o3RN4aPCfr/
LFWCwqJ3gWGHlOC7qqjRk+6pySVoiWcSk5/elBkU7WVk/
```



```
cSWrt4qFXJV373OCesKcEVeDvAa1Tvx6L3LQroBqUO0EXzDgOthPCmOqiqvS5h3JipzqVsesdSKjeInooCQzSOv5e
ePpBcFcIvU6wFiLIZ5vnf6YtypgTVzHuje/sh4xM7Iuuon7AYXKHT8NpO9jd9zA/lKaRPXyDtei1O1Bt/5lxnn rsa-key
 public-key-code end
 peer-public-key end
#
aaa
 local-user admin123 password irreversible-cipher $1d$+,JS+))\\2$KVNj(.
3`_5x0FCKGv}H&.kUTI`Ff&H*eBqO.ua>)$
 local-user admin123 service-type terminal ssh
 local-user admin123 privilege level 3
#
interface MEth0/0/0
 ip address 10.248.103.194 255.255.255.0
#
stelnet server enable
ssh user admin123
ssh user admin123 authentication-type rsa
ssh user admin123 assign rsa-key rsa01
ssh user admin123 service-type stelnet
ssh server-source all-interface
ssh server acl 2000
#
ssh server cipher aes128_ctr aes256_ctr aes192_ctr aes128_gcm aes256_gcm
ssh server hmac sha2_256 sha2_512
ssh server key-exchange dh_group_exchange_sha256 dh_group16_sha512
ssh server publickey rsa_sha2_256 rsa_sha2_512
ssh server dh-exchange min-len 3072
#
user-interface vty 0 4
 authentication-mode aaa
 protocol inbound ssh
#
return
```

2.1.1.2 举例：配置设备作为 STelnet 客户端登录其他设备

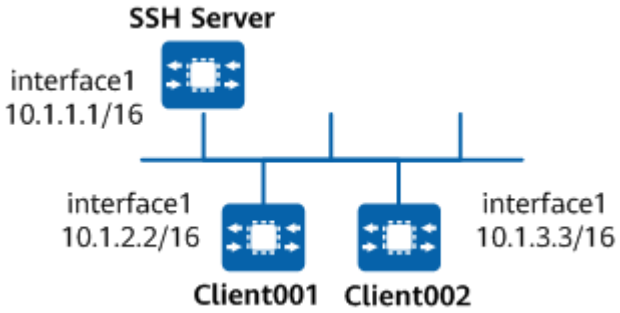
组网需求

如图2-2所示，用户希望在服务器端和客户端进行安全的数据交互，配置两个登录用户为Client001和Client002，分别使用password认证方式和RSA认证方式登录SSH服务器，并且配置新的端口号，而不使用缺省端口号。

图 2-2 配置通过 STelnet 登录其他设备组网图

说明

本例中interface1代表100GE1/0/1。



配置思路

采用如下的思路配置通过STelnet登录其他设备：

1. 在SSH服务器端生成本地密钥对，实现在服务器端和客户端进行安全的数据交互。
2. 在SSH服务器端配置SSH用户client001和client002分别使用不同的认证方式。
3. 在SSH服务器端开启STelnet服务功能。
4. 在SSH服务器端配置SSH用户client001和client002的服务方式为STelnet。
5. 在SSH服务器端配置SSH服务器的端口号，有效防止攻击者对SSH服务标准端口的访问，确保安全性。
6. 用户client001和client002分别以STelnet方式实现登录SSH服务器。

操作步骤

步骤1 在服务器端生成本地密钥对。

```
<HUAWEI> system-view
[~HUAWEI] sysname SSH Server
[*HUAWEI] commit
[~SSH Server] rsa local-key-pair create
The key name will be:Host
The range of public key size is (2048, 4096).
NOTE: Key pair generation will take a short while.
Please input the modulus [default = 3072]:
[*SSH Server] commit
```

步骤2 在服务器端创建SSH用户。

配置VTY用户界面。

```
[~SSH Server] user-interface vty 0 4
[~SSH Server-ui-vty0-4] authentication-mode aaa
[*SSH Server-ui-vty0-4] protocol inbound ssh
[*SSH Server-ui-vty0-4] quit
[*SSH Server] commit
```

- 创建SSH用户client001。

  # 新建用户名为client001的SSH用户，且认证方式为password。

  ```
  [~SSH Server] aaa
  [~SSH Server-aaa] local-user client001 password
  Please configure the login password (8-128)
  It is recommended that the password consist of at least 2 types of characters, i
  ncluding lowercase letters, uppercase letters, numerals and special characters.
  Please enter password:
  Please confirm password:
  Info: Add a new user.
  [*SSH Server-aaa] local-user client001 privilege level 3
  [*SSH Server-aaa] local-user client001 service-type ssh
  [*SSH Server-aaa] quit
  [*SSH Server] ssh user client001
  [*SSH Server] ssh user client001 authentication-type password
  [*SSH Server] commit
  ```

  #在客户端Client001，配置加密算法、HMAC认证算法、密钥交换算法列表、公钥算法。

  ```
  <HUAWEI> system-view
  [~HUAWEI] sysname client001
  [*HUAWEI] commit
  [*client001] ssh client cipher aes128_ctr aes256_ctr aes192_ctr aes128_gcm aes256_gcm
  [*client001] ssh client hmac sha2_256 sha2_512
  [*client001] ssh client key-exchange dh_group_exchange_sha256 dh_group16_sha512
  [*client001] ssh client publickey rsa_sha2_256 rsa_sha2_512
  [*client001] commit
  ```

- 创建SSH用户client002。

  # 新建用户名为client002的SSH用户，且认证方式为RSA。

```
[~SSH Server] ssh user client002
[*SSH Server] ssh user client002 authentication-type rsa
[*SSH Server] ssh authorization-type default root
[*SSH Server] commit
```

在STelnet客户端Client002生成客户端的本地密钥对。

```
<HUAWEI> system-view
[~HUAWEI] sysname client002
[*HUAWEI] commit
[~client002] rsa local-key-pair create
The key name will be: client002_Host
The range of public key size is (2048, 4096).
NOTE: Key pair generation will take a short while.
Please input the modulus [default = 3072]:
[*client002] commit
```

配置STelnet客户端Client002的加密算法、HMAC认证算法、密钥交换算法列表、公钥算法。

```
[*client002] ssh client cipher aes128_ctr aes256_ctr aes192_ctr aes128_gcm aes256_gcm
[*client002] ssh client hmac sha2_256 sha2_512
[*client002] ssh client key-exchange dh_group_exchange_sha256 dh_group16_sha512
[*client002] ssh client publickey rsa_sha2_256 rsa_sha2_512
[*client002] commit
```

查看客户端上生成的RSA密钥对的公钥部分。

```
[~client002] display rsa local-key-pair public
=====================Host key=========================
Time of key pair created : 2019-11-03 08:56:38
Key name                 : client002_Host
Key type                 : RSA encryption key
======================================================
Key code:

3082010A
  02820101
    00A4BAB8 B964077E F7657F7F E4BE1DE8 71EE1707
    E4EE2864 2D06FBE0 BFC1CB52 F99B7A99 0132B709
    3F841CA2 3544B8B2 6EE0A9ED 04B19FE3 FB3DA86D
    BE68FFE2 2303108D BDC24B80 A1793A08 FDA0B6C1
    13C31EA5 298EC9B1 2B0BC8BD 32CFF896 29F8CA98
    8B1724AF 5DA8A390 20906ADE 6A8AD77D 6234F0C8
    DC965BA0 1771D9C0 A89ED49B 5ECF7EE2 D5997527
    FC87FE03 E51658C1 0996DFDF DC456376 2FA4B268
    4345131D 431419D2 DD5E4003 6A7D3295 145F3175
    22E80686 E6B39A05 799D6BCF A78F69B6 BC2D0836
    F5013421 77D68B89 A9EC182A 04B87BE3 500FCE14
    9C95CF78 75704359 0C70FD60 1EFC0B99 32F02142
    4CE781E4 36A60BFC 2CBD07F6 9E700CEE 4D0203
    010001

Host public key for PEM format code:
---- BEGIN SSH2 PUBLIC KEY ----
AAAAB3NzaC1yc2EAAAADAQABAAABAQCkuri5ZAd+92V/f+S+Hehx7hcH5O4oZC0G
++C/wctS+Zt6mQEytwk/hByiNUS4sm7gqe0EsZ/j+z2obb5o/+IjAxCNvcJLgKF5
Ogj9oLbBE8MepSmOybErC8i9Ms/4lin4ypiLFySvXaijkCCQat5qitd9YjTwyNyW
W6AXcdnAqJ7Um17PfuLVmXUn/If+A+UWWMEJlt/f3EVjdi+ksmhDRRMdQxQZ0t1e
QANqfTKVFF8xdSLoBobms5oFeZ1rz6ePaba8LQg29QE0IXfWi4mp7BgqBLh741AP
zhSclc94dXBDWQxw/WAe/AuZMvAhQkzngeQ2pgv8LL0H9p5wDO5N
---- END SSH2 PUBLIC KEY ----

Public key code for pasting into OpenSSH authorized_keys file:
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCkuri5ZAd+92V/f+S+Hehx7hcH5O4oZC0G++C/wctS
+Zt6mQEytwk/hByiNUS4sm7gqe0EsZ/j+z2obb5o/+IjAxCNvcJLgKF5Ogj9oLbBE8MepSmOybErC8i9Ms/
4lin4ypiLFySvXaijkCCQat5qitd9YjTwyNyWW6AXcdnAqJ7Um17PfuLVmXUn/If+A+UWWMEJlt/f3EVjdi
+ksmhDRRMdQxQZ0t1eQANqfTKVFF8xdSLoBobms5oFeZ1rz6ePaba8LQg29QE0IXfWi4mp7BgqBLh741
APzhSclc94dXBDWQxw/WAe/AuZMvAhQkzngeQ2pgv8LL0H9p5wDO5N rsa-key

Host public key for SSH1 format code:
2048 65537
20795157856672359848547361269858029949242843585831182669194523227368193104900346497
```

```
5156406283877999441481175657431905603728398665186508263345707894349677484217580598190093729334060817838060780955449126599749626192655532498343534107533323544305478060443118682108915155361063215476748577556785624206276792428389535386415963031963197355449455867856248244224701824312943027014161231178397535397111353242333550044093772619909488601542170799462826313639069974340296484981794888174430354307491156572632525093810706287949592233095399772699929571517497640619130599435578042197052660114800711855592023422161491751889426268114695
```

```
======================Server key=======================
Time of key pair created : 2019-11-03 08:56:39
Key name                 : client002_Server
Key type                 : RSA encryption key
=======================================================
Key code:

3081B9
  0281B1
    00B9AE42 B8419F19 35C49A7B A55DBB6F 67D931F3
    9C19ECF9 9E17961B D01ED5DD 3AE68CFA 38C57113
    C93663F2 86768B19 AD0F603E 98F2C6AB A71A6C26
    8813411D 4AA56BC4 6505EC15 94647621 AB7D03BB
    79DA9B24 09BB1FD2 3927E2F9 00F79116 466411CD
    AC3D8FF6 A051FA5A 9BCE84CE 20842134 D2D27B4A
    219CB801 9F5A90E0 518DEEFC F48F5ED4 49215B1F
    11E1AC81 5E168A97 3AA5320D 7B158556 AF5CC95C
    9B508BBC 6EEFEEF9 0E23AA13 59E1F746 D5
  0203
    010001
```

将客户端上产生的RSA公钥配置到服务器端（上面**display**命令显示信息中黑体部分即为客户端产生的RSA公钥，将其拷贝粘贴至服务器端）。

```
[~SSH Server] rsa peer-public-key rsakey001
[*SSH Server-rsa-public-key] public-key-code begin
[*SSH Server-rsa-public-key-rsa-key-code] 3082010A
[*SSH Server-rsa-public-key-rsa-key-code] 2820101
[*SSH Server-rsa-public-key-rsa-key-code] 00A4BAB8 B964077E F7657F7F E4BE1DE8 71EE1707
[*SSH Server-rsa-public-key-rsa-key-code] E4EE2864 2D06FBE0 BFC1CB52 F99B7A99 0132B709
[*SSH Server-rsa-public-key-rsa-key-code] 3F841CA2 3544B8B2 6EE0A9ED 04B19FE3 FB3DA86D
[*SSH Server-rsa-public-key-rsa-key-code] BE68FFE2 2303108D BDC24B80 A1793A08 FDA0B6C1
[*SSH Server-rsa-public-key-rsa-key-code] 13C31EA5 298EC9B1 2B0BC8BD 32CFF896 29F8CA98
[*SSH Server-rsa-public-key-rsa-key-code] 8B1724AF 5DA8A390 20906ADE 6A8AD77D 6234F0C8
[*SSH Server-rsa-public-key-rsa-key-code] DC965BA0 1771D9C0 A89ED49B 5ECF7EE2 D5997527
[*SSH Server-rsa-public-key-rsa-key-code] FC87FE03 E51658C1 0996DFDF DC456376 2FA4B268
[*SSH Server-rsa-public-key-rsa-key-code] 4345131D 431419D2 DD5E4003 6A7D3295 145F3175
[*SSH Server-rsa-public-key-rsa-key-code] 22E80686 E6B39A05 799D6BCF A78F69B6 BC2D0836
[*SSH Server-rsa-public-key-rsa-key-code] F5013421 77D68B89 A9EC182A 04B87BE3 500FCE14
[*SSH Server-rsa-public-key-rsa-key-code] 9C95CF78 75704359 0C70FD60 1EFC0B99 32F02142
[*SSH Server-rsa-public-key-rsa-key-code] 4CE781E4 36A60BFC 2CBD07F6 9E700CEE 4D
[*SSH Server-rsa-public-key-rsa-key-code] 203
[*SSH Server-rsa-public-key-rsa-key-code] 10001
[*SSH Server-rsa-public-key-rsa-key-code] public-key-code end
[*SSH Server-rsa-public-key] peer-public-key end
[*SSH Server] commit
```

在SSH服务器端为SSH用户client002绑定STelnet客户端的RSA公钥。

```
[~SSH Server] ssh user client002 assign rsa-key rsakey001
```

步骤3 SSH服务器端开启STelnet服务功能，并指定SSH服务端的源接口。

开启STelnet服务功能。

```
[*SSH Server] stelnet server enable
```

指定SSH服务端的源接口。

```
[*SSH Server] ssh server-source all-interface
```

配置SSH服务器的公钥算法、加密算法、密钥交换算法列表、HMAC认证算法和最小密钥长度。


```
[~SSH Server] ssh server cipher aes128_ctr aes256_ctr aes192_ctr aes128_gcm aes256_gcm
[~SSH Server] ssh server hmac sha2_256 sha2_512
[~SSH Server] ssh server key-exchange dh_group_exchange_sha256 dh_group16_sha512
[~SSH Server] ssh server publickey rsa_sha2_256 rsa_sha2_512
[~SSH Server] ssh server dh-exchange min-len 3072
[*SSH Server] commit
```

步骤4 配置SSH用户client001、client002的服务方式为STelnet。

```
[*SSH Server] ssh user client001 service-type stelnet
[*SSH Server] ssh user client002 service-type stelnet
```

步骤5 配置SSH服务器端新的端口号。

```
[*SSH Server] ssh server port 1025
[*SSH Server] commit
```

步骤6 STelnet客户端连接SSH服务器。

第一次登录，需要使能SSH客户端首次登录功能。

使能客户端Client001首次登录功能。

```
<HUAWEI> system-view
[~HUAWEI] sysname client001
[*HUAWEI] commit
[~client001] ssh client first-time enable
[*client001] commit
[~client001] quit
```

使能客户端Client002首次登录功能。

```
[~client002] ssh client first-time enable
[*client002] commit
[~client002] quit
```

STelnet客户端Client001用password认证方式连接SSH服务器，输入配置的用户名和密码。

```
<client001> stelnet 10.1.1.1 1025
Trying 10.1.1.1 ...
Press CTRL+K to abort
Connected to 10.1.1.1 ...
The server's public key does not match the one cached before.
The server is not authenticated. Continue to access it?[Y/N]:y
The keyname:10.1.1.1 already exists. Update it? [Y/N]:n

Please input the username: client001
Please select public key type for user authentication [R for RSA/D for DSA/E for ECC] Please select [R/D/E]:r
Enter password:
```

输入密码，显示登录成功信息如下：

```
Warning: The initial password poses security risks.
The password needs to be changed. Change now? [Y/N]:n

Info: The max number of VTY users is 21, the number of current VTY users online
is 4, and total number of terminal users online is 4.
    The current login time is 2013-12-31 11:22:06.
    The last login time is 2013-12-31 10:24:13 from 10.1.2.2 through SSH.
<SSH Server>
```

STelnet客户端Client002用RSA认证方式连接SSH服务器。

```
<client002> stelnet 10.1.1.1 1025
Trying 10.1.1.1 ...
Press CTRL+K to abort
Connected to 10.1.1.1 ...
The server's public key does not match the one cached before.
The server is not authenticated. Continue to access it?[Y/N]:y
The keyname:192.168.1.182 already exists. Update it? [Y/N]: n
```



```
Please input the username: client002
Please select public key type for user authentication [R for RSA/D for DSA/E for ECC] Please select [R/D/E]:r

Info: The max number of VTY users is 21, the number of current VTY users online
is 4, and total number of terminal users online is 4.
    The current login time is 2013-12-31 11:36:06.

<SSH Server>
```

如果登录成功，用户将进入用户视图。如果登录失败，用户将收到**Session is disconnected**的信息。

----结束

检查配置结果

攻击者使用原端口号22登录SSH服务器，不能成功。

```
<client002> stelnet 10.1.1.1
Trying 10.1.1.1 ...
Press CTRL+K to abort
Error: Failed to connect to the remote host.
```

在SSH服务器端执行**display ssh server status**命令可以查看到STelnet服务已经使能。执行**display ssh user-information**命令可以查看服务器端SSH用户信息。

查看SSH状态信息。

```
[~SSH Server] display ssh server status
SSH Version                          : 2.0
SSH authentication timeout (Seconds)      : 60
SSH authentication retries (Times)        : 3
SSH server key generating interval (Hours) : 0
SSH version 1.x compatibility             : Disable
SSH server keepalive                      : Enable
SFTP server                          : Enable
STelnet server                       : Enable
SNETCONF server                           : Disable
SNETCONF server port(830)                 : Enable
SCP server                           : Disable
SSH server port                       : 1025
ACL name                             : --
ACL number                            : --
ACL6 name                            : --
ACL6 number                           : --
SSH server source address                 : 0.0.0.0
```

查看SSH用户信息。

```
[~SSH Server] display ssh user-information
------------------------------------------------------------------------------
User Name            : client001
Authentication type  : password
User public key name  : --
User public key type  : --
Sftp directory       : flash:
Service type         : stelnet

User Name            : client002
Authentication type  : rsa
User public key name  : --
User public key type  : --
Sftp directory       : flash:
Service type         : stelnet
------------------------------------------------------------------------------
Total 2, 2 printed
```

配置脚本

- SSH服务器的配置脚本

```
#
sysname SSH Server
#
rsa peer-public-key rsakey001
 public-key-code begin
 3082010A
  02820101
    00A4BAB8 B964077E F7657F7F E4BE1DE8 71EE1707 E4EE2864 2D06FBE0 BFC1CB52
    F99B7A99 0132B709 3F841CA2 3544B8B2 6EE0A9ED 04B19FE3 FB3DA86D BE68FFE2
    2303108D BDC24B80 A1793A08 FDA0B6C1 13C31EA5 298EC9B1 2B0BC8BD 32CFF896
    29F8CA98 8B1724AF 5DA8A390 20906ADE 6A8AD77D 6234F0C8 DC965BA0 1771D9C0
    A89ED49B 5ECF7EE2 D5997527 FC87FE03 E51658C1 0996DFDF DC456376 2FA4B268
    4345131D 431419D2 DD5E4003 6A7D3295 145F3175 22E80686 E6B39A05 799D6BCF
    A78F69B6 BC2D0836 F5013421 77D68B89 A9EC182A 04B87BE3 500FCE14 9C95CF78
    75704359 0C70FD60 1EFC0B99 32F02142 4CE781E4 36A60BFC 2CBD07F6 9E700CEE
    4D
  0203
    010001
 public-key-code end
 peer-public-key end
#
aaa
 local-user client001 password irreversible-cipher $1d$v!=.5/:(q-$xL=\K
+if"'S}>k7vGP5$_ox0B@ys7.'DBHL~3*aN$
 local-user client001 service-type ssh
 local-user client001 privilege level 3
#
ssh server port 1025
stelnet server enable
ssh user client001
ssh user client001 authentication-type password
ssh user client001 service-type stelnet
ssh user client002
ssh user client002 authentication-type rsa
ssh user client002 assign rsa-key rsakey001
ssh user client002 service-type stelnet
ssh server-source all-interface
#
ssh server cipher aes128_ctr aes256_ctr aes192_ctr aes128_gcm aes256_gcm
ssh server hmac sha2_256 sha2_512
ssh server key-exchange dh_group_exchange_sha256 dh_group16_sha512
ssh server publickey rsa_sha2_256 rsa_sha2_512
ssh server dh-exchange min-len 3072
#
user-interface vty 0 4
 authentication-mode aaa
 protocol inbound ssh
#
return
```

- SSH客户端Client001的配置脚本

```
#
sysname client001
#
ssh client first-time enable
#
ssh client cipher aes128_ctr aes256_ctr aes192_ctr aes128_gcm aes256_gcm
ssh client hmac sha2_256 sha2_512
ssh client key-exchange dh_group_exchange_sha256 dh_group16_sha512
ssh client publickey rsa_sha2_256 rsa_sha2_512
#
return
```

- SSH客户端Client002的配置脚本

```
#
sysname client002
```

```
#
ssh client first-time enable
#
ssh client cipher aes128_ctr aes256_ctr aes192_ctr aes128_gcm aes256_gcm
ssh client hmac sha2_256 sha2_512
ssh client key-exchange dh_group_exchange_sha256 dh_group16_sha512
ssh client publickey rsa_sha2_256 rsa_sha2_512
#
return
```

2.2 系统管理

2.2.1 NTP

2.2.1.1 举例：配置带认证的 NTP 客户端/服务器模式

组网需求

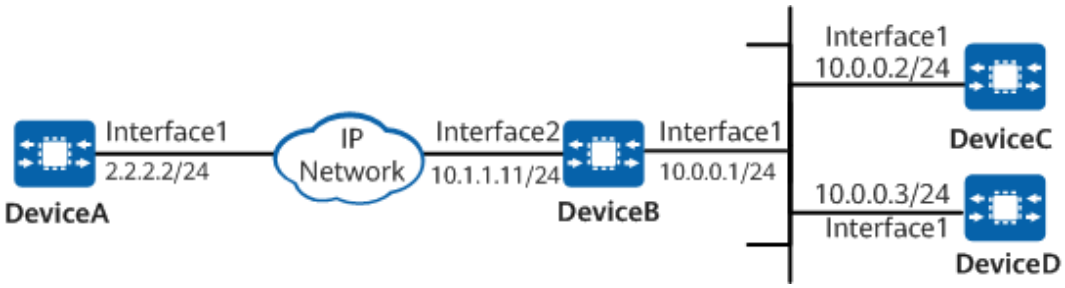
如图2-3所示。

- DeviceA作为NTP服务器，其本地时钟作为NTP主时钟，层数为2。
- DeviceB作为NTP客户端，同步远端服务器DeviceA的时钟。
- DeviceC和DeviceD作为NTP客户端，把DeviceB作为自己的NTP服务器。
- 在所有设备启用NTP认证。

图 2-3 NTP 客户端/服务器模式组网

说明

本例中Interface1和Interface2分别代表100GE1/0/1，100GE1/0/2。



配置注意事项

- 客户端必须先使能NTP认证，然后再指定NTP服务器地址，并同时指定发给服务器的验证密钥。否则将不进行验证，直接同步。
- 客户端与服务器端必须配置相同的验证密钥，并声明该密钥可信，否则无法通过验证。
- 在服务器和客户端均启用NTP认证。

配置思路

采用如下思路进行此案例的配置：

1. 配置DeviceA作为服务器，提供主时钟。
2. 配置DeviceB作为NTP客户端，同步DeviceA的时钟。
3. 配置DeviceC、DeviceD作为NTP客户端，同步DeviceB的时钟。
4. 在DeviceA、DeviceB、DeviceC和DeviceD上配置密码通过HMAC-SHC256算法加密后参与的NTP认证。

操作步骤

步骤1 配置各设备的IP地址并确保路由可达。

步骤2 在DeviceA配置NTP主时钟、侦听接口并启动验证功能。

在DeviceA上指定使用自己的本地时钟作为参考时钟，层数为2。

```
<DeviceA> system-view
[~DeviceA] ntp refclock-master 2
[*DeviceA] commit
```

在DeviceA上指定侦听接口。

```
[~DeviceA] ntp server source-interface 100ge 1/0/1
[*DeviceA] commit
```

在DeviceA上使能NTP认证功能并配置验证密钥。

```
[~DeviceA] ntp authentication enable
[*DeviceA] ntp authentication-keyid 42 authentication-mode hmac-sha256 ********
[*DeviceA] commit
```

说明

注意服务器端与客户端必须配置相同的验证密钥。

在DeviceA上使能NTP服务器功能。

```
[~DeviceA] undo ntp server disable
[*DeviceA] commit
```

步骤3 在DeviceB启动验证功能、指定侦听接口和NTP服务器。

在DeviceB上使能NTP认证功能、配置验证密钥并声明该密钥可信。

```
<DeviceB> system-view
[~DeviceB] ntp authentication enable
[*DeviceB] ntp authentication-keyid 42 authentication-mode hmac-sha256 ********
[*DeviceB] ntp trusted authentication-keyid 42
[*DeviceB] commit
```

在DeviceB上指定侦听接口。

```
[~DeviceB] ntp server source-interface 100ge 1/0/1
[*DeviceB] commit
```

DeviceB指定DeviceA为NTP服务器，并使用已配置的验证密钥。

```
[~DeviceB] ntp unicast-server 2.2.2.2 authentication-keyid 42
[*DeviceB] commit
```

在DeviceB上使能NTP服务器功能。

```
[~DeviceB] undo ntp server disable
[*DeviceB] commit
```

步骤4 在DeviceC启动验证功能并指定NTP服务器。

```
<DeviceC> system-view
[~DeviceC] ntp authentication enable
[*DeviceC] ntp authentication-keyid 42 authentication-mode hmac-sha256 ********
[*DeviceC] ntp trusted authentication-keyid 42
[*DeviceC] ntp unicast-server 10.0.0.1 authentication-keyid 42
[*DeviceC] commit
```

步骤5 在DeviceD启动验证功能并指定NTP服务器。

```
<DeviceD> system-view
[~DeviceD] ntp authentication enable
[*DeviceD] ntp authentication-keyid 42 authentication-mode hmac-sha256 ********
[*DeviceD] ntp trusted authentication-keyid 42
[*DeviceD] ntp unicast-server 10.0.0.1 authentication-keyid 42
[*DeviceD] commit
```

----结束

检查配置结果

查看DeviceB的NTP状态，可以看到时钟状态为“synchronized”，即已经完成同步。时钟的层数为3，比服务器DeviceA低1级。

```
[~DeviceB] display ntp status
 clock status: synchronized
 clock stratum: 3
 reference clock ID: 2.2.2.2
 nominal frequency: 60.0002 Hz
 actual frequency: 60.0002 Hz
 clock precision: 2^18
 clock offset: 3.8128 ms
 root delay: 31.26 ms
 root dispersion: 74.20 ms
 peer dispersion: 34.30 ms
 reference time: 11:55:56.833 UTC Feb 2 2020(C7B15BCC.D5604189)
 synchronization state: clock synchronized
```

查看DeviceC的NTP状态，可以看到时钟状态为“synchronized”，即，已经完成同步。时钟的层数为4，比服务器DeviceB低1级。

```
[~DeviceC] display ntp status
 clock status: synchronized
 clock stratum: 4
 reference clock ID: 10.0.0.1
 nominal frequency: 60.0002 Hz
 actual frequency: 60.0002 Hz
 clock precision: 2^18
 clock offset: 3.8128 ms
 root delay: 31.26 ms
 root dispersion: 74.20 ms
 peer dispersion: 34.30 ms
 reference time: 11:55:56.833 UTC Feb 2 2020(C7B15BCC.D5604189)
 synchronization state: clock synchronized
```

查看DeviceD的NTP状态，可以看到时钟状态为“synchronized”，即已经完成同步。时钟的层数为4，比服务器DeviceB低1级。

```
[~DeviceD] display ntp status
 clock status: synchronized
 clock stratum: 4
 reference clock ID: 10.0.0.1
 nominal frequency: 60.0002 Hz
 actual frequency: 60.0002 Hz
 clock precision: 2^18
```



```
clock offset: 3.8128 ms
root delay: 31.26 ms
root dispersion: 74.20 ms
peer dispersion: 34.30 ms
reference time: 11:55:56.833 UTC Feb 2 2020(C7B15BCC.D5604189)
synchronization state: clock synchronized
```

查看DeviceA的NTP状态。

```
[~DeviceA] display ntp status
clock status: synchronized
clock stratum: 2
reference clock ID: LOCAL(0)
nominal frequency: 60.0002 Hz
actual frequency: 60.0002 Hz
clock precision: 2^18
clock offset: 0.0000 ms
root delay: 0.00 ms
root dispersion: 26.50 ms
peer dispersion: 10.00 ms
reference time: 12:01:48.377 UTC Feb 2 2020(C7B15D2C.60A15981)
synchronization state: clock synchronized
```

配置脚本

- DeviceA

```
#
 sysname DeviceA
#
ntp authentication-keyid 42 authentication-mode hmac-sha256 cipher %+%#JA!v6M22=Gg\{>U.lx
%#)c%yY}0*"/`5mi><QS)L%+%#
ntp refclock-master 2
ntp authentication enable
ntp server source-interface 100GE1/0/1
#
interface 100GE1/0/1
 undo portswitch
 ip address 2.2.2.2 255.255.255.0
#
return
```

- DeviceB

```
#
 sysname DeviceB
#
ntp authentication-keyid 42 authentication-mode hmac-sha256 cipher %+%#>hD8))_H-XZVut2u3!
_0lq3,+Ph=:OE}pX;T2M'9%+%#
ntp trusted authentication-keyid 42
ntp unicast-server 2.2.2.2 authentication-keyid 42
ntp authentication enable
ntp server source-interface 100GE1/0/1
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.0.0.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.1.11 255.255.255.0
#
return
```

- DeviceC

```
#
 sysname DeviceC
#
ntp authentication-keyid 42 authentication-mode hmac-sha256 cipher %+
%#m:fVJfk*r&3x"1J`21^K`Y;LH;B+g(t2<ZX^}Q_~%+%#
```



```
ntp trusted authentication-keyid 42
ntp unicast-server 10.0.0.1 authentication-keyid 42
ntp authentication enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.0.0.2 255.255.255.0
#
return
```

- DeviceD

```
#
 sysname DeviceD
#
ntp authentication-keyid 42 authentication-mode hmac-sha256 cipher %+%#$
\`_6BKWy1]kdR@=c;O@UX!)Vor5iYi|zIYEG_v5%+%#
ntp trusted authentication-keyid 42
ntp unicast-server 10.0.0.1 authentication-keyid 42
ntp authentication enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.0.0.3 255.255.255.0
#
return
```

2.2.2 LLDP

2.2.2.1 举例：配置 LLDP 基本功能

组网需求

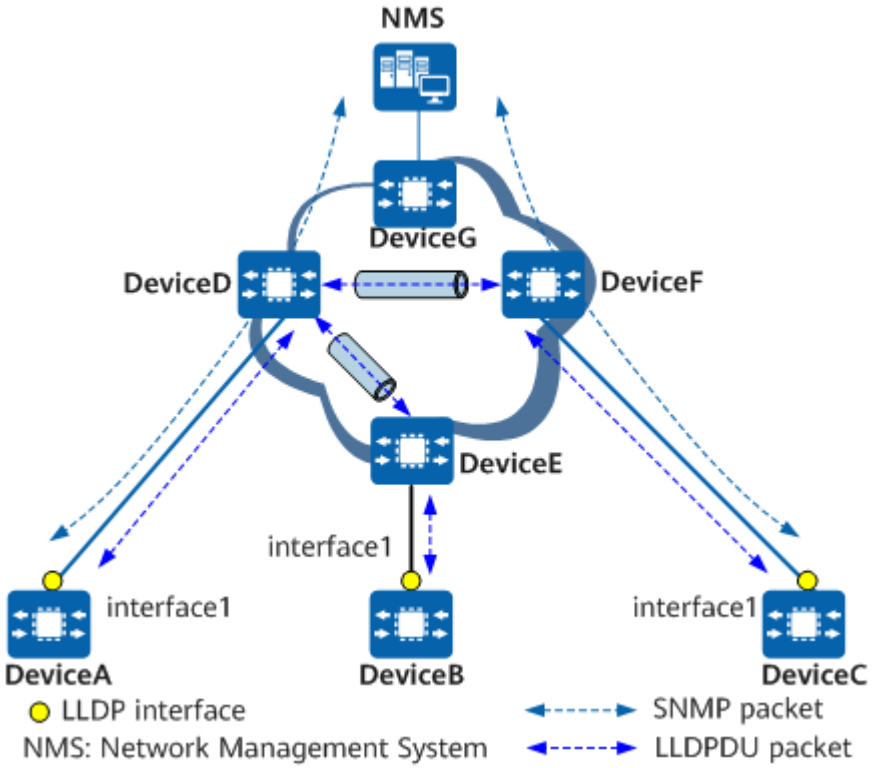
如图2-4所示，DeviceA和DeviceB、DeviceC之间有可达链路，DeviceA、DeviceC与NMS之间有可达路由。在配置LLDP功能之前，DeviceA无法获取DeviceB和DeviceC的状态信息，NMS也不能通过SNMP报文获取DeviceA、DeviceB和DeviceC之间的拓扑结构信息。

配置LLDP功能之后，设备间可以通过交互LLDP报文获取彼此的状态信息。同时NMS可以通过LLDP管理IP地址查找DeviceA、DeviceB和DeviceC，获取它们之间的拓扑信息。

图 2-4 LLDP 基本功能配置组网图

说明

本例中interface1代表100GE1/0/1。



配置思路

采用如下思路进行本例的配置：

1. 配置DeviceA、DeviceB和DeviceC的相关接口的IP地址和路由协议，使网络层路由可达。
2. 启用DeviceA、DeviceB和DeviceC的全局LLDP功能。
3. 配置DeviceA、DeviceB和DeviceC的管理IP地址，该IP地址用于在邻居节点中标识本端设备。
4. 配置DeviceA、DeviceB和DeviceC的LLDP相关参数，优化LLDP的性能。
5. 配置DeviceA、DeviceB和DeviceC的LLDP告警功能，合理配置设备发送LLDP告警的延迟时间，既能达到告警的目的，也能降低系统资源的消耗。

操作步骤

步骤1 如图2-4所示，配置相关接口的IP地址和路由协议，具体的配置过程略。

步骤2 启用DeviceA、DeviceB和DeviceC的全局LLDP功能。

配置DeviceA。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] lldp enable
[~DeviceA] commit
```

配置DeviceB。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*HUAWEI] commit
```

```
[~DeviceB] lldp enable
[~DeviceB] commit
```

配置DeviceC。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceC
[*HUAWEI] commit
[~DeviceC] lldp enable
[~DeviceC] commit
```

步骤3 配置DeviceA、DeviceB和DeviceC的管理IP地址。

配置DeviceA的管理IP地址为10.10.10.1。

```
[~DeviceA] lldp management-address 10.10.10.1
[*DeviceA] commit
```

配置DeviceB的管理IP地址为10.10.10.2。

```
[~DeviceB] lldp management-address 10.10.10.2
[*DeviceB] commit
```

配置DeviceC的管理IP地址为10.10.10.3。

```
[~DeviceC] lldp management-address 10.10.10.3
[*DeviceC] commit
```

步骤4 配置DeviceA、DeviceB和DeviceC的LLDP相关参数，包括设备发送LLDP报文的周期和延迟时间。

配置DeviceA发送LLDP报文的周期和延迟时间。

```
[~DeviceA] lldp transmit interval 60
[*DeviceA] lldp transmit delay 9
[*DeviceA] commit
```

配置DeviceB和DeviceC发送LLDP报文的周期和延迟时间。

DeviceB和DeviceC的配置同DeviceA，具体请参考配置脚本，此处不再赘述。

步骤5 启用DeviceA、DeviceB和DeviceC的LLDP告警功能，并配置设备发送LLDP告警的延迟时间。

配置DeviceA。

```
[~DeviceA] snmp-agent trap enable feature-name lldp
[~DeviceA] lldp trap-interval 10
[*DeviceA] commit
```

配置DeviceB和DeviceC。

DeviceB和DeviceC的配置同DeviceA，具体请参考配置脚本，此处不再赘述。

----结束

检查配置结果

查看DeviceA的LLDP是否启用、LLDP管理地址是否配置、LLDP告警功能是否启用以及LLDP属性的值是否为配置的值。

查看DeviceA的本地LLDP信息。

```
[DeviceA] display lldp local
System information
-------------------------------------------------------------------------
Chassis type                :Mac Address
```



```
Chassis ID                      :00e0-fc21-1220
System name                     :DeviceA
System description              :Huawei Versatile Routing Platform Software
VRP (R) software,Version 8.22.0.0 (CloudEngine 8800, 6800系列 V300R023C00)
Copyright (C) 2012-2020 Huawei Technologies Co., Ltd.
HUAWEI CloudEngine 8800, 6800系列

System capabilities supported     :bridge router
System capabilities enabled       :bridge router
LLDP Up time                    :2020/02/26 15:08:28

System configuration
------------------------------------------------------------------------------
LLDP Status                     :enabled            (default is disabled)
LLDP Message Tx Interval        :30                 (default is 30s)
LLDP Message Tx Hold Multiplier :4                  (default is 4)
LLDP Refresh Delay              :2                  (default is 2s)
LLDP Tx Delay                   :2                  (default is 2s)
LLDP Notification Interval      :5                  (default is 5s)
LLDP Notification Enable        :enabled            (default is enabled)
Management Address              :IPv4: 10.10.10.1
LLDP Fast Message Count         :4                  (default is 4)

Remote Table Statistics:
------------------------------------------------------------------------------
Remote Table Last Change Time   :0 days,0 hours, 11 minutes,49 seconds
Remote Neighbors Added          :0
Remote Neighbors Deleted        :0
Remote Neighbors Dropped        :0
Remote Neighbors Aged           :0
Total Neighbors                 :2

Port information:
------------------------------------------------------------------------------
Interface 100GE1/0/1:
LLDP Enable Status              :txAndRx            (default is disabled)
Total Neighbors                 :2

Port ID subtype                 :Interface Name
Port ID                         :100GE1/0/1
Port description                :HUAWEI,  100GE1/0/1 Interface

Port and Protocol VLAN ID(PPVID)  :unsupported
Port VLAN ID(PVID)              :1
VLAN name of VLAN 1             :VLAN1
Protocol identity               :LACP
Auto-negotiation supported      :Yes
Auto-negotiation enabled        :No
OperMau                         :speed (10000) /duplex (Full)
Link aggregation supported      :Yes
Link aggregation enabled        :No
Aggregation port ID             :0
Maximum frame Size              :9216
```

查看DeviceA的邻居设备的LLDP信息。

```
[DeviceA] display lldp neighbor interface 100GE1/0/1
100GE1/0/1 has 2 neighbor(s):

Neighbor index                  :1
Chassis type                    :MAC Address
Chassis ID                      :00e0-fc11-1220
Port ID subtype                 :Interface Name
Port ID                         :100GE1/0/1
Port description                :HUAWEI,  100GE1/0/1 Interface
System name                     :DeviceB
System description              :Huawei Versatile Routing Platform Software
VRP (R) software, Version 8.22.0.0 (CloudEngine 8800, 6800系列 V300R023C00)
Copyright (C) 2012-2020 Huawei Technologies Co., Ltd.
HUAWEI CloudEngine 8800, 6800系列
```

```
System capabilities supported      :bridge router
System capabilities enabled        :bridge router
Management address type             :IPv4
Management address                  :10.10.10.2
Expired time                      :104 (s)

Port VLAN ID(PVID)                  :1
Port And Protocol VLAN ID(PPVID)   :unsupported
VLAN name of VLAN 0                 :VLAN0
Protocol identity                 :LACP
Auto-negotiation supported         :Yes
Auto-negotiation enabled           :No
OperMau                          :speed (10000) /duplex (Full)
Link aggregation supported         :Yes
Link aggregation enabled           :No
Aggregation port ID                :0
Maximum frame Size                  :0
Discovered time                   :2020-02-21 11:09:15

Network Card ID                    :--

Neighbor index                     :2
Chassis type                      :MAC Address
Chassis ID                        :00e0-fc33-0013
Port ID type                      :Interface Name
Port ID                          :100GE1/0/1
Port description                   :HUAWEI,  100GE1/0/1 Interface
System name                        :DeviceC
System description                 :Huawei Versatile Routing Platform Software
VRP (R) software, Version 8.22.0.0 (CloudEngine 8800, 6800系列 V300R023C00)
Copyright (C) 2012-2015 Huawei Technologies Co., Ltd.
HUAWEI CloudEngine 8800, 6800系列

System capabilities supported      :bridge router
System capabilities enabled        :bridge router
Management address type  :IPv4
Management address                  :10.10.10.3
Expired time                      :104 (s)

Port VLAN ID(PVID)                  :1
Port And Protocol VLAN ID(PPVID)   :unsupported
VLAN name of VLAN 0                 :VLAN0
Protocol identity                 :LACP
Auto-negotiation supported         :Yes
Auto-negotiation enabled           :No
OperMau                          :speed (10000) /duplex (Full)
Link aggregation supported         :Yes
Link aggregation enabled           :No
Aggregation port ID                :0
Maximum frame Size                  :0
Discovered time                   :2020-02-21 11:09:15
```

查看DeviceB和DeviceC的LLDP功能是否启用、LLDP管理地址是否配置。

请参见查看DeviceA的配置信息的过程。

配置脚本

- DeviceA

```
#
 sysname DeviceA
#
lldp enable
lldp transmit interval 60
lldp transmit delay 9
lldp restart 3
lldp fast-count 3
#
interface 100GE1/0/1
```



```
 undo portswitch
 ip address 10.10.10.1 255.255.255.0
#
 lldp management-address  10.10.10.1
#
return
```

- DeviceB

```
#
 sysname DeviceB
#
lldp enable
lldp transmit interval 60
lldp transmit delay 9
lldp restart 3
lldp fast-count 3
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.10.10.2 255.255.255.0
#
 lldp management-address  10.10.10.2
#
return
```

- DeviceC

```
#
 sysname DeviceC
#
lldp enable
lldp transmit interval 60
lldp transmit delay 9
lldp restart 3
lldp fast-count 3
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.10.10.3 255.255.255.0
#
 lldp management-address  10.10.10.3
#
return
```

2.2.3 SNMP

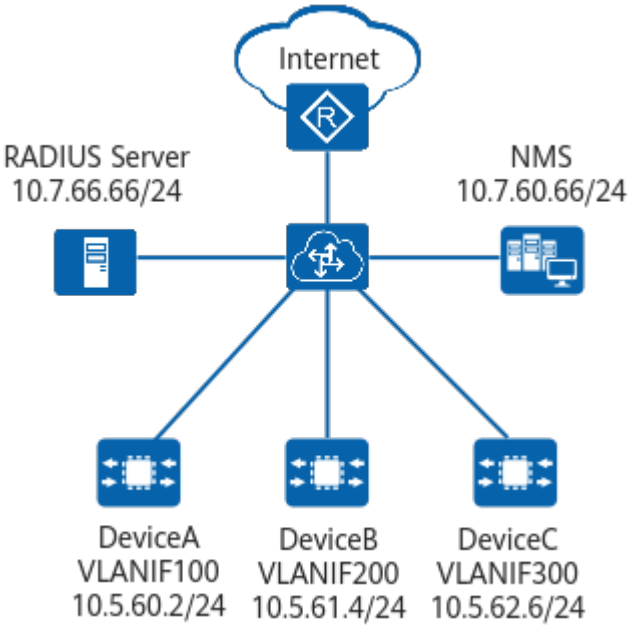
2.2.3.1 举例：数据中心网络管理（RADIUS 认证方式）

组网需求

某企业数据中心网络比较复杂，为了保证数据中心网络的安全和稳定性，需要对网络实时监控，并限制管理员的登录权限。此时可以部署一个综合的数据中心网络管理系统，满足网络监控和管理员受限接入的要求。

如图2-5所示，网络中的设备已经配置IP地址，且与RADIUS服务器以及网络管理系统(NMS)之间路由可达。所有用户登录设备时都需要通过RADIUS认证。NMS对整个网络进行监控，接收来自每台设备的告警和日志信息。

图 2-5 数据中心网络管理综合举例（RADIUS 认证方式）



配置思路

数据中心网络管理综合举例的配置思路如下：

1. 配置RADIUS协议，实现RADIUS认证。用户通过STelnet登录时使用RADIUS服务器上配置的用户名和密码，从而保证用户登录的安全性。
2. 配置STelnet登录设备。STelnet协议实现在不安全网络上提供安全的远程登录，保证了数据的完整性和可靠性，保证了数据的安全传输。
3. 配置SNMP功能。使用SNMPv3版本的认证和加密方式，保证设备和NMS连接的安全性。从而实现通过NMS对网络中的设备进行集中管理。
4. 配置将设备日志和告警信息通过SNMP发送到NMS，实现对网络的监控。

说明

以下配置仅以DeviceA为例。其他设备的配置请参考DeviceA。

请确保RADIUS服务器模板内的RADIUS服务器的地址、端口号和共享密钥配置正确，并且和RADIUS服务器保持一致。

请确保已在RADIUS服务器上配置了用户，本例中假设RADIUS服务器上已配置了用户名为admin@admin123，密码为YsHsjx_202206的用户。

如果RADIUS服务器上配置的用户较多，建议用户使用**ssh authentication-type default password**命令，对本地用户使用预设密码认证方式，从而简化配置。

操作步骤

步骤1 配置RADIUS。

1. 配置RADIUS模板。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] radius-server template shiva
[*DeviceA-radius-shiva] radius-server authentication 10.7.66.66 1812          //配置RADIUS服务器的地址和端口号。
```

```
[*DeviceA-radius-shiva] radius-server shared-key cipher hello                    //配置RADIUS服务器的共享密钥。
[*DeviceA-radius-shiva] radius-server retransmit 2                               //配置超时重传次数为2。
[*DeviceA-radius-shiva] quit
```

2. 创建AAA认证方案“auth”并配置认证方式为RADIUS。

```
[*DeviceA] aaa
[*DeviceA-aaa] authentication-scheme auth
[*DeviceA-aaa-authen-auth] authentication-mode radius
[*DeviceA-aaa-authen-auth] quit
```

3. 创建域“admin123”并在域内绑定AAA认证方案“auth”和RADIUS服务器模板“shiva”。

```
[*DeviceA-aaa] domain admin123
[*DeviceA-aaa-domain-admin123] authentication-scheme auth
[*DeviceA-aaa-domain-admin123] radius-server shiva
[*DeviceA-aaa-domain-admin123] quit
[*DeviceA-aaa] quit
[*DeviceA] commit
```

步骤2 配置STelnet。

1. 使能设备支持STelnet。

```
[~DeviceA] rsa local-key-pair create
The key name will be: DeviceA_Host
The range of public key size is (2048, 4096).
NOTE: Key pair generation will take a short while.
Please input the modulus [default = 3072]:3072 //执行本命令后，会提示用户输入生成的RSA密钥对长度，当前支持模数长度为2048比特位、3072比特位和4096比特位三种RSA密钥对。如果用户没有输入密钥对长度，直接回车，则会生成3072位RSA密钥对；如果用户没有任何操作，则设备放弃生成RSA密钥对。建议使用3072位及以上更安全的RSA密钥对。
[*DeviceA] stelnet server enable
```

2. 配置SSH用户登录的用户界面。

```
[*DeviceA] user-interface vty 0 4
[*DeviceA-ui-vty0-4] authentication-mode aaa
[*DeviceA-ui-vty0-4] protocol inbound ssh
[*DeviceA-ui-vty0-4] user privilege level 3
[*DeviceA-ui-vty0-4] quit
```

3. 配置SSH用户，用户名为admin@admin123。

```
[*DeviceA] ssh user admin@admin123 authentication-type password
[*DeviceA] ssh user admin@admin123 service-type stelnet
[*DeviceA] commit
```

步骤3 配置SNMP功能。

1. 配置SNMP与NMS的连接。

```
[~DeviceA] snmp-agent sys-info version v3
[*DeviceA] snmp-agent mib-view included iso-view iso
[*DeviceA] snmp-agent group v3 admingroup privacy write-view iso-view notify-view iso-view
[*DeviceA] snmp-agent usm-user v3 adminuser admingroup authentication-mode sha Admin@1234 privacy-mode aes128 Helloworld@6789  //认证算法有MD5和SHA两种，SHA安全性高，MD5运行速度块。本举例使用sha。加密算法有3DES168、AES128、AES192、AES256和DES56。ASE的加密方式安全性高。本举例使用AES128。
```

2. 配置告警主机。

```
[*DeviceA] snmp-agent target-host host-name nms trap address udp-domain 10.7.60.66 params securityname adminuser v3 privacy  //告警主机安全级别需要高于或等于用户的安全级别。此处配置为privacy（认证且加密）。
[*DeviceA] commit
```

步骤4 配置将设备日志和告警信息通过SNMP发送到NMS。

```
[~DeviceA] info-center source default channel 5 log state on
[*DeviceA] commit
```

----结束

验证

对于配置的验证主要通过以下两个方面完成：

- 能够通过RADIUS服务器上配置的用户名和密码实现STelnet登录设备。
- NMS和设备连接成功。NMS能够通过SNMP对设备操作，且能够接收到日志和告警信息。

配置脚本

DeviceA的配置脚本

```
#
sysname DeviceA
#
info-center source default channel 5 log state on
#
radius-server template shiva
 radius-server shared-key cipher %^%#L@71VU/>5>n/c$GKI>J!i:Uz~:!<.W'jc0X@nE4$%^%#  //此处密文格式仅为示例，不同版本之间可能存在不同
 radius-server authentication 10.7.66.66 1812
 radius-server retransmit 2
#
aaa
 authentication-scheme auth
  authentication-mode radius
 domain admin123
  authentication-scheme auth
  radius-server shiva
#
snmp-agent
snmp-agent local-engineid 800007DB03306B20792201
#
snmp-agent sys-info version v3
snmp-agent group v3 admingroup privacy write-view iso-view notify-view iso-view
snmp-agent target-host host-name nms trap address udp-domain 10.7.60.66 params securityname adminuser v3 privacy
#
snmp-agent mib-view included iso-view iso
snmp-agent usm-user v3 adminuser
snmp-agent usm-user v3 adminuser group admingroup
snmp-agent usm-user v3 adminuser authentication-mode sha cipher %^%#BQV1%E-zm5`pG^HCe.4-yi-EUx$iv=S(jiKO7tJN%^%#  //此处密文格式仅为示例，不同版本之间可能存在不同
snmp-agent usm-user v3 adminuser privacy-mode aes128 cipher %^%#4_o.,z8`_OmbfU4svg>8"[TxSo\9'R]d/[TXR3!&%^%#  //此处密文格式仅为示例，不同版本之间可能存在不同
#
stelnet server enable
ssh user admin@admin123
ssh user admin@admin123 authentication-type password
ssh user admin@admin123 service-type stelnet
ssh authorization-type default aaa
#
user-interface vty 0 4
 authentication-mode aaa
 user privilege level 3
 protocol inbound ssh
#
return
```

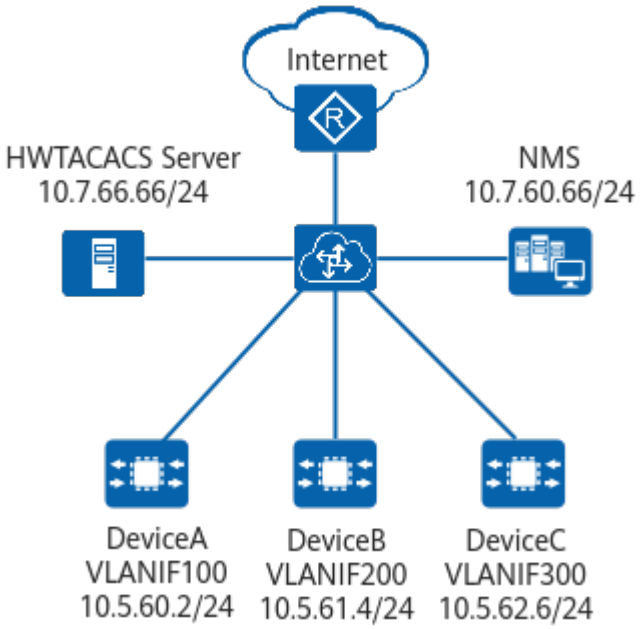
2.2.3.2 举例：数据中心网络管理（HWTACACS 认证方式）

组网需求

某企业数据中心网络比较复杂，为了保证数据中心网络的安全和稳定性，需要对网络实时监控，并限制管理员的登录权限。此时可以部署一个综合的数据中心网络管理系统，满足网络监控和管理员受限接入的要求。

如图2-6所示，网络中的设备已经配置IP地址，且与HWTACACS服务器以及网络管理系统(NMS)之间路由可达。所有用户登录设备时都需要通过HWTACACS认证。网络管理系统(NMS)对整个网络进行监控，接收来自每台设备的告警和日志信息。

图 2-6 数据中心网络管理综合举例（HWTACACS 认证方式）



配置思路

数据中心网络管理综合举例的配置思路如下：

1. 配置HWTACACS协议，实现HWTACACS认证。用户通过STelnet登录时使用HWTACACS服务器上配置的用户名和密码，从而保证用户登录的安全性。
2. 配置STelnet登录设备。STelnet协议实现在不安全网络上提供安全的远程登录，保证了数据的完整性和可靠性，保证了数据的安全传输。
3. 配置SNMP功能。使用SNMPv3版本的认证和加密方式，保证设备和NMS连接的安全性。从而实现通过NMS对网络中的设备进行集中管理。
4. 配置将设备日志和告警信息通过SNMP发送到NMS，实现对网络的监控。

说明

以下配置仅以DeviceA为例。其他设备的配置请参考DeviceA。

请确保HWTACACS服务器模板内的HWTACACS服务器的地址、端口号和共享密钥配置正确，并且和HWTACACS服务器保持一致。

请确保已在HWTACACS服务器上配置了用户，本例中假设HWTACACS服务器上已配置了用户名为admin@admin123，密码为huawei@1234的用户。

如果HWTACACS服务器上配置的用户较多，建议使用**ssh authentication-type default password**命令，对本地用户使用预设密码认证方式，从而简化配置。

操作步骤

步骤1 配置HWTACACS。

1. 配置HWTACACS服务器模板。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] hwtacacs-server template ht
[*DeviceA-hwtacacs-ht] hwtacacs-server authentication 10.7.66.66 49        //配置HWTACACS服务器的地址和端口号。
[*DeviceA-hwtacacs-ht] hwtacacs-server shared-key cipher hello             //配置HWTACACS服务器的共享密钥。
[*DeviceA-hwtacacs-ht] quit
```

2. 创建AAA认证方案“auth”并配置认证方式为HWTACACS。

```
[*DeviceA] aaa
[*DeviceA-aaa] authentication-scheme auth
[*DeviceA-aaa-authen-auth] authentication-mode hwtacacs
[*DeviceA-aaa-authen-auth] quit
```

3. 创建域“admin123”并在域内绑定AAA认证方案“auth”和HWTACACS服务器模板“ht”。

```
[*DeviceA-aaa] domain admin123
[*DeviceA-aaa-domain-admin123] authentication-scheme auth
[*DeviceA-aaa-domain-admin123] hwtacacs-server ht
[*DeviceA-aaa-domain-admin123] quit
[*DeviceA-aaa] quit
[*DeviceA] commit
```

步骤2 配置STelnet。

1. 使能设备支持STelnet。

```
[~DeviceA] rsa local-key-pair create
The key name will be: DeviceA_Host
The range of public key size is (2048, 4096).
NOTE: Key pair generation will take a short while.
Please input the modulus [default = 3072]:3072 //执行本命令后，会提示用户输入生成的RSA密钥对长度，当前支持模数长度为2048比特位、3072比特位和4096比特位三种RSA密钥对。如果用户没有输入密钥对长度，直接回车，则会生成3072位RSA密钥对；如果用户没有任何操作，则设备放弃生成RSA密钥对。建议使用3072位及以上更安全的RSA密钥对。
[*DeviceA] stelnet server enable
```

2. 配置SSH用户登录的用户界面。

```
[*DeviceA] user-interface vty 0 4
[*DeviceA-ui-vty0-4] authentication-mode aaa
[*DeviceA-ui-vty0-4] protocol inbound ssh
[*DeviceA-ui-vty0-4] user privilege level 3
[*DeviceA-ui-vty0-4] quit
```

3. 配置SSH用户，用户名为admin@admin123。

```
[*DeviceA] ssh user admin@admin123 authentication-type password
[*DeviceA] ssh user admin@admin123 service-type stelnet
[*DeviceA] commit
```

步骤3 配置SNMP功能。

1. 配置SNMP与NMS的连接。

```
[~DeviceA] snmp-agent sys-info version v3
[*DeviceA] snmp-agent mib-view included iso-view iso
[*DeviceA] snmp-agent group v3 admingroup privacy write-view iso-view notify-view iso-view
[*DeviceA] snmp-agent usm-user v3 adminuser admingroup authentication-mode sha
Admin@1234 privacy-mode aes128 Helloworld@6789  //认证算法有MD5和SHA两种，SHA安全性高，MD5运行速度块。本举例使用MD5。加密算法有3DES168、AES128、AES192、AES256和DES56。ASE的加密方式安全性高。本举例使用AES128。
```

2. 配置告警主机。

```
[*DeviceA] snmp-agent target-host host-name nms trap address udp-domain 10.7.60.66 params securityname adminuser v3 privacy  //告警主机安全级别需要高于或等于用户的安全级别。此处配置为privacy（认证且加密）。
[*DeviceA] commit
```

步骤4 配置将设备日志和告警信息通过SNMP发送到NMS。

```
[~DeviceA] info-center source default channel 5 log state on
[*DeviceA] commit
```

----结束

验证

对于配置的验证主要通过以下两个方面完成：

- 能够通过HWTACACS服务器上配置的用户名和密码实现STelnet登录。
- NMS和设备连接成功。NMS能够通过SNMP对设备操作，且能够接收到日志和告警信息。

配置脚本

DeviceA的配置脚本

```
#
sysname DeviceA
#
info-center source default channel 5 log state on
#
hwtacacs-server template ht
 hwtacacs-server authentication 10.7.66.66
 hwtacacs-server shared-key cipher %^%#ysFK('!^0Wz][c#{!F(O]=t6.;g.'>E49.;k#gd<%^%#  //此处密文格式仅为示例，不同版本之间可能存在不同
#
aaa
 authentication-scheme auth
  authentication-mode hwtacacs
 domain admin123
  authentication-scheme auth
  hwtacacs-server ht
#
snmp-agent
snmp-agent local-engineid 800007DB03306B20792201
#
snmp-agent sys-info version v3
snmp-agent group v3 admingroup privacy write-view iso-view notify-view iso-view
snmp-agent target-host host-name nms trap address udp-domain 10.7.60.66 params securityname adminuser v3 privacy
#
snmp-agent mib-view included iso-view iso
snmp-agent usm-user v3 adminuser
snmp-agent usm-user v3 adminuser group admingroup
snmp-agent usm-user v3 adminuser authentication-mode sha cipher %^%#/d6nQ7mD^%v]l%(F!H_0Z=2L>3&cJ.G]Yt=:YdN0%^%#  //此处密文格式仅为示例，不同版本之间可能存在不同
snmp-agent usm-user v3 adminuser privacy-mode aes128 cipher %^%#
```

```
\v7aU_Bx6QYP[SP)*B'ARgceMAS<D<BxG7AMhv(;%^%#  //此处密文格式仅为示例，不同版本之间可能存在不同
#
stelnet server enable
ssh user admin@admin123
ssh user admin@admin123 authentication-type password
ssh user admin@admin123 service-type stelnet
ssh authorization-type default aaa
#
user-interface vty 0 4
 authentication-mode aaa
 user privilege level 3
 protocol inbound ssh
#
return
```

2.2.4 BootLoader 管理

2.2.4.1 举例：通过 BootLoader 清除 Console 口密码

组网需求

用户通过Console口登录设备时，如果遗忘了Console口的登录密码，则无法登录设备。此时，可以通过BootLoader菜单清除Console口的用户登录密码。如图1所示，用户PC的串口与设备的Console口已连接，设备已上电。

说明

具体设备的实际回显请以设备实际显示为准。此处的显示信息仅为举例。

图 2-7 通过 Console 口连接设备组网图

操作步骤

步骤1 重启设备。在设备启动过程中，当界面出现**Press Ctrl+B to enter BOOT menu**提示信息时，请三秒内按下快捷键“Ctrl+B”进入BootLoader主菜单。

```
Press Ctrl+B to enter BOOT menu: 3
Info: The password is empty. For security purposes, change the password.

New password:
Confirm password:
Warning: The bootloader password will be written to the
device.
Continue now? Yes(y) or No(n): y

The password is changed successfully.

    Main Menu

  1. Default startup
  2. Serial submenu
  3. Ethernet submenu
  4. Startup parameters submenu
  5. List file
  6. Password manager submenu
  7. DFX submenu
```

```
   8. Reboot
Enter your choice(1-8):
```

步骤2 在BootLoader主菜单中，输入“6”，进入“Password manager submenu”密码管理子菜单。

```
      Main Menu

   1. Default startup
   2. Serial submenu
   3. Ethernet submenu
   4. Startup parameters submenu
   5. List file
   6. Password manager submenu
   7. DFX submenu
   8. Reboot
Enter your choice(1-8): 6          //进入密码管理子菜单

      Password manager submenu

   1. Modify bootloader password
   2. Clear the console login password
   3. Reset bootloader password
   0. Return

Enter your choice(0-3):
```

步骤3 在密码管理子菜单中，输入“2”，进入“Clear the console login password”清空Console口登录密码菜单。

```
      Password manager submenu

   1. Modify bootloader password
   2. Clear the console login password
   3. Reset bootloader password
   0. Return

 Enter your choice(0-3): 2

 Caution: A new console password must be set after the restart.
 Continue now? Yes(y) or No(n):
```

步骤4 在清空Console口登录密码菜单中，输入“**y**”后，让设备继续启动。

```
Caution: A new console password must be set after the restart.
Continue now? Yes(y) or No(n): y
Password:    //输入BootLoader密码，然后按Enter键让设备继续启动
```

步骤5 设备启动后，请在登录后重新设置Console口密码。

```
<HUAWEI> system-view
[~HUAWEI] user-interface console 0
[~HUAWEI-ui-console0] authentication-mode password
[*HUAWEI-ui-console0] set authentication password
Please configure the login password (8-16)
Enter Password:
Confirm Password:
[~HUAWEI-ui-console0] commit
[~HUAWEI-ui-console0] return
```

步骤6 为了防止重启后配置丢失，请保存配置。

```
<HUAWEI> save
 Warning: The current configuration will be written to the device. Continue? [Y/N]:y
 Now saving the current configuration to the slot 1
 Info: Save the configuration successfully.
```

----结束

检查配置结果

配置完成后，后续用户通过Console口登录设备时可以使用新密码登录。

2.3 以太网交换

2.3.1 VLAN

2.3.1.1 举例：配置基于接口划分 VLAN，实现同一 VLAN 内的互通（跨设备）

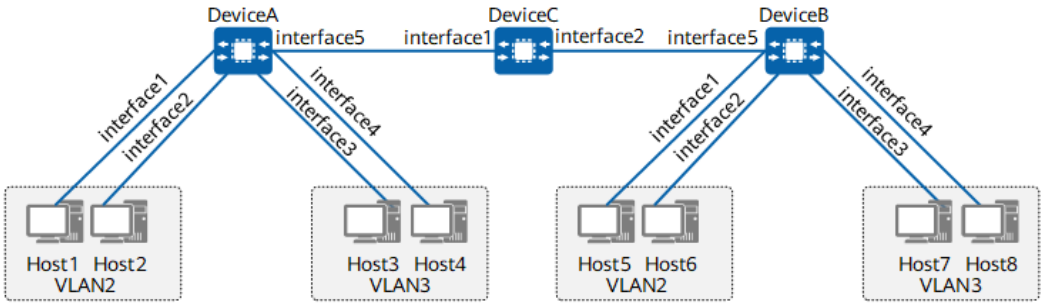
组网需求

如图2-8所示，把Host1、Host2、Host5、Host6划分到VLAN2，把Host3、Host4、Host7、Host8划分到VLAN3。DeviceA与DeviceC、DeviceC与DeviceB之间相连的链路允许VLAN2和VLAN3的报文通过。希望实现DeviceA和DeviceB下属于同一VLAN内的主机可以直接通信，属于不同VLAN间的主机不能直接进行二层通信。

图 2-8 基于接口划分 VLAN 组网图（跨设备）

说明

本例中interface1、interface2、interface3、interface4、interface5分别代表100GE1/0/1、100GE1/0/2、100GE1/0/3、100GE1/0/4、100GE1/0/5。



操作步骤

步骤1 配置DeviceA、DeviceB与主机相连的接口为Access类型的接口，并将Host1、Host2、Host5、Host6划分到VLAN2，将Host3、Host4、Host7、Host8划分到VLAN3。

配置DeviceA。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] vlan batch 2 3
[*DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] portswitch
[*DeviceA-100GE1/0/1] port link-type access
[*DeviceA-100GE1/0/1] port default vlan 2
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] portswitch
[*DeviceA-100GE1/0/2] port link-type access
```

```
[*DeviceA-100GE1/0/2] port default vlan 2
[*DeviceA-100GE1/0/2] quit
[*DeviceA] interface 100ge 1/0/3
[*DeviceA-100GE1/0/3] portswitch
[*DeviceA-100GE1/0/3] port link-type access
[*DeviceA-100GE1/0/3] port default vlan 3
[*DeviceA-100GE1/0/3] quit
[*DeviceA] interface 100ge 1/0/4
[*DeviceA-100GE1/0/4] portswitch
[*DeviceA-100GE1/0/4] port link-type access
[*DeviceA-100GE1/0/4] port default vlan 3
[*DeviceA-100GE1/0/4] quit
[*DeviceA] commit
```

配置DeviceB。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*HUAWEI] commit
[~DeviceB] vlan batch 2 3
[*DeviceB] interface 100ge 1/0/1
[*DeviceB-100GE1/0/1] portswitch
[*DeviceB-100GE1/0/1] port link-type access
[*DeviceB-100GE1/0/1] port default vlan 2
[*DeviceB-100GE1/0/1] quit
[*DeviceB] interface 100ge 1/0/2
[*DeviceB-100GE1/0/2] portswitch
[*DeviceB-100GE1/0/2] port link-type access
[*DeviceB-100GE1/0/2] port default vlan 2
[*DeviceB-100GE1/0/2] quit
[*DeviceB] interface 100ge 1/0/3
[*DeviceB-100GE1/0/3] portswitch
[*DeviceB-100GE1/0/3] port link-type access
[*DeviceB-100GE1/0/3] port default vlan 3
[*DeviceB-100GE1/0/3] quit
[*DeviceB] interface 100ge 1/0/4
[*DeviceB-100GE1/0/4] portswitch
[*DeviceB-100GE1/0/4] port link-type access
[*DeviceB-100GE1/0/4] port default vlan 3
[*DeviceB-100GE1/0/4] quit
[*DeviceB] commit
```

步骤2 配置DeviceA与DeviceC、DeviceB与DeviceC之间的链路为干道链路

配置DeviceA。

```
[~DeviceA] interface 100ge 1/0/5
[~DeviceA-100GE1/0/5] portswitch
[*DeviceA-100GE1/0/5] port link-type trunk
[*DeviceA-100GE1/0/5] port trunk allow-pass vlan 2 3
[*DeviceA-100GE1/0/5] quit
[*DeviceA] commit
```

配置DeviceB。

```
[~DeviceB] interface 100ge 1/0/5
[~DeviceB-100GE1/0/5] portswitch
[*DeviceB-100GE1/0/5] port link-type trunk
[*DeviceB-100GE1/0/5] port trunk allow-pass vlan 2 3
[*DeviceB-100GE1/0/5] quit
[*DeviceB] commit
```

配置DeviceC。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceC
[*HUAWEI] commit
[~DeviceC] vlan batch 2 3
[*DeviceC] interface 100ge 1/0/1
[*DeviceC-100GE1/0/1] portswitch
```

```
[*DeviceC-100GE1/0/1] port link-type trunk
[*DeviceC-100GE1/0/1] port trunk allow-pass vlan 2 3
[*DeviceC-100GE1/0/1] quit
[*DeviceC] interface 100ge 1/0/2
[*DeviceC-100GE1/0/2] portswitch
[*DeviceC-100GE1/0/2] port link-type trunk
[*DeviceC-100GE1/0/2] port trunk allow-pass vlan 2 3
[*DeviceC-100GE1/0/2] quit
[*DeviceC] commit
```

----结束

检查配置结果

执行命令**display vlan**可以查看VLAN状态，以DeviceA为例：

```
[~DeviceA] display vlan 2
--------------------------------------------------------------------------------
U: Up;         D: Down;         TG: Tagged;         UT: Untagged;
MP: Vlan-mapping;               ST: Vlan-stacking;
#: ProtocolTransparent-vlan;    *: Management-vlan;
MAC-LRN: MAC-address learning;  STAT: Statistic;
BC: Broadcast; MC: Multicast;   UC: Unknown-unicast;
FWD: Forward;  DSD: Discard;
--------------------------------------------------------------------------------

VID          Ports
--------------------------------------------------------------------------------
  2          UT:100GE1/0/1(U)    100GE1/0/2(U)
             TG:100GE1/0/5(U)

VID  Type     Status  Property  MAC-LRN STAT    BC  MC  UC  Description
--------------------------------------------------------------------------------
  2 common   enable  default   enable  disable FWD FWD FWD VLAN 0002
```

执行命令**display port vlan**，查看100GE1/0/5接口上可以通过的VLAN信息，以DeviceA为例：

```
[~DeviceA] display port vlan 100ge 1/0/5
Port                Link Type    PVID  Trunk VLAN List                 Port Description
--------------------------------------------------------------------------------------------------------------------
100GE1/0/5          trunk          1  1-3
```

在DeviceA和DeviceB下，属于相同VLAN2或相同VLAN3内的主机之间能够互相Ping通，并且VLAN2的主机无法Ping通VLAN3内的主机。

配置脚本

- DeviceA

```
#
sysname DeviceA
#
vlan batch 2 to 3
#
interface 100GE1/0/1
 port link-type access
 port default vlan 2
#
interface 100GE1/0/2
 port link-type access
 port default vlan 2
#
interface 100GE1/0/3
 port link-type access
 port default vlan 3
#
```

```
interface 100GE1/0/4
 port link-type access
 port default vlan 3
#
interface 100GE1/0/5
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
#
return
```

- DeviceB

```
#
sysname DeviceB
#
vlan batch 2 to 3
#
interface 100GE1/0/1
 port link-type access
 port default vlan 2
#
interface 100GE1/0/2
 port link-type access
 port default vlan 2
#
interface 100GE1/0/3
 port link-type access
 port default vlan 3
#
interface 100GE1/0/4
 port link-type access
 port default vlan 3
#
interface 100GE1/0/5
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
#
return
```

- DeviceC

```
#
sysname DeviceC
#
vlan batch 2 to 3
#
interface 100GE1/0/1
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
#
interface 100GE1/0/2
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
#
return
```

2.3.2 STP/RSTP/MSTP

2.3.2.1 举例：配置 MSTP+VRRP 组合组网

组网需求

如图2-9所示，主机通过DeviceC接入网络，DeviceC通过双上行连接DeviceA和DeviceB来接入Internet。由于接入备份的需要，用户部署了冗余链路。冗余备份链路的存在导致出现环网，可能会引起广播风暴和MAC地址表项被破坏。用户希望在存在

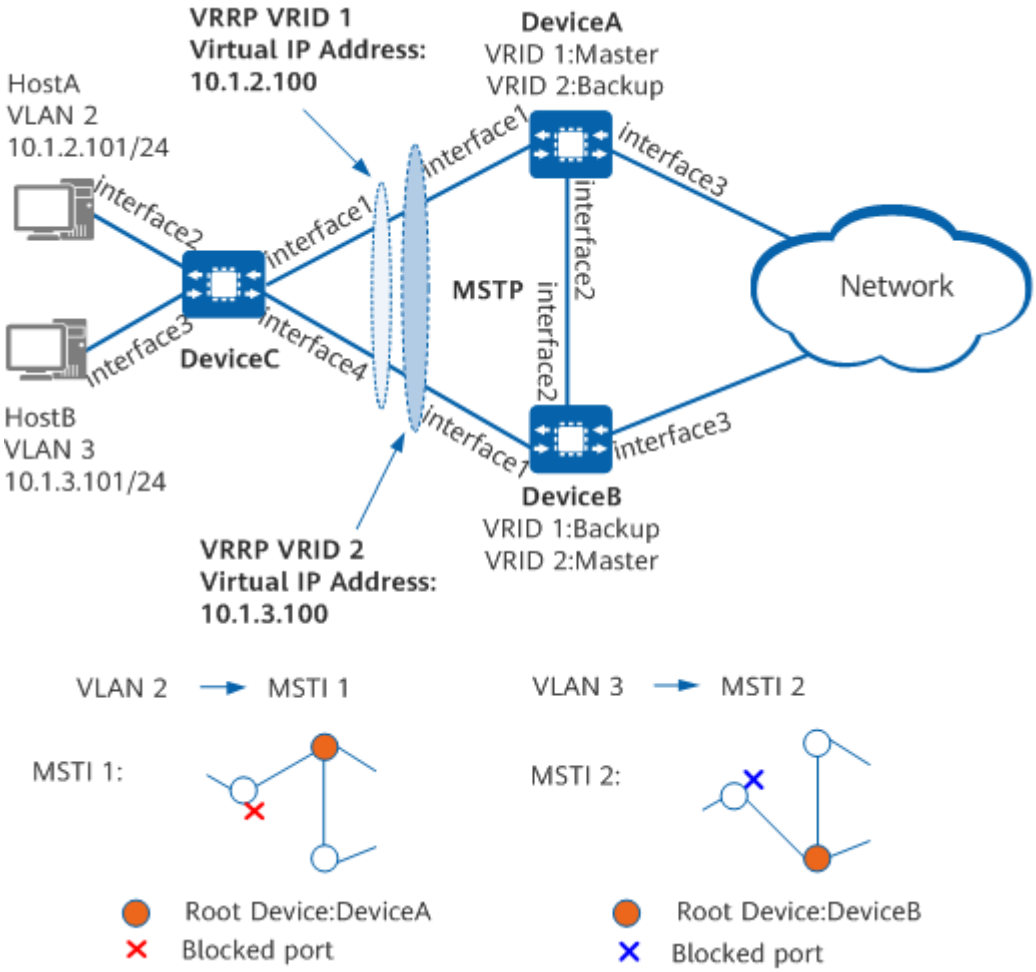
冗余备份链路的同时消除网络中的环路，在一条上行链路断开的时候，流量能切换到另外一条上行链路转发，还能合理利用网络带宽。

此时可以在网络中部署MSTP解决环路问题。MSTP可阻塞二层网络中的冗余链路，将网络修剪成树状，达到消除环路的目的。同时在DeviceA和DeviceB上配置VRRP，HostA以DeviceA为默认网关接入Internet，DeviceB作为备份网关；HostB以DeviceB为默认网关接入Internet，DeviceA作为备份网关，以实现可靠性及流量的负载分担。

图 2-9 配置 MSTP+VRRP 组合组网图

说明

本例中interface1、interface2、interface3和interface4分别代表100GE1/0/1、100GE1/0/2、100GE1/0/3和100GE1/0/4。



设备	接口	对应的VLANIF	IP地址
DeviceA	interface1和interface2	VLANIF2	10.1.2.102/24
	interface1和interface2	VLANIF3	10.1.3.102/24
	interface3	VLANIF4	10.1.4.102/24

设备	接口	对应的VLANIF	IP地址
DeviceB	interface1和interface2	VLANIF2	10.1.2.103/24
	interface1和interface2	VLANIF3	10.1.3.103/24
	interface3	VLANIF5	10.1.5.103/24

配置思路

采用以下思路配置：

1. 在处于环形网络中的设备上配置MSTP基本功能，包括：
   a. 配置MST域并创建多实例，配置VLAN2映射到MSTI1，VLAN3映射到MSTI2，实现流量的负载分担。
   b. 在MST域内，配置各实例的根桥与备份根桥。
   c. 配置各实例中某端口的路径开销值，实现将该端口阻塞。
   d. 使能MSTP，实现破除环路，包括：
      - 设备全局使能MSTP。
      - 除与终端设备相连的端口外，其他端口使能MSTP。

      说明

      与终端相连的端口不用参与MSTP计算，建议将其设置为边缘端口。
2. 配置保护功能，实现对设备或链路的保护。例如：在各实例的根桥设备指定端口配置根保护功能。
3. 配置设备的二层转发功能。
4. 配置各设备端口IP地址及路由协议，使各设备间网络层连通。
5. 在DeviceA和DeviceB上创建VRRP备份组1和VRRP备份组2，在备份组1中，配置DeviceA为Master设备，DeviceB为Backup设备；在备份组2中，配置DeviceB为Master设备，DeviceA为Backup设备，实现流量的负载均衡。

操作步骤

步骤1 配置MSTP基本功能

1. 配置DeviceA、DeviceB、DeviceC到域名为RG1的域内，创建实例MSTI1和实例MSTI2

   # 配置DeviceA的MST域。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] stp region-configuration
[*DeviceA-mst-region] region-name RG1
[*DeviceA-mst-region] instance 1 vlan 2
[*DeviceA-mst-region] instance 2 vlan 3
[*DeviceA-mst-region] quit
[*DeviceA] commit
```

配置DeviceB的MST域。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*HUAWEI] commit
[~DeviceB] stp region-configuration
[*DeviceB-mst-region] region-name RG1
[*DeviceB-mst-region] instance 1 vlan 2
[*DeviceB-mst-region] instance 2 vlan 3
[*DeviceB-mst-region] quit
[*DeviceB] commit
```

配置DeviceC的MST域。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceC
[*HUAWEI] commit
[~DeviceC] stp region-configuration
[*DeviceC-mst-region] region-name RG1
[*DeviceC-mst-region] instance 1 vlan 2
[*DeviceC-mst-region] instance 2 vlan 3
[*DeviceC-mst-region] quit
[*DeviceC] commit
```

2. 在域RG1内，配置MSTI1与MSTI2的根桥与备份根桥
   - 配置MSTI1的根桥与备份根桥

     # 配置DeviceA为MSTI1的根桥。

     ```
     [~DeviceA] stp instance 1 root primary
     [*DeviceA] commit
     ```

     # 配置DeviceB为MSTI1的备份根桥。

     ```
     [~DeviceB] stp instance 1 root secondary
     [*DeviceB] commit
     ```

   - 配置MSTI2的根桥与备份根桥

     # 配置DeviceB为MSTI2的根桥。

     ```
     [~DeviceB] stp instance 2 root primary
     [*DeviceB] commit
     ```

     # 配置DeviceA为MSTI2的备份根桥。

     ```
     [~DeviceA] stp instance 2 root secondary
     [*DeviceA] commit
     ```

3. 配置实例MSTI1和MSTI2中将要被阻塞端口的路径开销值大于缺省值

   说明

   - 端口路径开销值取值范围由路径开销计算方法决定，这里选择使用华为计算方法为例，配置实例MSTI1和MSTI2中将被阻塞端口的路径开销值为20000。
   - 同一网络内所有设备的端口路径开销应使用相同的计算方法。

   # 配置DeviceA的端口路径开销计算方法为华为计算方法。

   ```
   [~DeviceA] stp pathcost-standard legacy
   [*DeviceA] commit
   ```

   # 配置DeviceB的端口路径开销计算方法为华为计算方法。

   ```
   [~DeviceB] stp pathcost-standard legacy
   [*DeviceB] commit
   ```

   # 配置DeviceC的端口路径开销计算方法为华为计算方法，将端口100GE1/0/1在实例MSTI2中的路径开销值配置为20000，将端口100GE1/0/4在实例MSTI1中的路径开销值配置为20000。

   ```
   [~DeviceC] stp pathcost-standard legacy
   [*DeviceC] commit
   [~DeviceC] interface 100ge1/0/1
   [*DeviceC-100GE1/0/1] stp instance 2 cost 20000
   [*DeviceC-100GE1/0/1] quit
   ```

```
[*DeviceC] interface 100ge1/0/4
[*DeviceC-100GE1/0/4] stp instance 1 cost 20000
[*DeviceC-100GE1/0/4] quit
[*DeviceC] commit
```

4. 使能MSTP，实现破除环路
   - 设备全局使能MSTP

     # 在DeviceA上启动MSTP。

     ```
     [~DeviceA] stp enable
     [*DeviceA] commit
     ```

     # 在DeviceB上启动MSTP。

     ```
     [~DeviceB] stp enable
     [*DeviceB] commit
     ```

     # 在DeviceC上启动MSTP。

     ```
     [~DeviceC] stp enable
     [*DeviceC] commit
     ```

   - 将与Host相连的端口设置为边缘端口

     # 配置DeviceC端口的100GE1/0/2和100GE1/0/3为边缘端口。

     ```
     [~DeviceC] interface 100ge1/0/2
     [*DeviceC-100GE1/0/2] stp edged-port enable
     [*DeviceC-100GE1/0/2] quit
     [*DeviceC] interface 100ge1/0/3
     [*DeviceC-100GE1/0/3] stp edged-port enable
     [*DeviceC-100GE1/0/3] quit
     [*DeviceC] commit
     ```

     （可选）配置DeviceC的BPDU保护功能。

     ```
     [~DeviceC] stp bpdu-protection
     [*DeviceC] commit
     ```

   - 将与Network相连的端口设置为边缘端口

     # 配置DeviceA端口100GE1/0/3为边缘端口。

     ```
     [~DeviceA] interface 100ge1/0/3
     [~DeviceA-100GE1/0/3] stp edged-port enable
     [*DeviceA-100GE1/0/3] quit
     [*DeviceA] commit
     ```

     （可选）配置DeviceA的BPDU保护功能。

     ```
     [~DeviceA] stp bpdu-protection
     [*DeviceA] commit
     ```

     # 配置DeviceB端口100GE1/0/3为边缘端口。

     ```
     [~DeviceB] interface 100ge1/0/3
     [~DeviceB-100GE1/0/3] stp edged-port enable
     [*DeviceB-100GE1/0/3] quit
     [*DeviceB] commit
     ```

     （可选）配置DeviceB的BPDU保护功能。

     ```
     [~DeviceB] stp bpdu-protection
     [*DeviceB] commit
     ```

     说明

     如果与边缘端口相连的是使能了STP功能的网络设备，配置BPDU保护功能后，如果边缘端口收到BPDU报文，边缘端口将会被shutdown，边缘端口属性不变。

步骤2 配置保护功能，如在各实例的根桥设备的指定端口配置根保护功能

在DeviceA端口100GE1/0/1上启动根保护。

```
[~DeviceA] interface 100ge1/0/1
[*DeviceA-100GE1/0/1] stp root-protection
```



```
[*DeviceA-100GE1/0/1] quit
[*DeviceA] commit
```

在DeviceB端口100GE1/0/1上启动根保护。

```
[~DeviceB] interface 100ge1/0/1
[*DeviceB-100GE1/0/1] stp root-protection
[*DeviceB-100GE1/0/1] quit
[*DeviceB] commit
```

步骤3 配置处于环网中的设备的二层转发功能

- 在设备DeviceA、DeviceB、DeviceC上创建VLAN2～3

  # 在DeviceA上创建VLAN2～3。

  ```
  [~DeviceA] vlan batch 2 to 3
  [*DeviceA] commit
  ```

  # 在DeviceB上创建VLAN2～3。

  ```
  [~DeviceB] vlan batch 2 to 3
  [*DeviceB] commit
  ```

  # 在DeviceC上创建VLAN2～3。

  ```
  [~DeviceC] vlan batch 2 to 3
  [*DeviceC] commit
  ```

- 将设备上相应的端口加入VLAN

  # 将DeviceA端口100GE1/0/1加入VLAN。

  ```
  [~DeviceA] interface 100ge1/0/1
  [~DeviceA-100GE1/0/1] port link-type trunk
  [*DeviceA-100GE1/0/1] port trunk allow-pass vlan 2 to 3
  [*DeviceA-100GE1/0/1] quit
  [*DeviceA] commit
  ```

  # 将DeviceA端口100GE1/0/2加入VLAN。

  ```
  [~DeviceA] interface 100ge1/0/2
  [*DeviceA-100GE1/0/2] port link-type trunk
  [*DeviceA-100GE1/0/2] port trunk allow-pass vlan 2 to 3
  [*DeviceA-100GE1/0/2] quit
  [*DeviceA] commit
  ```

  # 将DeviceB端口100GE1/0/1加入VLAN。

  ```
  [~DeviceB] interface 100ge1/0/1
  [~DeviceB-100GE1/0/1] port link-type trunk
  [*DeviceB-100GE1/0/1] port trunk allow-pass vlan 2 to 3
  [*DeviceB-100GE1/0/1] quit
  [*DeviceB] commit
  ```

  # 将DeviceB端口100GE1/0/2加入VLAN。

  ```
  [~DeviceB] interface 100ge1/0/2
  [*DeviceB-100GE1/0/2] port link-type trunk
  [*DeviceB-100GE1/0/2] port trunk allow-pass vlan 2 to 3
  [*DeviceB-100GE1/0/2] quit
  [*DeviceB] commit
  ```

  # 将DeviceC端口100GE1/0/1加入VLAN。

  ```
  [~DeviceC] interface 100ge1/0/1
  [~DeviceC-100GE1/0/1] port link-type trunk
  [*DeviceC-100GE1/0/1] port trunk allow-pass vlan 2 to 3
  [*DeviceC-100GE1/0/1] quit
  [*DeviceC] commit
  ```

  # 将DeviceC端口100GE1/0/2加入VLAN。

  ```
  [~DeviceC] interface 100ge1/0/2
  [~DeviceC-100GE1/0/2] port link-type trunk
  [*DeviceC-100GE1/0/2] port trunk allow-pass vlan 2 to 3
  [*DeviceC-100GE1/0/2] quit
  [*DeviceC] commit
  ```

将DeviceC端口100GE1/0/3加入VLAN。

```
[~DeviceC] interface 100ge1/0/3
[~DeviceC-100GE1/0/3] port link-type trunk
[*DeviceC-100GE1/0/3] port trunk allow-pass vlan 2 to 3
[*DeviceC-100GE1/0/3] quit
[*DeviceC] commit
```

将DeviceC端口100GE1/0/4加入VLAN。

```
[~DeviceC] interface 100ge1/0/4
[~DeviceC-100GE1/0/4] port link-type trunk
[*DeviceC-100GE1/0/4] port trunk allow-pass vlan 2 to 3
[*DeviceC-100GE1/0/4] quit
[*DeviceC] commit
```

步骤4 验证配置结果

经过以上配置，在网络计算稳定后，执行以下操作，验证配置结果。

说明

本配置举例以实例1和实例2为例，因此不用关注实例0中端口的状态。

在DeviceA上执行**display stp brief**命令，查看端口状态和端口的保护类型，结果如下：

```
[~DeviceA] display stp brief
 MSTID     Port                Role  STP State       Protection
   0       100GE1/0/1          DESI  FORWARDING      ROOT
   0       100GE1/0/2          DESI  FORWARDING      NONE
   1       100GE1/0/1          DESI  FORWARDING      ROOT
   1       100GE1/0/2          DESI  FORWARDING      NONE
   2       100GE1/0/1          DESI  FORWARDING      ROOT
   2       100GE1/0/2          ROOT  FORWARDING      NONE
```

在MSTI1中，由于DeviceA是根桥，DeviceA的端口100GE1/0/1和100GE1/0/2成为指定端口。在MSTI2中，DeviceA的端口100GE1/0/1成为指定端口，端口100GE1/0/2成为根端口。

在DeviceB上执行**display stp brief**命令，结果如下：

```
[~DeviceB] display stp brief
 MSTID     Port                Role  STP State       Protection
   0       100GE1/0/1          DESI  FORWARDING      ROOT
   0       100GE1/0/2          ROOT  FORWARDING      NONE
   1       100GE1/0/1          DESI  FORWARDING      ROOT
   1       100GE1/0/2          ROOT  FORWARDING      NONE
   2       100GE1/0/1          DESI  FORWARDING      ROOT
   2       100GE1/0/2          DESI  FORWARDING      NONE
```

在MSTI2中，由于DeviceB是根桥，端口100GE1/0/1和100GE1/0/2在MSTI2中成为指定端口。在MSTI1中，DeviceB的端口100GE1/0/1成为指定端口，端口100GE1/0/2成为根端口。

在DeviceC上执行**display stp interface brief**命令，结果如下：

```
[~DeviceC] display stp interface
100ge1/0/1
 brief
 MSTID     Port                Role  STP State       Protection
   0       100GE1/0/1          ROOT  FORWARDING      NONE
   1       100GE1/0/1          ROOT  FORWARDING      NONE
   2       100GE1/0/1          ALTE  DISCARDING      NONE
[~DeviceC] display stp interface 100ge1/0/4 brief
 MSTID     Port                Role  STP State       Protection
   0       100GE1/0/4          ALTE  DISCARDING      NONE
   1       100GE1/0/4          ALTE  DISCARDING      NONE
   2       100GE1/0/4          ROOT  FORWARDING      NONE
```

DeviceC的端口100GE1/0/1在MSTI1中为根端口，在MSTI2中被阻塞。DeviceC的另一个端口100GE1/0/4，在MSTI1中被阻塞，在MSTI2中为根端口。

步骤5 配置设备间的网络互连

配置设备各端口的IP地址，以DeviceA为例。DeviceB的配置与DeviceA类似，详见配置文件。

```
[~DeviceA] vlan batch 4
[*DeviceA] interface 100ge1/0/3
[*DeviceA-100GE1/0/3] port link-type trunk
[*DeviceA-100GE1/0/3] port trunk allow-pass vlan 4
[*DeviceA-100GE1/0/3] quit
[*DeviceA] interface vlanif 2
[*DeviceA-Vlanif2] ip address 10.1.2.102 24
[*DeviceA-Vlanif2] quit
[*DeviceA] interface vlanif 3
[*DeviceA-Vlanif3] ip address 10.1.3.102 24
[*DeviceA-Vlanif3] quit
[*DeviceA] interface vlanif 4
[*DeviceA-Vlanif4] ip address 10.1.4.102 24
[*DeviceA-Vlanif4] quit
[*DeviceA] commit
```

配置DeviceA、DeviceB和Network间采用OSPF协议进行互连。以DeviceA为例，DeviceB的配置与DeviceA类似，详见配置文件。

```
[~DeviceA] ospf 1
[*DeviceA-ospf-1] area 0
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.2.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.3.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 10.1.4.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] quit
[*DeviceA-ospf-1] quit
[*DeviceA] commit
```

步骤6 配置VRRP备份组

在DeviceA和DeviceB上创建VRRP备份组1，配置DeviceA的优先级为120，抢占延时为20秒，作为Master设备。

```
[~DeviceA] interface vlanif 2
[~DeviceA-Vlanif2] vrrp vrid 1 virtual-ip 10.1.2.100
[*DeviceA-Vlanif2] vrrp vrid 1 priority 120
[*DeviceA-Vlanif2] vrrp vrid 1 preempt timer delay 20
[*DeviceA-Vlanif2] quit
[*DeviceA] commit
```

DeviceB的优先级为缺省值，作为Backup设备。

```
[~DeviceB] interface vlanif 2
[~DeviceB-Vlanif2] vrrp vrid 1 virtual-ip 10.1.2.100
[*DeviceB-Vlanif2] quit
[*DeviceB] commit
```

在DeviceA和DeviceB上创建VRRP备份组2，配置DeviceB的优先级为120，抢占延时为20秒，作为Master设备。

```
[~DeviceB] interface vlanif 3
[~DeviceB-Vlanif3] vrrp vrid 2 virtual-ip 10.1.3.100
[*DeviceB-Vlanif3] vrrp vrid 2 priority 120
[*DeviceB-Vlanif3] vrrp vrid 2 preempt timer delay 20
[*DeviceB-Vlanif3] quit
[*DeviceB] commit
```

DeviceA的优先级为缺省值，作为Backup设备。

```
[~DeviceA] interface vlanif 3
[~DeviceA-Vlanif3] vrrp vrid 2 virtual-ip 10.1.3.100
```



```
[*DeviceA-Vlanif3] quit
[*DeviceA] commit
```

配置主机HostA的缺省网关为备份组1的虚拟IP地址10.1.2.100，配置主机HostB的缺省网关为备份组2的虚拟IP地址10.1.3.100。

步骤7 验证配置结果

完成上述配置后，在DeviceA上执行**display vrrp**命令，可以看到DeviceA在备份组1中作为Master设备，在备份组2中作为Backup设备。

```
[~DeviceA] display vrrp
  Vlanif2 | Virtual Router 1
    State : Master
    Virtual IP : 10.1.2.100
    Master IP : 10.1.2.102
    PriorityRun : 120
    PriorityConfig : 120
    MasterPriority : 120
    Preempt : YES   Delay Time : 20 s
    TimerRun : 1 s
    TimerConfig : 1 s
    Auth type : NONE
    Virtual MAC : 00e0-fc12-3456
    Check TTL : YES
    Config type : normal-vrrp
    Backup-forward : disabled
    Create time : 2021-05-11 11:39:18
    Last change time : 2021-05-26 11:38:58

  Vlanif3 | Virtual Router 2
    State : Backup
    Virtual IP : 10.1.3.100
    Master IP : 10.1.3.103
    PriorityRun : 100
    PriorityConfig : 100
    MasterPriority : 120
    Preempt : YES   Delay Time : 0 s
    TimerRun : 1 s
    TimerConfig : 1 s
    Auth type : NONE
    Virtual MAC : 00e0-fc12-3457
    Check TTL : YES
    Config type : normal-vrrp
    Backup-forward : disabled
    Create time : 2021-05-11 11:40:18
    Last change time : 2021-05-26 11:48:58
```

在DeviceB上执行**display vrrp**命令，可以看到DeviceB在备份组1中作为Backup设备，在备份组2中作为Master设备。

```
[~DeviceB] display vrrp
  Vlanif2 | Virtual Router 1
    State : Backup
    Virtual IP : 10.1.2.100
    Master IP : 10.1.2.102
    PriorityRun : 100
    PriorityConfig : 100
    MasterPriority : 120
    Preempt : YES   Delay Time : 0 s
    TimerRun : 1 s
    TimerConfig : 1 s
    Auth type : NONE
    Virtual MAC : 00e0-fc12-3456
    Check TTL : YES
    Config type : normal-vrrp
    Backup-forward : disabled
    Create time : 2021-05-11 11:39:18
    Last change time : 2021-05-26 11:38:58
```

```
Vlanif3 | Virtual Router 2
  State : Master
  Virtual IP : 10.1.3.100
  Master IP : 10.1.3.103
  PriorityRun : 120
  PriorityConfig : 120
  MasterPriority : 120
  Preempt : YES   Delay Time : 20 s
  TimerRun : 1 s
  TimerConfig : 1 s
  Auth type : NONE
  Virtual MAC : 00e0-fc12-3457
  Check TTL : YES
  Config type : normal-vrrp
  Backup-forward : disabled
  Create time : 2021-05-11 11:40:18
  Last change time : 2021-05-26 11:48:58
```

----结束

配置脚本

- DeviceA的配置文件

```
#
sysname DeviceA
#
vlan batch 2 to 4
#
stp instance 1 root primary
stp instance 2 root secondary
stp bpdu-protection
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
#
interface Vlanif2
 ip address 10.1.2.102 255.255.255.0
 vrrp vrid 1 virtual-ip 10.1.2.100
 vrrp vrid 1 priority 120
 vrrp vrid 1 preempt timer delay 20
#
interface Vlanif3
 ip address 10.1.3.102 255.255.255.0
 vrrp vrid 2 virtual-ip 10.1.3.100
#
interface Vlanif4
 ip address 10.1.4.102 255.255.255.0
#
interface 100GE1/0/1
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
 stp root-protection
#
interface 100GE1/0/2
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
#
interface 100GE1/0/3
 port link-type trunk
 port trunk allow-pass vlan 4
 stp edged-port enable
#
ospf 1
 area 0.0.0.0
```



```
 network 10.1.2.0 0.0.0.255
 network 10.1.3.0 0.0.0.255
 network 10.1.4.0 0.0.0.255
#
return
```

- DeviceB的配置文件

```
#
sysname DeviceB
#
vlan batch 2 to 3 5
#
stp instance 1 root secondary
stp instance 2 root primary
stp bpdu-protection
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
#
interface Vlanif2
 ip address 10.1.2.103 255.255.255.0
 vrrp vrid 1 virtual-ip 10.1.2.100
#
interface Vlanif3
 ip address 10.1.3.103 255.255.255.0
 vrrp vrid 2 virtual-ip 10.1.3.100
 vrrp vrid 2 priority 120
 vrrp vrid 2 preempt timer delay 20
#
interface Vlanif5
 ip address 10.1.5.103 255.255.255.0
#
interface 100GE1/0/1
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
 stp root-protection
#
interface 100GE1/0/2
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
#
interface 100GE1/0/3
 port link-type trunk
 port trunk allow-pass vlan 5
 stp edged-port enable
#
ospf 1
 area 0.0.0.0
  network 10.1.2.0 0.0.0.255
  network 10.1.3.0 0.0.0.255
  network 10.1.5.0 0.0.0.255
#
return
```

- DeviceC的配置文件

```
#
sysname DeviceC
#
vlan batch 2 to 3
#
stp bpdu-protection
stp pathcost-standard legacy
#
stp region-configuration
 region-name RG1
 instance 1 vlan 2
 instance 2 vlan 3
```

```
#
interface 100GE1/0/1
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
 stp instance 2 cost 20000
#
interface 100GE1/0/2
 port link-type access
 port default vlan 2
 stp edged-port enable
#
interface 100GE1/0/3
 port link-type access
 port default vlan 3
 stp edged-port enable
#
interface 100GE1/0/4
 port link-type trunk
 port trunk allow-pass vlan 2 to 3
 stp instance 1 cost 20000
#
return
```

2.3.3 ERPS

2.3.3.1 举例：配置 ERPS 多实例

组网需求

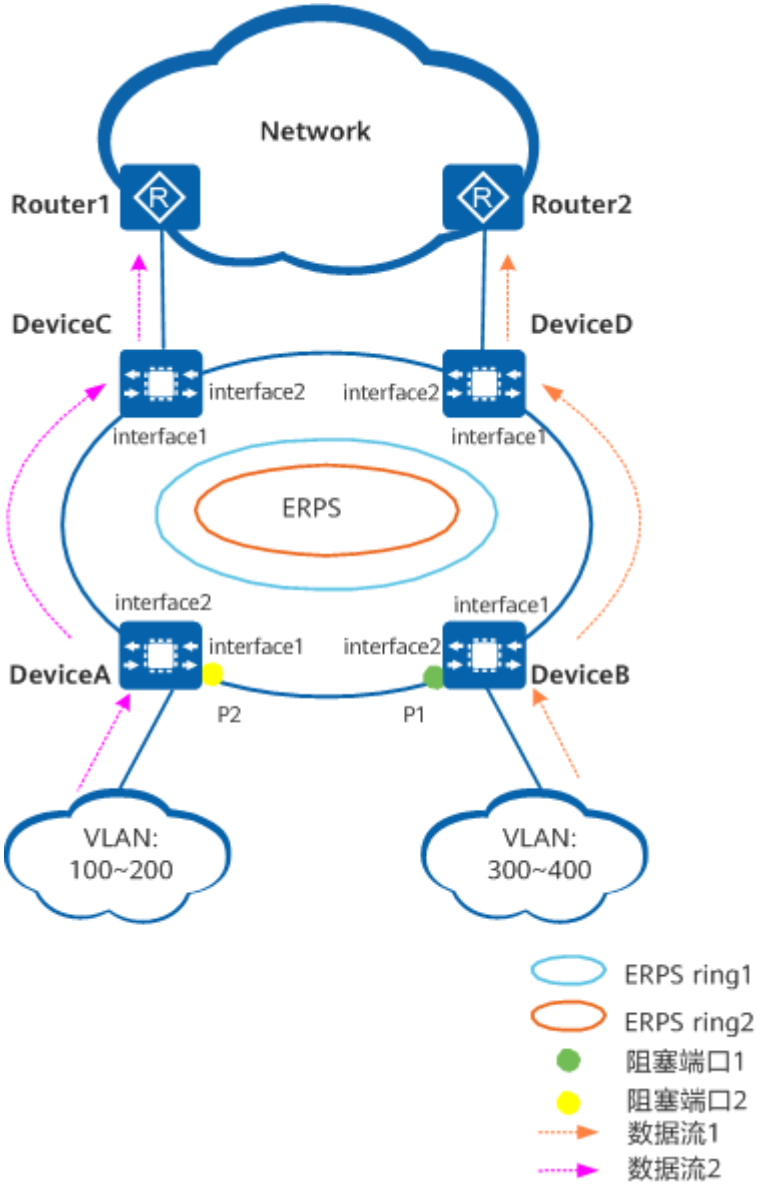
在ERPS组网中，一个物理环上只能配置一个ERPS环，也只能指定一个阻塞点。当ERPS环处于完整状态时，阻塞端口会阻止所有的用户报文通过，这导致所有用户报文在ERPS环上只能通过一条路径传输，阻塞端口另一侧的链路空闲，造成了带宽浪费。

如**图 ERPS单环多实例组网图**所示，在DeviceA～DeviceD上配置两个ERPS实例，ERPS环1和ERPS环2，ERPS环1阻塞DeviceB的P1端口，ERPS环2阻塞DeviceA的P2端口，实现负载分担并提供链路备份。

图 2-10 ERPS 单环多实例组网图

说明

本例中interface1、interface2分别代表100GE1/0/1、100GE1/0/2。



配置思路

采用如下的思路配置ERPS单环多实例：

1. 配置加入ERPS环的所有端口类型为Trunk型。
2. 创建ERPS环，并配置控制VLAN和保护实例。
3. 将二层端口加入ERPS环并配置端口角色。
4. 配置ERPS环的Guard Timer和WTR Timer定时器。
5. 配置DeviceA～DeviceD二层转发功能。

操作步骤

步骤1 配置加入ERPS环的所有端口类型为Trunk型。

配置DeviceA。DeviceB、DeviceC、DeviceD的配置与DeviceA类似，详见配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[~DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] portswitch
[*DeviceA-100GE1/0/1] port link-type trunk
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] portswitch
[*DeviceA-100GE1/0/2] port link-type trunk
[*DeviceA-100GE1/0/2] quit
[*DeviceA-100GE1/0/2] commit
```

步骤2 创建ERPS环1、ERPS环2并配置两个ERPS环的保护实例，配置ERPS环1的控制VLAN ID为10，ERPS环2的控制VLAN ID为20，ERPS环1传递VLAN100～VLAN200的数据报文，ERPS环2传递VLAN300～VLAN400的数据报文。

配置DeviceA。DeviceB、DeviceC、DeviceD的配置与DeviceA类似，详见配置脚本。

```
[~DeviceA] erps ring 1
[*DeviceA-erps-ring1] control-vlan 10
[*DeviceA-erps-ring1] protected-instance 1
[*DeviceA-erps-ring1] quit
[*DeviceA] stp region-configuration
[*DeviceA-mst-region] instance 1 vlan 10 100 to 200
[*DeviceA-mst-region] quit
[*DeviceA] erps ring 2
[*DeviceA-erps-ring2] control-vlan 20
[*DeviceA-erps-ring2] protected-instance 2
[*DeviceA-erps-ring2] quit
[*DeviceA] stp region-configuration
[*DeviceA-mst-region] instance 2 vlan 20 300 to 400
[*DeviceA-mst-region] quit
[*DeviceA-mst-region] commit
```

步骤3 将二层端口加入ERPS环并配置端口角色，分别将DeviceA的端口100GE1/0/1和DeviceB的端口100GE1/0/2配置为RPL owner。

配置DeviceA。

```
[~DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] stp disable
[*DeviceA-100GE1/0/1] erps ring 1
[*DeviceA-100GE1/0/1] erps ring 2 rpl owner
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] stp disable
[*DeviceA-100GE1/0/2] erps ring 1
[*DeviceA-100GE1/0/2] erps ring 2
[*DeviceA-100GE1/0/2] quit
[*DeviceA-100GE1/0/2] commit
```

配置DeviceB。

```
[~DeviceB] interface 100ge 1/0/1
[*DeviceB-100GE1/0/1] stp disable
[*DeviceB-100GE1/0/1] erps ring 1
[*DeviceB-100GE1/0/1] erps ring 2
[*DeviceB-100GE1/0/1] quit
[*DeviceB] interface 100ge 1/0/2
[*DeviceB-100GE1/0/2] stp disable
[*DeviceB-100GE1/0/2] erps ring 1 rpl owner
[*DeviceB-100GE1/0/2] erps ring 2
[*DeviceB-100GE1/0/2] quit
[*DeviceB-100GE1/0/2] commit
```

配置DeviceC。DeviceD的配置与DeviceC类似，详见配置脚本。

```
[~DeviceC] interface 100ge 1/0/1
[*DeviceC-100GE1/0/1] stp disable
[*DeviceC-100GE1/0/1] erps ring 1
[*DeviceC-100GE1/0/1] erps ring 2
[*DeviceC-100GE1/0/1] quit
[*DeviceC] interface 100ge 1/0/2
[*DeviceC-100GE1/0/2] stp disable
[*DeviceC-100GE1/0/2] erps ring 1
[*DeviceC-100GE1/0/2] erps ring 2
[*DeviceC-100GE1/0/2] quit
[*DeviceC-100GE1/0/2] commit
```

步骤4 配置ERPS环的Guard Timer和WTR Timer定时器。

配置DeviceA。DeviceB、DeviceC、DeviceD的配置与DeviceA类似，详见配置脚本。

```
[~DeviceA] erps ring 1
[~DeviceA-erps-ring1] wtr-timer 6
[*DeviceA-erps-ring1] guard-timer 100
[*DeviceA-erps-ring1] quit
[*DeviceA] erps ring 2
[*DeviceA-erps-ring2] wtr-timer 6
[*DeviceA-erps-ring2] guard-timer 100
[*DeviceA-erps-ring2] quit
[*DeviceA-erps-ring2] commit
```

步骤5 配置DeviceA～DeviceD二层转发功能。

配置DeviceA。DeviceB、DeviceC、DeviceD的配置与DeviceA类似，详见配置脚本。

```
[~DeviceA] vlan batch 100 to 200 300 to 400
[*DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] undo port trunk allow-pass vlan 1
[*DeviceA-100GE1/0/1] port trunk allow-pass vlan 100 to 200 300 to 400
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] undo port trunk allow-pass vlan 1
[*DeviceA-100GE1/0/2] port trunk allow-pass vlan 100 to 200 300 to 400
[*DeviceA-100GE1/0/2] quit
[*DeviceA-100GE1/0/2] commit
```

步骤6 验证配置结果

在网络稳定后，在设备上执行**display erps**，查看设备加入的ERPS环的端口和环的概要信息。以DeviceB为例。

```
[~DeviceB] display erps
D : Discarding
F : Forwarding
R : RPL Owner
N  : RPL Neighbour
FS : Forced Device
MS : Manual Device
Total number of rings configured = 2
Ring  Control  WTR Timer  Guard Timer  Port 1            Port 2
ID    VLAN     (min)      (csec)
--------------------------------------------------------------------------------
  1      10         6         100  (F)100GE1/0/1         (D,R)100GE1/0/2
  2      20         6         100  (F)100GE1/0/1         (F)100GE1/0/2
--------------------------------------------------------------------------------
```

在设备上执行**display erps verbose**，查看设备加入的ERPS环的端口和环的详细信息。以DeviceB为例。

```
[~DeviceB] display erps verbose
Ring ID                          : 1
Description                      : Ring 1
Control Vlan                     : 10
Protected Instance               : 1
Service Vlan                     : 100 to 200
WTR Timer Setting (min)          : 6      Running (s)          : 0
Guard Timer Setting (csec)       : 100    Running (csec)       : 0
Holdoff Timer Setting (deciseconds) : 0      Running (deciseconds) : 0
WTB Timer Running (csec)         : 0
Ring State                       : Idle
RAPS_MEL                         : 7
Revertive Mode                   : Revertive
R-APS Channel Mode               : -
Version                          : 1
Sub-ring                         : No
Forced Device Port               : -
Manual Device Port               : -
TC-Notify                        : -
Time since last topology change     : 0 days 0h:35m:5s
--------------------------------------------------------------------------------
Port              Port Role     Port Status     Signal Status
--------------------------------------------------------------------------------
100GE1/0/1        Common        Forwarding      Non-failed
100GE1/0/2        RPL Owner     Discarding      Non-failed

Ring ID                          : 2
Description                      : Ring 2
Control Vlan                     : 20
Protected Instance               : 2
Service Vlan                     : 300 to 400
WTR Timer Setting (min)          : 6      Running (s)          : 0
Guard Timer Setting (csec)       : 100    Running (csec)       : 0
Holdoff Timer Setting (deciseconds) : 0      Running (deciseconds) : 0
WTB Timer Running (csec)         : 0
Ring State                       : Idle
RAPS_MEL                         : 7
Revertive Mode                   : Revertive
R-APS Channel Mode               : -
Version                          : 1
Sub-ring                         : No
Forced Device Port               : -
Manual Device Port               : -
TC-Notify                        : -
Time since last topology change     : 0 days 0h:35m:30s
--------------------------------------------------------------------------------
Port              Port Role     Port Status     Signal Status
--------------------------------------------------------------------------------
100GE1/0/1        Common        Forwarding      Non-failed
100GE1/0/2        Common        Forwarding      Non-failed
```

----结束

配置脚本

- DeviceA

```
#
sysname DeviceA
#
vlan batch 10 20 100 to 200 300 to 400
#
stp region-configuration
 instance 1 vlan 10 100 to 200
 instance 2 vlan 20 300 to 400
#
erps ring 1
 control-vlan 10
 protected-instance 1
```

```
 wtr-timer 6
 guard-timer 100
erps ring 2
 control-vlan 20
 protected-instance 2
 wtr-timer 6
 guard-timer 100
#
interface 100ge 1/0/1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10 20 100 to 200 300 to 400
 stp disable
 erps ring 1
 erps ring 2 rpl owner
#
interface 100ge 1/0/2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10 20 100 to 200 300 to 400
 stp disable
 erps ring 1
 erps ring 2
#
return
```

- DeviceB

```
#
sysname DeviceB
#
vlan batch 10 20 100 to 200 300 to 400
#
stp region-configuration
 instance 1 vlan 10 100 to 200
 instance 2 vlan 20 300 to 400
#
erps ring 1
 control-vlan 10
 protected-instance 1
 wtr-timer 6
 guard-timer 100
erps ring 2
 control-vlan 20
 protected-instance 2
 wtr-timer 6
 guard-timer 100
#
interface 100ge 1/0/1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10 20 100 to 200 300 to 400
 stp disable
 erps ring 1
 erps ring 2
#
interface 100ge 1/0/2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10 20 100 to 200 300 to 400
 stp disable
 erps ring 1 rpl owner
 erps ring 2
#
return
```

- DeviceC

```
#
sysname DeviceC
#
vlan batch 10 20 100 to 200 300 to 400
```

```
#
stp region-configuration
 instance 1 vlan 10 100 to 200
 instance 2 vlan 20 300 to 400
#
erps ring 1
 control-vlan 10
 protected-instance 1
 wtr-timer 6
 guard-timer 100
erps ring 2
 control-vlan 20
 protected-instance 2
 wtr-timer 6
 guard-timer 100
#
interface 100ge 1/0/1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10 20 100 to 200 300 to 400
 stp disable
 erps ring 1
 erps ring 2
#
interface 100ge 1/0/2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10 20 100 to 200 300 to 400
 stp disable
 erps ring 1
 erps ring 2
#
return
```

- DeviceD

```
#
sysname DeviceD
#
vlan batch 10 20 100 to 200 300 to 400
#
stp region-configuration
 instance 1 vlan 10 100 to 200
 instance 2 vlan 20 300 to 400
#
erps ring 1
 control-vlan 10
 protected-instance 1
 wtr-timer 6
 guard-timer 100
erps ring 2
 control-vlan 20
 protected-instance 2
 wtr-timer 6
 guard-timer 100
#
interface 100ge 1/0/1
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10 20 100 to 200 300 to 400
 stp disable
 erps ring 1
 erps ring 2
#
interface 100ge 1/0/2
 port link-type trunk
 undo port trunk allow-pass vlan 1
 port trunk allow-pass vlan 10 20 100 to 200 300 to 400
 stp disable
 erps ring 1
 erps ring 2
```



```
#
return
```

2.4 IP 地址与服务

2.4.1 ARP 安全

2.4.1.1 举例：配置 ARP 安全功能

组网需求

如图2-11所示，设备通过接口100GE1/0/3连接一台服务器，通过接口100GE1/0/1和接口100GE1/0/2连接VLAN10和VLAN20下的四个用户主机。网络中存在的ARP攻击包括：

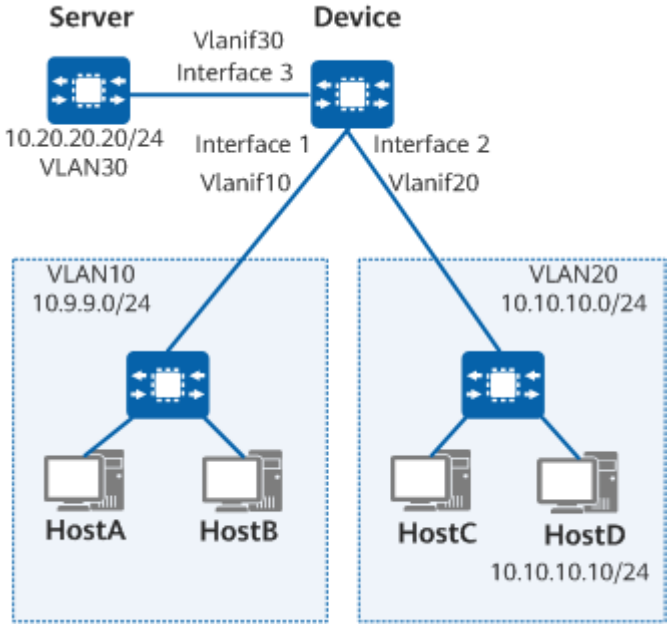
- 服务器可能会发送一些目的IP地址不可达的报文，而且这种报文相对其他普通用户主机的报文要多。
- 用户HostA遭到网络攻击者攻击后，会在网络中发送大量的伪造ARP攻击报文，这种报文中的源IP地址会不断变化。
- 用户HostC遭到网络攻击者攻击后，会构造大量的源IP地址固定的ARP报文对网络进行攻击。
- 用户HostD可能会发送一些目的IP地址不可达的报文。

要求在Device配置ARP的安全功能，预防上述攻击。

图 2-11 配置 ARP 安全功能组网图

说明

本例中interface1代表100GE1/0/1，interface2代表100GE1/0/2，interface3代表100GE1/0/3。



配置思路

采用如下的思路在Device上配置ARP安全功能：

1. 配置全局的ARP表项严格学习的功能，限制设备只学习自己发送的ARP请求报文的应答报文。
2. 基于接口配置ARP表项限制功能，限制接口学习ARP表项的最大数目，防止ARP表项溢出。
3. 配置ARP表项固化功能，防止网络攻击者通过发送伪造的ARP报文篡改ARP表项。
4. 配置ARP报文限速功能，限制设备1秒内处理ARP报文的数量，节省系统开销，保证用户业务不受影响。
5. 配置ARP Miss消息限速功能，限制设备1秒内处理ARP Miss消息的数量，节省系统开销，保证用户业务不受影响。同时需要保证设备可以正常处理服务器发出的大量此类报文，避免因丢弃服务器发出的大量此类报文而造成网络无法正常通信。

数据准备

为完成此配置例，需准备如下的数据：

- 基于接口配置ARP表项限制的阈值为20。
- 配置ARP表项固化的固化模式为**fixed-mac**模式。
- 配置ARP报文限速的阈值为15。
- 配置ARP Miss消息限速的阈值：对用户主机HostD的限速阈值为30，对其他用户主机的限制阈值为20，对服务器的限速阈值为50。

操作步骤

步骤1 配置相关接口的IP地址和路由协议，具体的配置过程参见**配置脚本**。

步骤2 配置ARP表项严格学习功能。

```
<HUAWEI> system-view
[~HUAWEI] sysname Device
[*Device] arp learning strict
[*Device] commit
```

步骤3 基于接口配置ARP表项限制功能。

配置接口100GE1/0/1的ARP表项限制阈值为20。

```
[~Device] interface 100ge1/0/1
[~Device-100GE1/0/1] portswitch
[*Device-100GE1/0/1] arp limit vlan 10 20
[*Device-100GE1/0/1] quit
[*Device] commit
```

#参考接口100GE1/0/1的配置，配置接口100GE1/0/2和接口100GE1/0/3的ARP表项限制阈值。

步骤4 配置ARP表项固化功能。

```
[~Device] arp anti-attack entry-check fixed-mac enable
[*Device] commit
```

步骤5 配置ARP报文限速功能。

```
[~Device] arp anti-attack rate-limit source-ip maximum 15
[*Device] commit
```

步骤6 配置ARP Miss消息限速功能。

```
[~Device] arp miss anti-attack rate-limit source-ip 10.10.10.10 maximum 30
[*Device] arp miss anti-attack rate-limit source-ip maximum 20
[*Device] arp miss anti-attack rate-limit source-ip 10.20.20.20 maximum 50
[*Device] commit
```

----结束

检查配置结果

执行命令**display arp learning strict**，查看ARP表项严格学习的情况。

```
<HUAWEI> display arp learning strict
 The global arp learning strict state:enable
------------------------------------------------------------------
 Interface                       LearningStrictState
------------------------------------------------------------------
------------------------------------------------------------------
 Total:0     Force-enable:0     Force-disable:0
```

执行命令**display arp limit interface**，查看100GE 1/0/1接口上配置的ARP表项限制的阈值。

```
<HUAWEI> display arp limit interface 100ge1/0/1
Interface               VLAN     Limit       Learnt
--------------------------------------------------------------------------------
 100GE1/0/1       10         20          0
--------------------------------------------------------------------------------
 Total:1
```

执行命令**display arp miss anti-attack rate-limit**，查看配置的ARP Miss消息限速的限速值。

```
<HUAWEI> display arp miss anti-attack rate-limit
Global ARP miss rate-limit ： 500 (0 means no limit)

VLAN ID          Suppress Rate(pps)(0 means no limit)
--------------------------------------------------------------------------------
All               0
--------------------------------------------------------------------------------
Total: 0, spec of rate-limit configuration for VLAN is 1024.

Source IP        Suppress Rate(pps)(0 means no limit)
--------------------------------------------------------------------------------
10.10.10.10/32          30
10.20.20.20/32          50
Other                  20
--------------------------------------------------------------------------------
Total: 2, spec of rate-limit configuration for Source IP is 1024.
```

执行命令**display arp packet statistics**，查看ARP报文的统计信息。

```
<HUAWEI> display arp packet statistics
ARP Packets Received
  Total:                    154333
  Learnt Count:                  8
  Discard For Entry Limit:       5
  Discard For Speed Limit:       0
  Discard For Proxy Suppress:    0
  Discard For Other:        151597
ARP Packets Sent
  Total:                         0
  Request:                       0
  Reply:                         0
  Gratuitous ARP:                0
ARP-Miss Message Received:
  Total:                         0
```



```
Discard For Speed Limit:        0
Discard For Other:              3
```

配置脚本

Device

```
#
sysname Device
#
vlan batch 10 20 30
#
arp learning strict
arp anti-attack entry-check fixed-mac enable
arp anti-attack rate-limit source-ip maximum 15
arp miss anti-attack rate-limit source-ip 10.10.10.10 maximum 30
arp miss anti-attack rate-limit source-ip maximum 20
arp miss anti-attack rate-limit source-ip 10.20.20.20 maximum 50
#
interface Vlanif10
 ip address 10.9.9.1 255.255.255.0
#
interface Vlanif20
 ip address 10.10.10.1 255.255.255.0
#
interface Vlanif30
 ip address 10.20.20.1 255.255.255.0
#
interface 100GE1/0/1
 arp limit vlan 10 20
#
interface 100GE1/0/2
 arp limit vlan 20 20
#
interface 100GE1/0/3
 arp limit vlan 30 20
#
return
```

2.4.2 DHCPv4

2.4.2.1 举例：配置 DHCPv4 中继

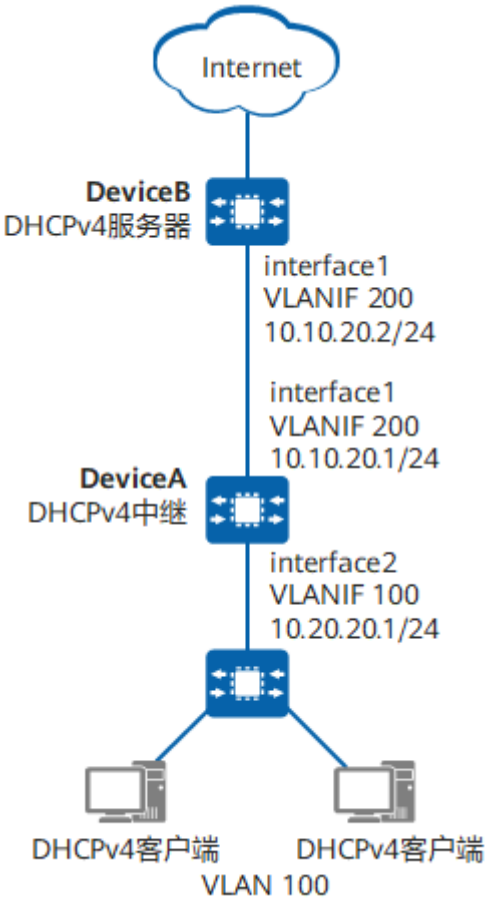
背景信息

如图2-12所示，DHCPv4服务器与DHCPv4客户端不在同一网段，用户希望通过配置DHCPv4中继使DHCPv4客户端动态获取IPv4地址。

图 2-12 配置 DHCPv4 中继组网图

说明

本例中interface1和interface2分别代表100GE1/0/1和100GE1/0/2。



操作步骤

步骤1 配置设备间的网络互连

配置DeviceA。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] vlan batch 100 200
[*DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] portswitch
[*DeviceA-100GE1/0/1] port link-type hybrid
[*DeviceA-100GE1/0/1] port hybrid pvid vlan 200
[*DeviceA-100GE1/0/1] port hybrid untagged vlan 200
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] portswitch
[*DeviceA-100GE1/0/2] port link-type hybrid
[*DeviceA-100GE1/0/2] port hybrid pvid vlan 100
[*DeviceA-100GE1/0/2] port hybrid untagged vlan 100
[*DeviceA-100GE1/0/2] quit
[*DeviceA] interface vlanif 200
[*DeviceA-Vlanif200] ip address 10.10.20.1 24
[*DeviceA-Vlanif200] quit
[*DeviceA] commit
```

配置DeviceB。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
```

```
[*HUAWEI] commit
[~DeviceB] vlan batch 200
[*DeviceB] interface 100ge 1/0/1
[*DeviceB-100GE1/0/1] portswitch
[*DeviceB-100GE1/0/1] port link-type hybrid
[*DeviceB-100GE1/0/1] port hybrid pvid vlan 200
[*DeviceB-100GE1/0/1] port hybrid untagged vlan 200
[*DeviceB-100GE1/0/1] quit
[*DeviceB] interface vlanif 200
[*DeviceB-Vlanif200] ip address 10.10.20.2 24
[*DeviceB-Vlanif200] quit
[*DeviceB] commit
```

步骤2 配置DHCPv4中继。

在DeviceA的接口VLANIF100下使能DHCPv4中继功能，并配置DHCPv4服务器的IPv4地址。

```
[~DeviceA] dhcp enable
[*DeviceA] interface vlanif 100
[*DeviceA-Vlanif100] ip address 10.20.20.1 24
[*DeviceA-Vlanif100] dhcp select relay
[*DeviceA-Vlanif100] dhcp relay server-ip 10.10.20.2
[*DeviceA-Vlanif100] quit
[*DeviceA] commit
```

步骤3 配置DHCPv4服务器。

在DeviceB上配置基于全局地址池的DHCPv4服务器功能。

```
[~DeviceB] dhcp enable
[*DeviceB] interface vlanif 200
[*DeviceB-Vlanif200] dhcp select global
[*DeviceB-Vlanif200] quit
[*DeviceB] ip pool pool1
[*DeviceB-ip-pool-pool1] network 10.20.20.0 mask 24
[*DeviceB-ip-pool-pool1] gateway-list 10.20.20.1
[*DeviceB-ip-pool-pool1] option121 ip-address 10.10.20.0 24 10.20.20.1
[*DeviceB-ip-pool-pool1] quit
[*DeviceB] commit
```

步骤4 配置路由。

在DeviceA上配置缺省路由。

```
[~DeviceA] ip route-static 0.0.0.0 0.0.0.0 10.10.20.2
[*DeviceA] commit
```

在DeviceB上配置静态路由。

```
[~DeviceB] ip route-static 10.20.20.0 255.255.255.0 10.10.20.1
[*DeviceB] commit
```

----结束

检查配置结果

在DeviceA上执行命令**display dhcp relay interface vlanif 100**命令用来查看DHCPv4中继的配置信息。

```
[~DeviceA] display dhcp relay interface vlanif 100
 Server IP address [00] : 10.10.20.2
 Gateway address in use : 10.20.20.1
 Gateway switch         : Disable
```

在DHCPv4客户端上查看IPv4地址信息，可以看到已经成功获取到IPv4地址。

配置脚本

- DeviceA的配置文件

```
#
sysname DeviceA
#
vlan batch 100 200
#
dhcp enable
#
interface Vlanif100
 ip address 10.20.20.1 255.255.255.0
 dhcp select relay
 dhcp relay server-ip 10.10.20.2
#
interface Vlanif200
 ip address 10.10.20.1 255.255.255.0
#
interface 100GE1/0/1
 port link-type hybrid
 port hybrid pvid vlan 200
 port hybrid untagged vlan 200
#
interface 100GE1/0/2
 port link-type hybrid
 port hybrid pvid vlan 100
 port hybrid untagged vlan 100
#
ip route-static 0.0.0.0 0.0.0.0 10.10.20.2
#
return
```

- DeviceB的配置文件

```
#
sysname DeviceB
#
vlan batch 200
#
dhcp enable
#
ip pool pool1
 gateway-list 10.20.20.1
 network 10.20.20.0 mask 255.255.255.0
 option121 ip-address 10.10.20.0 24 10.20.20.1
#
interface Vlanif200
 ip address 10.10.20.2 255.255.255.0
 dhcp select global
#
interface 100GE1/0/1
 port link-type hybrid
 port hybrid pvid vlan 200
 port hybrid untagged vlan 200
#
ip route-static 10.20.20.0 255.255.255.0 10.10.20.1
#
return
```

2.5 IP 路由

2.5.1 IPv4 静态路由

2.5.1.1 举例：配置静态 BFD 检测 IPv4 静态路由

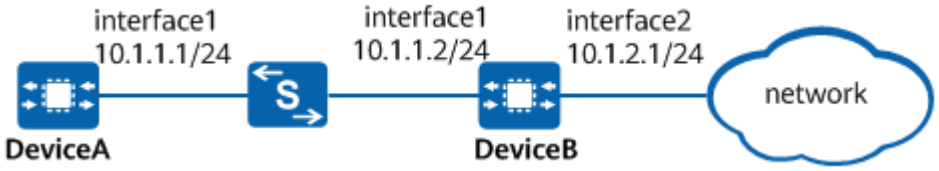
组网需求

如图2-13所示，DeviceA通过交换机和DeviceB相连。在DeviceA上配置静态缺省路由可以与外部进行正常通信。同时，在DeviceA和DeviceB之间配置BFD会话来快速检测链路故障。

图 2-13 配置静态 BFD for IPv4 静态路由组网图

说明

本例中interface1，interface2分别代表100GE1/0/1，100GE1/0/2。



配置思路

采用如下思路配置静态BFD for IPv4静态路由：

1. 配置各路由设备接口IP地址。
2. 在DeviceA和DeviceB上配置BFD会话，检测DeviceA和DeviceB之间的链路。
3. 配置DeviceA到外部的缺省路由并绑定BFD会话。

操作步骤

步骤1 配置各路由设备接口IPv4地址。

配置DeviceA。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] interface 100ge 1/0/1
[~DeviceA-100GE1/0/1] undo portswitch
[*DeviceA-100GE1/0/1] ip address 10.1.1.1 255.255.255.0
[*DeviceA-100GE1/0/1] quit
[*DeviceA] commit
```

DeviceB的配置过程与DeviceA类似，在此不再赘述，具体请参考配置脚本。

步骤2 在DeviceA和DeviceB上配置BFD会话，检测DeviceA和DeviceB之间的链路。

在DeviceA上配置与DeviceB之间的BFD Session。

```
[~DeviceA] bfd
[*DeviceA-bfd] quit
[*DeviceA] bfd aa bind peer-ip 10.1.1.2
[*DeviceA-bfd-session-aa] discriminator local 10
[*DeviceA-bfd-session-aa] discriminator remote 20
[*LSRDeviceA-bfd-session-aa] quit
[*DeviceA] commit
```

在DeviceB上配置与DeviceA之间的BFD Session。

```
[~DeviceB] bfd
[*DeviceB-bfd] quit
```

```
[*DeviceB] bfd bb bind peer-ip 10.1.1.1
[*DeviceB-bfd-session-bb] discriminator local 20
[*DeviceB-bfd-session-bb] discriminator remote 10
[*DeviceB-bfd-session-bb] quit
[*DeviceB] commit
```

步骤3 配置DeviceA到外部的缺省路由并绑定BFD会话。

在DeviceA上配置到外部网络的静态缺省路由，并绑定BFD会话aa。

```
[~DeviceA] ip route-static 0.0.0.0 0 10.1.1.2 track bfd-session aa
[*DeviceA] commit
```

----结束

检查配置结果

在DeviceA上执行**display bfd session all**命令，查看静态BFD的状态。

```
[~DeviceA] display bfd session all
(w): State in WTR
(*): State is invalid
--------------------------------------------------------------------------------
Local  Remote PeerIpAddr       State     Type        InterfaceName
--------------------------------------------------------------------------------
10     20     10.1.1.2        Up        S_IP_IF       100GE1/0/1
--------------------------------------------------------------------------------
Total UP/DOWN Session Number : 1/0
```

State字段是UP，可以看到BFD会话已经建立。

在DeviceA上查看IP路由表，静态路由存在于路由表中。

```
[~DeviceA] display ip routing-table
Route Flags: R - relay, D - download to fib, T - to vpn-instance, B - black hole route
------------------------------------------------------------------------------
Routing Table: Public
         Destinations : 5        Routes : 5
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

      0.0.0.0/0  Static 60   0         RD  10.1.1.2      100GE1/0/1
      10.1.1.0/24  Direct  0    0           D   10.1.1.1        100GE1/0/1
      10.1.1.1/32  Direct  0    0           D   127.0.0.1       100GE1/0/1
    10.1.1.255/32  Direct  0    0           D   127.0.0.1       100GE1/0/1
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

从路由表中可以看到缺省IPv4静态路由存在路由表中。

对DeviceA的接口100GE1/0/1执行**shutdown**命令模拟链路故障。

```
[~DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] shutdown
[*DeviceA] commit
```

在DeviceA上执行**display bfd session all**命令，查看静态BFD的状态。

```
[~DeviceA] display bfd session all
(w): State in WTR
(*): State is invalid
--------------------------------------------------------------------------------
Local  Remote PeerIpAddr       State     Type        InterfaceName
--------------------------------------------------------------------------------
10     20     10.1.1.2        Down      S_IP_IF       100GE1/0/1
--------------------------------------------------------------------------------
Total UP/DOWN Session Number : 0/1
```

State字段是DOWN，可以看到BFD会话Down掉了。

查看DeviceA的路由表。

```
[~DeviceA] display ip routing-table
Route Flags: R - relay, D - download to fib, T - to vpn-instance, B - black hole route
------------------------------------------------------------------------------
Routing Table: Public
         Destinations : 4        Routes : 4
Destination/Mask    Proto  Pre  Cost      Flags NextHop         Interface
        10.1.1.0/24  Direct 0    0           D  10.1.1.1        100GE1/0/1
        10.1.1.1/32  Direct 0    0           D  127.0.0.1       100GE1/0/1
      10.1.1.255/32  Direct 0    0           D  127.0.0.1       100GE1/0/1
255.255.255.255/32   Direct 0    0           D  127.0.0.1       InLoopBack0
```

DeviceA上的缺省IPv4静态路由0.0.0.0/0也已经不存在路由表中。

配置脚本

- DeviceA

```
#
sysname DeviceA
#
bfd
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.1 255.255.255.0
#
ip route-static 0.0.0.0 0.0.0.0 10.1.1.2 track bfd-session aa
#
bfd aa bind peer-ip 10.1.1.2
 discriminator local 10
 discriminator remote 20
#
return
```

- DeviceB

```
#
sysname DeviceB
#
bfd
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.2 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.2.1 255.255.255.0
#
bfd bb bind peer-ip 10.1.1.1
 discriminator local 20
 discriminator remote 10
#
return
```

2.5.2 OSPF

2.5.2.1 举例：配置 BFD for OSPF

组网需求

OSPF通过周期性的向邻居发送Hello报文来实现邻居检测，检测到故障所需时间比较长，超过1秒钟。随着科技的发展，语音、视频及其他点播业务应用广泛，而这些业务对于丢包率和延时非常敏感，当数据达到吉比特速率级时，较长的检测时间会导致大

量数据丢失，无法满足电信级网络高可靠性的需求。通过配置BFD for OSPF特性，可以快速检测链路的状态，故障检测时间可以达到毫秒级，提高链路状态变化时OSPF的收敛速度。

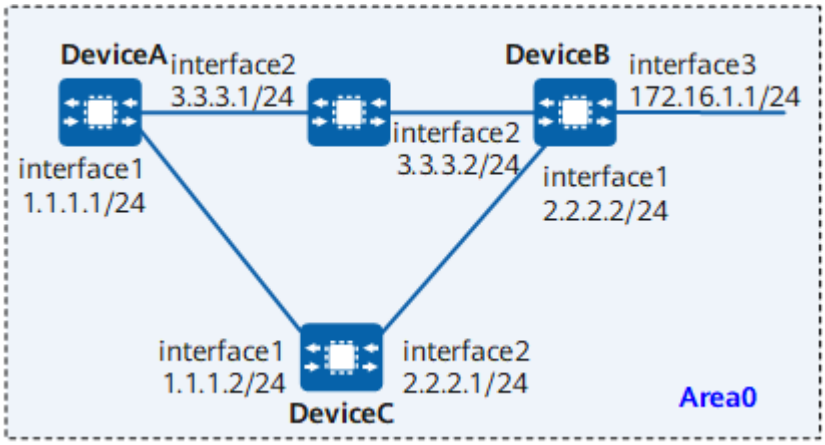
例如，如图2-14所示。网络部署为主/备链路，主链路为DeviceA→DeviceB，备链路为DeviceA→DeviceC→DeviceB。正常情况下，业务流量在主链路上传送。当主链路故障时，用户希望能够快速感知，及时把业务流量切换到备份链路上。

此时，可以配置BFD for OSPF功能，使用BFD检测DeviceA和DeviceB之间的OSPF邻居关系，当DeviceA和DeviceB之间的链路发生故障时，BFD能够快速检测到故障并通告给OSPF协议，使业务流量切换到备份链路上传送。

图 2-14 配置 BFD for OSPF 特性组网图

说明

本例中interface1，interface2，interface3分别代表100GE1/0/1，100GE1/0/2，100GE1/0/3。



配置注意事项

为了提升安全性，推荐部署OSPF区域验证方式或接口验证方式（参见“提高OSPF网络的安全性”）。其中，以配置OSPF区域验证方式为例，详细配置方法请参见“举例：配置OSPF基本功能”。

配置思路

采用如下思路配置BFD for OSPF：

1. 在各设备上配置OSPF基本功能，实现路由互通。
2. 使能全局BFD特性。
3. 在DeviceA、DeviceB和DeviceC上使能指定进程的BFD for OSPF。

操作步骤

步骤1 配置各接口的IP地址。

请参考图2-14，配置各接口的IP地址，具体配置过程请参见配置脚本。

步骤2 配置OSPF基本功能。

具体配置过程请参见配置脚本。

步骤3 配置指定进程的BFD for OSPF。

配置DeviceA。

```
[~DeviceA] bfd
[*DeviceA-bfd] quit
[*DeviceA] ospf 1
[*DeviceA-ospf-1] bfd all-interfaces enable
[*DeviceA-ospf-1] quit
[*DeviceA] commit
```

配置DeviceB。

```
[~DeviceB] bfd
[*DeviceB-bfd] quit
[*DeviceB] ospf 1
[*DeviceB-ospf-1] bfd all-interfaces enable
[*DeviceB-ospf-1] quit
[*DeviceB] commit
```

配置DeviceC。

```
[~DeviceC] bfd
[*DeviceC-bfd] quit
[*DeviceC] ospf 1
[*DeviceC-ospf-1] bfd all-interfaces enable
[*DeviceC-ospf-1] quit
[*DeviceC] commit
```

----结束

检查配置结果

配置完成后，在DeviceA或DeviceB、DeviceC上执行**display ospf bfd session all**命令，可以看到BFDState的状态为Up。

以DeviceA的显示为例。

```
[~DeviceA] display ospf bfd session all
        OSPF Process 1 with Router ID 1.1.1.1
 Area 0.0.0.0 interface 1.1.1.1(100GE1/0/1)'s BFD Sessions

NeighborId:2.2.2.2          AreaId:0.0.0.0          Interface: 100GE1/0/1
BFDState:up                 rx    :1000             tx       :1000
Multiplier:3                BFD Local Dis:0         LocalIpAdd:1.1.1.1
RemoteIpAdd:1.1.1.2         Diagnostic Info:0

 Area 0.0.0.0 interface 3.3.3.1(100GE1/0/2)'s BFD Sessions

NeighborId:3.3.3.3          AreaId:0.0.0.0          Interface: 100GE1/0/2
BFDState:up                 rx    :1000             tx       :1000
Multiplier:3                BFD Local Dis:0         LocalIpAdd:3.3.3.1
RemoteIpAdd:3.3.3.2         Diagnostic Info:0
```

对DeviceB的100GE1/0/2接口执行**shutdown**命令，模拟主链路故障。

```
[~DeviceB] interface 100ge 1/0/2
[~DeviceB-100GE1/0/2] shutdown
```

在DeviceA上，查看路由表。可以看出，在主链路失效后，备份链路DeviceA-DeviceC-DeviceB生效，去往172.16.1.0/24的路由下一跳地址为1.1.1.2。

```
[~DeviceA] display ospf routing
        OSPF Process 1 with Router ID 1.1.1.1
              Routing Tables

 Routing for Network
 Destination        Cost      Type       NextHop         AdvRouter       Area
```

```
2.2.2.0/24        2         Stub       1.1.1.2         3.3.3.3         0.0.0.0
172.16.1.0/24     3         Stub       1.1.1.2         2.2.2.2         0.0.0.0

Total Nets: 2
Intra Area: 2  Inter Area: 0  ASE: 0  NSSA: 0
```

配置脚本

- DeviceA

```
#
sysname DeviceA
#
router id 1.1.1.1
#
bfd
#
interface 100GE1/0/1
 undo portswitch
 ip address 1.1.1.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 3.3.3.1 255.255.255.0
#
ospf 1
 bfd all-interfaces enable
 area 0.0.0.0
  network 3.3.3.0 0.0.0.255
  network 1.1.1.0 0.0.0.255
#
return
```

- DeviceB

```
#
sysname DeviceB
#
router id 2.2.2.2
#
bfd
#
interface 100GE1/0/1
 undo portswitch
 ip address 2.2.2.2 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 3.3.3.2 255.255.255.0
#
interface 100GE1/0/3
 undo portswitch
 ip address 172.16.1.1 255.255.255.0
#
ospf 1
 bfd all-interfaces enable
 area 0.0.0.0
  network 3.3.3.0 0.0.0.255
  network 2.2.2.0 0.0.0.255
  network 172.16.1.0 0.0.0.255
#
return
```

- DeviceC

```
#
sysname DeviceC
#
router id 3.3.3.3
#
bfd
```

```
#
interface 100GE1/0/1
 undo portswitch
 ip address 1.1.1.2 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 2.2.2.1 255.255.255.0
#
ospf 1
 bfd all-interfaces enable
 area 0.0.0.0
  network 1.1.1.0 0.0.0.255
  network 2.2.2.0 0.0.0.255
#
return
```

2.5.3 IS-IS

2.5.3.1 举例：配置动态 BFD for IS-IS

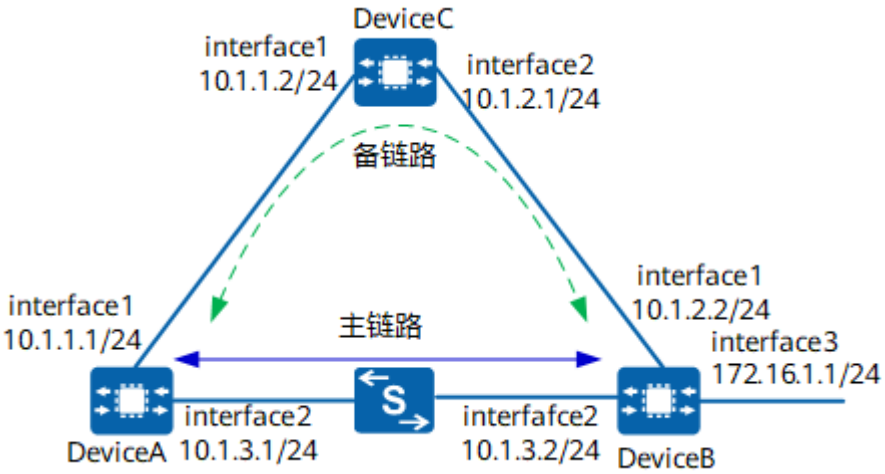
组网需求

如图2-15所示，DeviceA、DeviceB和DeviceC之间运行IS-IS协议。使能DeviceA、DeviceB和DeviceC的IS-IS进程BFD特性。业务流量在主链路DeviceA→DeviceB上传送，链路DeviceA→DeviceC→DeviceB作为备份链路。在DeviceA和DeviceB之间的链路上创建接口的BFD特性，当DeviceA和DeviceB之间的链路出现故障时，BFD能够快速检测到故障并通告给IS-IS协议，使业务流量使用备份链路传送。

图 2-15 配置动态 BFD for IS-IS 组网图

说明

本例中interface1，interface2和interface3分别代表100GE1/0/1，100GE1/0/2，100GE1/0/3。



配置注意事项

为了提升安全性，推荐部署IS-IS认证功能（参见“配置IS-IS认证”）。其中，以配置IS-IS接口认证为例，详细配置方法请参见“举例：配置IS-IS基本功能”。

配置思路

采用如下思路配置BFD for IS-IS特性：

1. 在各设备上使能IS-IS基本功能，保证各路由的连通。
2. 配置IS-IS接口开销值控制路由的选路功能。
3. 使能全局BFD特性。
4. 在DeviceA、DeviceB和DeviceC上使能IS-IS进程的BFD检测机制。
5. 在DeviceA和DeviceB上使能接口的BFD检测机制。

操作步骤

步骤1 配置各设备接口的IPv4地址。

配置DeviceA。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] interface 100ge 1/0/1
[~DeviceA-100GE1/0/1] undo portswitch
[*DeviceA-100GE1/0/1] ip address 10.1.1.1 255.255.255.0
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] undo portswitch
[*DeviceA-100GE1/0/2] ip address 10.1.3.1 255.255.255.0
[*DeviceA-100GE1/0/2] quit
[*DeviceA] commit
```

DeviceB和DeviceC的配置过程与DeviceA类似，在此不再赘述，具体请参考配置脚本。

步骤2 配置IS-IS基本功能。

配置DeviceA。

```
[~DeviceA] isis
[*DeviceA-isis-1] is-level level-2
[*DeviceA-isis-1] network-entity 10.0000.0000.0001.00
[*DeviceA-isis-1] quit
[*DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] isis enable 1
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] isis enable 1
[*DeviceA-100GE1/0/2] quit
[*DeviceA] commit
```

DeviceB和DeviceC的配置过程与DeviceA类似，在此不再赘述，具体请参考配置脚本。

配置完成后，使用**display isis peer**命令，可以查看DeviceA和DeviceB、DeviceA和DeviceC建立了邻居关系。以DeviceA为例。

```
[~DeviceA] display isis peer
--------------------------------------------------------------------------------
0000.0000.0002  100GE1/0/2          0000.0000.0002.01 Up   9s      L2      64
0000.0000.0003  100GE1/0/1          0000.0000.0001.02 Up   21s     L2      64
Total Peer(s): 2
```

设备之间已经互相学到路由。以查看DeviceA的路由表为例。

```
[~DeviceA] display ip routing-table
Route Flags: R - relay, D - download to fib, T - to vpn-instance, B - black hole route
```

```
------------------------------------------------------------------------------
Routing Table: _public_
         Destinations : 8        Routes : 9
Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface
        10.1.1.0/24  Direct  0    0           D  10.1.1.1        100GE1/0/1
        10.1.1.1/32  Direct  0    0           D  127.0.0.1       InLoopBack0
        10.1.2.0/24  ISIS    15   20          D  10.1.1.2        100GE1/0/1
        10.1.3.0/24  Direct  0    0           D  10.1.3.1        100GE1/0/2
        10.1.3.1/32  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
     172.16.1.0/24  ISIS    15   20          D  10.1.3.2        100GE1/0/2
```

从路由表可以看出，去往172.16.1.0/24的路由下一跳地址为10.1.3.2，流量在主链路DeviceA→DeviceB上传输。

步骤3 配置接口开销值。

配置DeviceA。

```
[~DeviceA] interface 100ge 1/0/2
[~DeviceA-100GE1/0/2] isis cost 5
[*DeviceA-100GE1/0/2] quit
[*DeviceA] commit
```

DeviceB的配置过程与DeviceA类似，在此不再赘述，具体请参考配置脚本。

步骤4 配置IS-IS进程的BFD特性。

在DeviceA上使能IS-IS的BFD特性。

```
[~DeviceA] bfd
[*DeviceA-bfd] quit
[*DeviceA] isis
[*DeviceA-isis-1] bfd all-interfaces enable
[*DeviceA-isis-1] quit
[*DeviceA] commit
```

DeviceB和DeviceC的配置过程与DeviceA类似，在此不再赘述，具体请参考配置脚本。

配置完成后，在DeviceA、DeviceB或DeviceC上执行**display isis bfd session all**命令，可以看到BFD State的状态为Up。以DeviceA的显示为例。

```
[~DeviceA] display isis bfd session all
Peer System ID : 0000.0000.0002        Interface : 100GE1/0/2
TX : 10           BFD State : up      Peer IP Address : 10.1.3.2
RX : 10           LocDis : 16385      Local IP Address: 10.1.3.1
Multiplier : 3     RemDis : 16388      Type : L2
Diag : No diagnostic information

Peer System ID : 0000.0000.0003        Interface : 100GE1/0/1
TX : 10           BFD State : up      Peer IP Address : 10.1.1.2
RX : 10           LocDis : 16386      Local IP Address: 10.1.1.1
Multiplier : 3     RemDis : 16387      Type : L2
Diag : No diagnostic information
Total BFD session(s): 2
```

从上面信息可以看出，DeviceA与DeviceB、DeviceC的BFD会话状态为Up。

步骤5 配置接口的BFD特性。

在DeviceA的100GE1/0/2接口上配置BFD特性，并指定最小发送和接收间隔为100ms本地检测时间倍数为4。

```
[~DeviceA] interface 100ge 1/0/2
[~DeviceA-100GE1/0/2] isis bfd enable
[*DeviceA-100GE1/0/2] isis bfd min-tx-interval 100 min-rx-interval 100 detect-multiplier 4
```

```
[*DeviceA-100GE1/0/2] quit
[*DeviceA] commit
```

在DeviceB的100GE1/0/2接口上配置BFD特性，并指定最小发送和接收间隔为100ms本地检测时间倍数为4。

```
[~DeviceB] interface 100ge 1/0/2
[~DeviceB-100GE1/0/2] isis bfd enable
[*DeviceB-100GE1/0/2] isis bfd min-tx-interval 100 min-rx-interval 100 detect-multiplier 4
[*DeviceB-100GE1/0/2] quit
[*DeviceB] commit
```

----结束

检查配置结果

配置完成后，在DeviceB上执行**display isis bfd session all**命令，可以看到BFD参数已生效。

```
[~DeviceB] display isis bfd session all
Peer System ID : 0000.0000.0001        Interface : 100GE1/0/2
TX : 100           BFD State : up      Peer IP Address : 10.1.3.1
RX : 100           LocDis : 16385      Local IP Address: 10.1.3.2
Multiplier : 4     RemDis : 16385      Type : L2
Diag : No diagnostic information

Peer System ID : 0000.0000.0003        Interface : 100GE1/0/1
TX : 10            BFD State : up      Peer IP Address : 10.1.2.1
RX : 10            LocDis : 16385      Local IP Address: 10.1.2.2
Multiplier : 4     RemDis : 16385      Type : L2
Diag : No diagnostic information
Total BFD session (s):2
```

对DeviceB的100GE1/0/2接口执行**shutdown**命令，模拟主链路故障。

```
[~DeviceB] interface 100ge 1/0/2
[~DeviceB-100GE1/0/2] shutdown
[*DeviceB] commit
```

在DeviceA上，查看IP路由表。

```
[~DeviceA] display ip routing-table
Route Flags: R - relay, D - download to fib, T - to vpn-instance, B - black hole route
------------------------------------------------------------------------------
Routing Table : _public_
         Destinations : 9        Routes : 9

Destination/Mask    Proto   Pre  Cost        Flags NextHop         Interface

        10.1.1.0/24  Direct 0    0           D  10.1.1.1        100GE1/0/1
        10.1.1.1/32  Direct 0    0           D  127.0.0.1       100GE1/0/1
      10.1.1.255/32  Direct 0    0           D  127.0.0.1       100GE1/0/1
        10.1.2.0/24  ISIS  15   20           D  10.1.1.2        100GE1/0/1
      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
127.255.255.255/32   Direct 0    0           D  127.0.0.1       InLoopBack0
  172.16.1.0/24   ISIS  15   30           D  10.1.1.2        100GE1/0/1
255.255.255.255/32   Direct 0    0           D  127.0.0.1       InLoopBack0
```

从路由表可以看出，在主链路失效后，备份链路DeviceA-DeviceC-DeviceB生效，去往172.16.1.0/24的路由下一跳地址为10.1.1.2。

在DeviceA上执行**display isis bfd session all**命令，只能看到DeviceA和DeviceC之间的BFD State的状态为Up。

```
[~DeviceA] display isis bfd session all
Peer System ID : 0000.0000.0003        Interface : 100GE1/0/1
```

```
TX : 10          BFD State : up      Peer IP Address : 10.1.1.2
RX : 10          LocDis : 16385      Local IP Address: 10.1.1.1
Multiplier : 3     RemDis : 16388      Type : L2
Diag : No diagnostic information
Total BFD session (s):1
```

配置脚本

- DeviceA

```
#
sysname DeviceA
#
bfd
#
isis 1
 is-level level-2
 bfd all-interfaces enable
 network-entity 10.0000.0000.0001.00
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.1 255.255.255.0
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.3.1 255.255.255.0
 isis enable 1
 isis cost 5
 isis bfd enable
 isis bfd min-tx-interval 100 min-rx-interval 100 detect-multiplier 4
#
return
```

- DeviceB

```
#
sysname DeviceB
#
bfd
#
isis 1
 is-level level-2
 bfd all-interfaces enable
 network-entity 10.0000.0000.0002.00
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.2.2 255.255.255.0
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.3.2 255.255.255.0
 isis enable 1
 isis cost 5
 isis bfd enable
 isis bfd min-tx-interval 100 min-rx-interval 100 detect-multiplier 4
#
interface 100GE1/0/3
 undo portswitch
 ip address 172.16.1.1 255.255.255.0
 isis enable 1
#
return
```

- DeviceC

```
#
sysname DeviceC
```



```
#
bfd
#
isis 1
 is-level level-2
 bfd all-interfaces enable
 network-entity 10.0000.0000.0003.00
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.2 255.255.255.0
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.1.2.1 255.255.255.0
 isis enable 1
#
return
```

2.6 IP 组播

2.6.1 PIM

2.6.1.1 举例：配置 ASM 模型的 PIM-SM

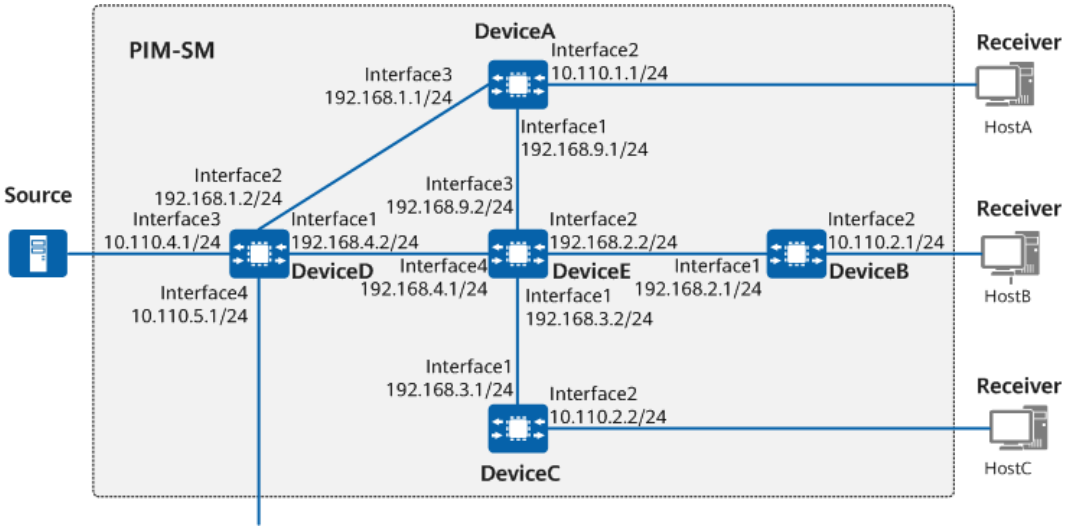
组网需求

在如图2-16所示的ISP（Internet Service Provider）网络中部署组播业务。该网络中已经部署了完备的IGP，单播运行正常，并接入Internet。要求网络中的用户主机能够通过组播方式接收视频点播信息。可以同时配置静态RP和动态RP，也可以只配置其中一种。同时配置两种RP时，优先选择动态RP，静态RP作为备份。

图 2-16 配置 ASM 模型的 PIM-SM 组网图

说明

本例中interface 1，interface 2，interface 3 ，interface4分别代表100GE1/0/1，100GE1/0/2，100GE1/0/3和100GE1/0/4。



配置注意事项

在配置过程中，需注意以下事项：

- 使用静态RP（Rendezvous Point）时，所有设备上需要配置完全相同的静态RP。
- 用户需要接收指定组播源发送的数据时，使用SSM模型的PIM-SM服务。确保所有设备上的SSM组地址范围相同。

配置思路

采用如下思路配置ASM模型的PIM-SM组播功能：

1. 配置各设备的接口IP地址和单播路由协议。
2. 在所有提供组播服务的设备上使能组播功能，在各接口上使能PIM-SM功能。
3. 在与主机直连的设备接口上使能IGMP。
4. 配置RP。在PIM-SM网络中，RP是RPT（Rendezvous Point Tree）的根节点。建议将RP的位置配置在组播流量分支较多的设备上，如图2-16中的DeviceE的位置。
5. （可选）在DeviceD与Internet相连的接口上配置BSR边界。

操作步骤

步骤1 配置各设备的接口IP地址和单播路由协议，具体配置过程请参考配置脚本。

步骤2 在所有设备上使能组播功能，在各接口上使能PIM-SM功能。

以DeviceE为例，使能PIM-SM功能。其余设备配置相似，配置过程略。

```
[~DeviceE] multicast routing-enable
[*DeviceE] interface 100GE 1/0/1
[*DeviceE-100GE1/0/1] undo portswitch
[*DeviceE-100GE1/0/1] pim sm
[*DeviceE-100GE1/0/1] quit
[*DeviceE] interface 100GE 1/0/2
[*DeviceE-100GE1/0/2] undo portswitch
[*DeviceE-100GE1/0/2] pim sm
[*DeviceE-100GE1/0/2] quit
[*DeviceE] interface 100GE 1/0/3
[*DeviceE-100GE1/0/3] undo portswitch
[*DeviceE-100GE1/0/3] pim sm
[*DeviceE-100GE1/0/3] quit
[*DeviceE] interface 100GE 1/0/4
[*DeviceE-100GE1/0/4] undo portswitch
[*DeviceE-100GE1/0/4] pim sm
[*DeviceE-100GE1/0/4] quit
[*DeviceE] commit
```

步骤3 在连接用户主机的接口上使能IGMP功能。

以DeviceA为例，使能IGMP功能。

```
[~DeviceA] interface 100GE 1/0/2
[~DeviceA-100GE1/0/2] undo portswitch
[*DeviceA-100GE1/0/2] igmp enable
[*DeviceA-100GE1/0/2] igmp version 2
[*DeviceA-100GE1/0/2] quit
[*DeviceA-100GE1/0/2] commit
```

步骤4 配置RP。

配置动态RP，需要在PIM-SM域的一个或多个设备上进行如下配置。本例中在DeviceE上配置RP的服务范围，及C-BSR（Candidate-BootStrap Router）和C-RP（Candidate-Rendezvous Point）的位置。

```
[~DeviceE] acl number 2000
[*DeviceE-acl4-basic-2000] rule permit source 225.1.1.0 0.0.0.255
[*DeviceE-acl4-basic-2000] quit
[*DeviceE] pim
[*DeviceE-pim] c-bsr 100GE 1/0/2
[*DeviceE-pim] c-rp 100GE 1/0/2 group-policy 2000 priority 0
[*DeviceE-pim] quit
[*DeviceE-pim] commit
```

配置静态RP，需要在所有组播设备上配置。DeviceA、DeviceB、DeviceC和DeviceD上的配置过程与DeviceE上的配置相似，配置过程略。可以通过在命令**static-rp** *rp-address*后面设置参数**preferred**来优先选择静态RP。

```
[~DeviceE] pim
[*DeviceE-pim] static-rp 192.168.4.1
[*DeviceE-pim] quit
[*DeviceE-pim] commit
```

步骤5 （可选）在DeviceD与Internet相连的接口上配置BSR边界。

```
[~DeviceD] interface 100GE 1/0/4
[~DeviceD-100GE1/0/4] undo portswitch
[*DeviceD-100GE1/0/4] pim bsr-boundary
[*DeviceD-100GE1/0/4] quit
[*DeviceD-100GE1/0/4] commit
```

----结束

检查配置结果

通过使用**display pim interface**命令可以查看设备上的PIM接口信息。例如DeviceE上PIM接口的信息如下：

```
<DeviceE> display pim interface
VPN-Instance: public net
Interface  State NbrCnt HelloInt   DR-Pri     DR-Address
100GE1/0/1   up     1       30        1       192.168.3.2
100GE1/0/2   up     1       30        1       192.168.2.2
100GE1/0/3   up     1       30        1       192.168.9.2
100GE1/0/4   up     1       30        1       192.168.4.2
```

通过使用**display pim bsr-info**命令可以查看设备上的BSR信息。例如DeviceD和DeviceE上的BSR信息分别如下（DeviceE上还显示C-BSR信息）：

```
<DeviceD> display pim bsr-info
 VPN-Instance: public net
 Elected AdminScope BSR Count: 0
 Elected BSR Address: 192.168.2.2
     Priority: 0
     Hash mask length: 30
     State: Accept Preferred
     Scope: Not scoped
     Uptime: 02:08:57
     Expires: 00:01:15
     C-RP Count: 1
<DeviceE> display pim bsr-info
 VPN-Instance: public net
 Elected AdminScope BSR Count: 0
 Elected BSR Address: 192.168.2.2
     Priority: 0
     Hash mask length: 30
     State: Elected
     Scope: Not scoped
```



```
    Uptime: 02:25:03
    Next BSR message scheduled at: 00:00:57
    C-RP Count: 1
 Candidate AdminScope BSR Count: 0
 Candidate BSR Address: 192.168.2.2
    Priority: 0
    Hash mask length: 30
    State: Elected
    Scope: Not scoped
    Wait to be BSR: 0
```

通过使用**display pim rp-info**命令可以查看设备上的RP信息。例如DeviceD和DeviceE上的RP信息分别如下：

```
<DeviceD> display pim rp-info
 VPN-Instance: public net
 PIM-SM BSR RP Number:1
 Group/MaskLen: 225.1.1.0/24
    RP: 192.168.2.2
    Priority: 0
    Uptime: 02:27:17
    Expires: 00:02:15
 PIM SM static RP Number:1
    Static RP: 192.168.4.1
<DeviceE> display pim rp-info
 VPN-Instance: public net
 PIM-SM BSR RP Number:1
 Group/MaskLen: 225.1.1.0/24
    RP: 192.168.2.2 (local)
    Priority: 0
    Uptime: 02:27:27
    Expires: 00:02:03
 PIM SM static RP Number:1
    Static RP: 192.168.4.1 (local)
```

通过使用**display pim routing-table**命令可以查看设备上的PIM路由表。例如DeviceD和DeviceE上的PIM路由表信息如下：

```
<DeviceD> display pim routing-table
 VPN-Instance: public net
 Total 0 (*, G) entry; 2 (S, G) entries

 (10.110.5.100, 225.1.1.1)
    RP: 192.168.2.2
    Protocol: pim-sm, Flag: SPT LOC ACT
    UpTime: 00:57:20
    Upstream interface: 100GE1/0/3, Refresh time: 00:57:20
        Upstream neighbor: NULL
        RPF prime neighbor: 10.110.5.100
    Downstream interface(s) information:
    Total number of downstreams: 1
      1: 100GE1/0/2
           Protocol:  pim-sm, UpTime: 00:57:20, Expires: 00:03:02
<DeviceE> display pim routing-table
 VPN-Instance: public net
 Total 1 (*, G) entry; 1 (S, G) entry

 (*, 225.1.1.1)
    RP: 192.168.2.2 (local)
    Protocol: pim-sm, Flag: WC
    UpTime: 00:21:40
    Upstream interface: register, Refresh time: 00:21:40
        Upstream neighbor: 192.168.4.2
        RPF prime neighbor: 192.168.4.2
    Downstream interface(s) information:
    Total number of downstreams: 1
      1: 100GE1/0/3
           Protocol:  pim-sm, UpTime: 00:21:40, Expires: 00:02:43
```

配置脚本

- DeviceA

```
#
sysname DeviceA
#
multicast routing-enable
isis 1
 network-entity 10.0000.0000.0001.00
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.9.1 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
 pim sm
 igmp enable
 igmp version 2
 isis enable 1
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.110.1.1 255.255.255.0
 pim sm
 isis enable 1
#
pim
 static-rp 192.168.4.1
#
return
```

- DeviceB

```
#
sysname DeviceB
#
multicast routing-enable
isis 1
 network-entity 10.0000.0000.0002.00
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.2.1 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.2.1 255.255.255.0
 pim sm
 igmp enable
 igmp version 2
 isis enable 1
#
pim
 static-rp 192.168.4.1
#
return
```

- DeviceC

```
#
sysname DeviceC
#
multicast routing-enable
isis 1
 network-entity 10.0000.0000.0003.00
```

```
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.3.1 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.2.2 255.255.255.0
 pim sm
 igmp enable
 igmp version 2
 isis enable 1
#
pim
 static-rp 192.168.4.1
#
return
```

- DeviceD

```
#
sysname DeviceD
#
multicast routing-enable
isis 1
 network-entity 10.0000.0000.0004.00
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.4.2 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.110.4.1 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/4
 undo portswitch
 ip address 10.110.5.1 255.255.255.0
 pim bsr-boundary
 pim sm
 isis enable 1
#
pim
 static-rp 192.168.4.1
#
return
```

- DeviceE

```
#
sysname DeviceE
#
multicast routing-enable
#
acl number 2000
 rule 5 permit source 225.1.1.0 0.0.0.255
isis 1
 network-entity 10.0000.0000.0005.00
#
interface 100GE1/0/1
```



```
 undo portswitch
 ip address 192.168.3.2 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.2.2 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.9.2 255.255.255.0
 pim sm
 isis enable 1
#
 interface 100GE1/0/4
 undo portswitch
 ip address 192.168.4.1 255.255.255.0
 pim sm
 isis enable 1
#
pim
 static-rp 192.168.4.1
 c-bsr 100GE 1/0/2
 c-rp 100GE 1/0/2 group-policy 2000
#
return
```

2.6.1.2 举例：配置 SSM 模型的 PIM-SM

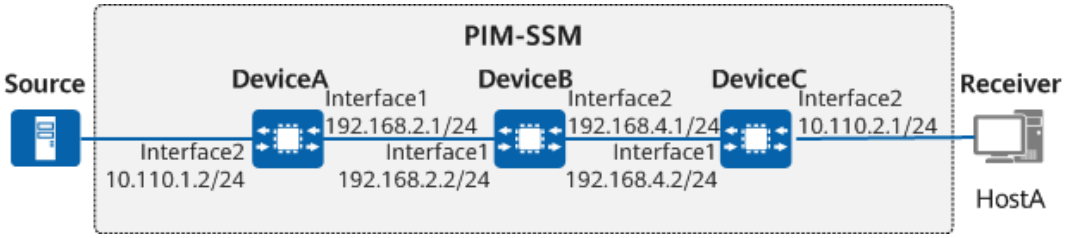
组网需求

在如图2-17所示的ISP（Internet Service Provider）网络中部署组播业务。已知该网络中已经部署了IGP协议，单播运行正常。要求网络中的用户主机在加入组播组时，可以明确指定组播源。

说明

本例中interface 1，interface 2分别代表100GE1/0/1，100GE1/0/2。

图 2-17 配置 SSM 模型的 PIM-SM 组播组网图



配置注意事项

在配置过程中，需注意确保所有设备上的SSM组地址范围相同。

配置思路

采用如下思路配置SSM模型的PIM-SM组播功能：

1. 配置各设备的接口IP地址和单播路由协议。
2. 在所有提供组播服务的设备上使能组播功能，并在各接口上使能PIM-SM功能。
3. 在与主机直连的设备接口上使能IGMP。
4. 在各设备上设置相同的SSM组地址范围。

操作步骤

步骤1 配置各设备的接口IP地址和单播路由协议，具体配置过程请参考配置脚本。

步骤2 在所有设备上使能组播功能，在各接口上使能PIM-SM功能。

DeviceB和DeviceC上的配置过程与DeviceA上的配置相似，配置过程略。

```
[~DeviceA] multicast routing-enable
[*DeviceA] interface 100GE 1/0/2
[*DeviceA-100GE1/0/2] undo portswitch
[*DeviceA-100GE1/0/2] pim sm
[*DeviceA-100GE1/0/2] quit
[*DeviceA] interface 100GE 1/0/1
[*DeviceA-100GE1/0/1] undo portswitch
[*DeviceA-100GE1/0/1] pim sm
[*DeviceA-100GE1/0/1] quit
[*DeviceA-100GE1/0/1] commit
```

步骤3 在连接用户主机的接口上使能IGMP功能。

在DeviceC连接用户主机的接口上使能IGMP，并将IGMP版本号配置为3。

```
[~DeviceC] interface 100GE 1/0/2
[~DeviceC-100GE1/0/2] undo portswitch
[*DeviceC-100GE1/0/2] igmp enable
[*DeviceC-100GE1/0/2] igmp version 3
[*DeviceC-100GE1/0/2] quit
[*DeviceC-100GE1/0/2] commit
```

步骤4 配置SSM组地址范围。

在所有设备上配置SSM组播组地址范围为232.1.1.0/24。DeviceB和DeviceC上的配置过程与DeviceA上的配置完全相同，配置过程略。

```
[~DeviceA] acl number 2000
[*DeviceA-acl4-basic-2000] rule permit source 232.1.1.0 0.0.0.255
[*DeviceA-acl4-basic-2000] quit
[*DeviceA] pim
[*DeviceA-pim] ssm-policy 2000
[*DeviceA-pim] quit
[*DeviceA-pim] commit
```

----结束

检查配置结果

HostA需要接收组播源（10.110.1.1/24）发往组播组（232.1.1.1/24）的信息。通过使用**display pim routing-table**命令可以查看设备上的PIM路由表。DeviceA和DeviceB上的显示信息如下：

```
<DeviceA> display pim routing-table
VPN-Instance: public net
 Total 0 (*, G) entry; 1 (S, G) entry

 (10.110.1.1, 232.1.1.1)
     Protocol: pim-ssm, Flag: LOC
     UpTime: 00:02:13
```



```
    Upstream interface: 100GE1/0/2, Refresh time: 00:02:13
        Upstream neighbor: 10.110.1.1
        RPF prime neighbor: 10.110.1.1
    Downstream interface(s) information:
    Total number of downstreams: 1
      1: 100GE1/0/1
           Protocol:  pim-ssm, UpTime: 00:02:13, Expires: 00:03:17

<DeviceB> display pim routing-table
 VPN-Instance: public net
 Total 0 (*, G) entry; 1 (S, G) entry

 (10.110.1.1, 232.1.1.1)
    Protocol: pim-ssm, Flag:
    UpTime: 00:09:15
    Upstream interface: 100GE1/0/1, Refresh time: 00:09:15
        Upstream neighbor: 192.168.2.1
        RPF prime neighbor: 192.168.2.1
    Downstream interface(s) information:
    Total number of downstreams: 1
      1: 100GE1/0/2
           Protocol:  pim-ssm, UpTime: 00:09:15, Expires: 00:03:13
```

配置脚本

- DeviceA

```
#
sysname DeviceA
#
multicast routing-enable
#
acl number 2000
 rule 5 permit source 232.1.1.0 0.0.0.255
#
isis 1
 network-entity 10.0000.0000.0001.00
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.2.1 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.1.2 255.255.255.0
 pim sm
 isis enable 1
#
pim
 ssm-policy 2000
#
return
```

- DeviceB

```
#
sysname DeviceB
#
multicast routing-enable
#
acl number 2000
 rule 5 permit source 232.1.1.0 0.0.0.255
#
isis 1
 network-entity 10.0000.0000.0002.00
#
interface 100GE1/0/1
 undo portswitch
```

```
 ip address 192.168.2.2 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.4.1 255.255.255.0
 pim sm
 isis enable 1
#
pim
 ssm-policy 2000
#
return
```

- DeviceC

```
#
sysname DeviceC
#
multicast routing-enable
#
acl number 2000
 rule 5 permit source 232.1.1.0 0.0.0.255
#
isis 1
 network-entity 10.0000.0000.0003.00
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.4.2 255.255.255.0
 pim sm
 isis enable 1
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.2.1 255.255.255.0
 pim sm
 igmp enable
 igmp version 3
 isis enable 1
#
pim
 ssm-policy 2000
#
return
```

2.6.1.3 举例：配置基于 PIM 的 Anycast RP

组网需求

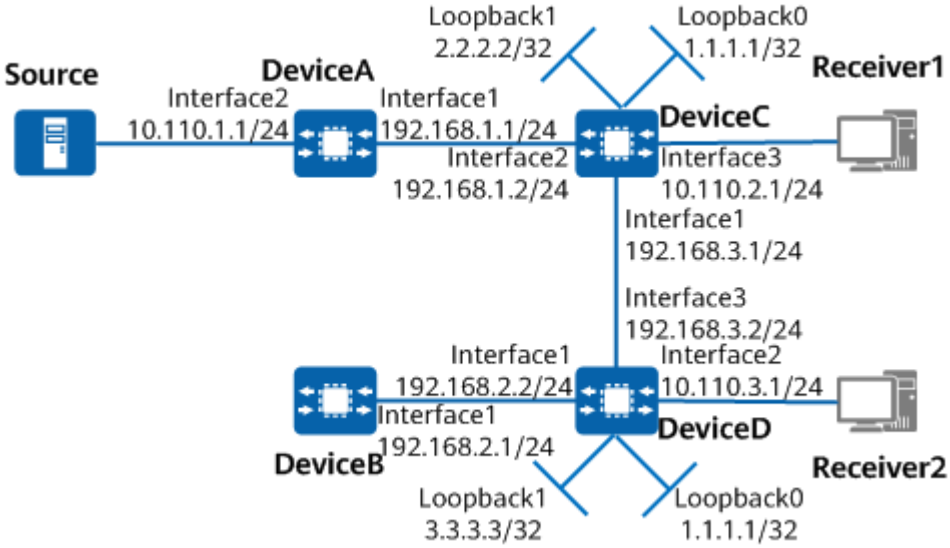
PIM-SM域内有多个组播源和多个接收者，通过配置Anycast RP（Rendezvous Point）对等体，可实现组播源就近注册和接收者就近加入，缓解RP负担，优化组播数据转发路径。

在传统的PIM-SM域中，每个组播组都只能映射到一个RP。当网络负载较大或流量过于集中时，可能导致RP压力过大、RP失效后路由收敛较慢、组播转发路径非最优等问题。在单自治域中应用基于PIM协议的Anycast RP，可实现组播源就近注册和接收者就近加入。从而使接收者快速接收到组播数据。如图2-18所示，Receiver2需要接收Source的组播数据，配置DeviceC和DeviceD为Anycast RP对等体，Receiver2就近加入DeviceD，DeviceA收到Source的组播数据后，封装成注册消息向DeviceC注册，DeviceC收到注册报文后，将注册报文转发给DeviceD，Receiver2可以收到组播源的数据。

说明

本例中interface1，interface2，interface3分别代表100GE1/0/1，100GE1/0/2，100GE1/0/3。

图 2-18 配置基于 PIM 协议的 Anycast RP 组网图



为完成此配置例，需准备如下的数据：

- 组播组地址：226.1.1.1/24
- RP地址
- Anycast RP本地地址

配置思路

采用如下的思路配置基于PIM协议的Anycast RP功能：

1. 配置各设备的接口IP地址，采用OSPF协议实现网络层互通。
2. 使能组播功能，在各接口启动PIM-SM功能。
3. 在设备与主机侧相连的接口使能IGMP功能。
4. 配置DeviceC和DeviceD的Loopback0接口为C-RP（Candidate-Rendezvous Point）和C-BSR（Candidate-BootStrap Router）。
5. 配置DeviceC和DeviceD的Loopback0接口地址为Anycast RP地址。
6. 配置DeviceC和DeviceD的Loopback1接口地址为各自的Anycast RP本地地址。
7. 配置DeviceC和DeviceD互为Anycast RP对等体。

操作步骤

步骤1 配置各设备的接口IP地址，采用OSPF协议实现网络层互通。使能组播功能，在各接口启动PIM-SM功能

按照图2-18，在PIM-SM域内，配置各设备接口的IP地址和掩码，配置各设备之间采用OSPF进行互连。使能组播功能，在各接口启动PIM-SM功能。

配置DeviceA。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] multicast routing-enable
[*DeviceA] interface 100GE 1/0/1
[*DeviceA-100GE1/0/1] undo portswitch
[*DeviceA-100GE1/0/1] ip address 10.110.1.1 24
[*DeviceA-100GE1/0/1] pim sm
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100GE 1/0/2
[*DeviceA-100GE1/0/2] undo portswitch
[*DeviceA-100GE1/0/2] ip address 192.168.1.1 24
[*DeviceA-100GE1/0/2] pim sm
[*DeviceA-100GE1/0/2] quit
[*DeviceA] ospf
[*DeviceA-ospf-1] area 0
[*DeviceA-ospf-1-area-0.0.0.0] network 10.110.1.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[*DeviceA-ospf-1-area-0.0.0.0] quit
[*DeviceA-ospf-1] quit
[*DeviceA] commit
```

配置DeviceB。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*HUAWEI] commit
[~DeviceB] multicast routing-enable
[*DeviceB] interface 100GE 1/0/1
[*DeviceB-100GE1/0/1] undo portswitch
[*DeviceB-100GE1/0/1] ip address 192.168.2.1 24
[*DeviceB-100GE1/0/1] pim sm
[*DeviceB-100GE1/0/1] quit
[*DeviceB] ospf
[*DeviceB-ospf-1] area 0
[*DeviceB-ospf-1-area-0.0.0.0] network 192.168.2.0 0.0.0.255
[*DeviceB-ospf-1-area-0.0.0.0] quit
[*DeviceB-ospf-1] quit
[*DeviceB] commit
```

配置DeviceC。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceC
[*HUAWEI] commit
[~DeviceC] multicast routing-enable
[*DeviceC] interface 100GE 1/0/1
[*DeviceC-100GE1/0/1] undo portswitch
[*DeviceC-100GE1/0/1] ip address 192.168.3.1 24
[*DeviceC-100GE1/0/1] pim sm
[*DeviceC-100GE1/0/1] quit
[*DeviceC] interface 100GE 1/0/2
[*DeviceC-100GE1/0/2] undo portswitch
[*DeviceC-100GE1/0/2] ip address 192.168.1.2 24
[*DeviceC-100GE1/0/2] pim sm
[*DeviceC-100GE1/0/2] quit
[*DeviceC] interface 100GE 1/0/3
[*DeviceC-100GE1/0/3] undo portswitch
[*DeviceC-100GE1/0/3] ip address 10.110.2.1 24
[*DeviceC-100GE1/0/3] pim sm
[*DeviceC-100GE1/0/3] quit
[*DeviceC] interface loopback 0
[*DeviceC-LoopBack0] ip address 1.1.1.1 32
[*DeviceC-LoopBack0] pim sm
[*DeviceC-LoopBack0] quit
[*DeviceC] interface loopback 1
[*DeviceC-LoopBack1] ip address 2.2.2.2 32
[*DeviceC-LoopBack1] pim sm
[*DeviceC-LoopBack1] quit
[*DeviceC] ospf
```



```
[*DeviceC-ospf-1] area 0
[*DeviceC-ospf-1-area-0.0.0.0] network 192.168.3.0 0.0.0.255
[*DeviceC-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[*DeviceC-ospf-1-area-0.0.0.0] network 10.110.2.0 0.0.0.255
[*DeviceC-ospf-1-area-0.0.0.0] network 1.1.1.1 0.0.0.0
[*DeviceC-ospf-1-area-0.0.0.0] network 2.2.2.2 0.0.0.0
[*DeviceC-ospf-1-area-0.0.0.0] quit
[*DeviceC-ospf-1] quit
[*DeviceC] commit
```

配置DeviceD。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceD
[*HUAWEI] commit
[~DeviceD] multicast routing-enable
[*DeviceD] interface 100GE1/0/1
[*DeviceD-100GE1/0/1] undo portswitch
[*DeviceD-100GE1/0/1] ip address 192.168.2.2 24
[*DeviceD-100GE1/0/1] pim sm
[*DeviceD-100GE1/0/1] quit
[*DeviceD] interface 100GE 1/0/2
[*DeviceD-100GE1/0/2] undo portswitch
[*DeviceD-100GE1/0/2] ip address 10.110.3.1 24
[*DeviceD-100GE1/0/2] pim sm
[*DeviceD-100GE1/0/2] quit
[*DeviceD] interface 100GE 1/0/3
[*DeviceD-100GE1/0/3] undo portswitch
[*DeviceD-100GE1/0/3] ip address 192.168.3.2 24
[*DeviceD-100GE1/0/3] pim sm
[*DeviceD-100GE1/0/3] quit
[*DeviceD] interface loopback 0
[*DeviceD-LoopBack0] ip address 1.1.1.1 32
[*DeviceD-LoopBack0] pim sm
[*DeviceD-LoopBack0] quit
[*DeviceD] interface loopback 1
[*DeviceD-LoopBack1] ip address 3.3.3.3 32
[*DeviceD-LoopBack1] pim sm
[*DeviceD-LoopBack1] quit
[*DeviceD] ospf
[*DeviceD-ospf-1] area 0
[*DeviceD-ospf-1-area-0.0.0.0] network 192.168.2.0 0.0.0.255
[*DeviceD-ospf-1-area-0.0.0.0] network 10.110.3.0 0.0.0.255
[*DeviceD-ospf-1-area-0.0.0.0] network 192.168.3.0 0.0.0.255
[*DeviceD-ospf-1-area-0.0.0.0] network 1.1.1.1 0.0.0.0
[*DeviceD-ospf-1-area-0.0.0.0] network 3.3.3.3 0.0.0.0
[*DeviceD-ospf-1-area-0.0.0.0] quit
[*DeviceD-ospf-1] quit
[*DeviceD] commit
```

步骤2 在设备与主机侧相连的接口使能IGMP功能。

在DeviceC和DeviceD与主机侧相连的接口使能IGMP功能。

配置DeviceC。

```
[~DeviceC] interface 100GE 1/0/3
[~DeviceC-100GE1/0/3] igmp enable
[*DeviceC-100GE1/0/3] quit
[*DeviceC] commit
```

配置DeviceD。

```
[~DeviceD] interface 100GE 1/0/2
[~DeviceD-100GE1/0/2] igmp enable
[*DeviceD-100GE1/0/2] quit
[*DeviceD] commit
```

步骤3 配置DeviceC和DeviceD的Loopback0接口为C-RP和C-BSR。

配置DeviceC。

```
[~DeviceC] pim
[*DeviceC-pim] c-bsr loopback 0
[*DeviceC-pim] c-rp loopback 0
```

配置DeviceD。

```
[~DeviceD] pim
[*DeviceD-pim] c-bsr loopback 0
[*DeviceD-pim] c-rp loopback 0
```

步骤4 配置DeviceC和DeviceD的Loopback0接口地址为Anycast RP地址。

配置DeviceC。

```
[~DeviceC-pim] anycast-rp 1.1.1.1
[*DeviceC-pim-anycast-rp-1.1.1.1] quit
```

配置DeviceD。

```
[~DeviceD-pim] anycast-rp 1.1.1.1
[*DeviceD-pim-anycast-rp-1.1.1.1] quit
```

步骤5 配置DeviceC和DeviceD的Loopback1接口地址为各自的Anycast RP本地地址。

配置DeviceC。

```
[*DeviceC-pim] anycast-rp 1.1.1.1
[*DeviceC-pim-anycast-rp-1.1.1.1] local-address 2.2.2.2
[*DeviceC-pim-anycast-rp-1.1.1.1] quit
```

配置DeviceD。

```
[*DeviceD-pim] anycast-rp 1.1.1.1
[*DeviceD-pim-anycast-rp-1.1.1.1] local-address 3.3.3.3
[*DeviceD-pim-anycast-rp-1.1.1.1] quit
```

步骤6 配置DeviceC和DeviceD互为Anycast RP对等体。

配置DeviceC。

```
[*DeviceC-pim] anycast-rp 1.1.1.1
[*DeviceC-pim-anycast-rp-1.1.1.1] peer 3.3.3.3
[*DeviceC-pim-anycast-rp-1.1.1.1] quit
[*DeviceC-pim] quit
[*DeviceC] commit
```

配置DeviceD。

```
[*DeviceD-pim] anycast-rp 1.1.1.1
[*DeviceD-pim-anycast-rp-1.1.1.1] peer 2.2.2.2
[*DeviceD-pim-anycast-rp-1.1.1.1] quit
[*DeviceD-pim] quit
[*DeviceD] commit
```

----结束

检查配置结果

通过使用**display pim rp-info**命令可以查看DeviceC和DeviceD上的RP信息。

```
<DeviceC> display pim rp-info
 VPN-Instance: public net
 PIM-SM BSR RP Number:1
 Group/MaskLen: 224.0.0.0/4
    RP: 1.1.1.1 (local)
    Priority: 0
```

```
    Uptime: 00:45:19
    Expires: 00:02:11
<DeviceD> display pim rp-info
 VPN-Instance: public net
 PIM-SM BSR RP Number:1
 Group/MaskLen: 224.0.0.0/4
    RP: 1.1.1.1 (local)
    Priority: 0
    Uptime: 02:27:56
    Expires: 00:01:39
```

由以上显示信息可知，DeviceC和DeviceD都作为网络中的RP，可以相互转发组播源注册信息。

通过使用**display pim routing-table**命令可以查看设备上的PIM表项。PIM-SM域内组播源Source（10.110.1.2/24）向组播组G（226.1.1.1）发送组播信息，用户Receiver2加入组播组G，接收发往组G的组播数据。Source向DeviceC注册，Receiver2向DeviceD发起加入。

```
<DeviceC> display pim routing-table
 VPN-Instance: public net
 Total 0 (*, G) entry; 1 (S, G) entries

 (10.110.1.2, 226.1.1.1)
    RP: 1.1.1.1 (local)
    Protocol: pim-sm, Flag: 2MSDP ACT
    UpTime: 00:00:38
    Upstream interface: Register, Refresh time: 00:00:38
        Upstream neighbor: NULL
        RPF prime neighbor: NULL
    Downstream interface(s) information: None
<DeviceD> display pim routing-table
 VPN-Instance: public net
 Total 1 (*, G) entry; 1 (S, G) entries

 (*, 226.1.1.1)
    RP: 1.1.1.1 (local)
    Protocol: pim-sm, Flag: WC
    UpTime: 00:01:25
    Upstream interface: Register, Refresh time: 00:01:25
        Upstream neighbor: NULL
        RPF prime neighbor: NULL
    Downstream interface(s) information:
    Total number of downstreams: 1
        1: 100GE1/0/2
            Protocol: igmp, UpTime: 00:01:25, Expires: -

 (10.110.1.2, 226.1.1.1)
    RP: 1.1.1.1 (local)
    Protocol: pim-sm, Flag: 2MSDP SWT ACT
    UpTime: 00:00:02
    Upstream interface: Register, Refresh time: 00:00:02
        Upstream neighbor: NULL
        RPF prime neighbor: NULL
    Downstream interface(s) information:
    Total number of downstreams: 1
        1: 100GE1/0/2
            Protocol: pim-sm, UpTime: 00:00:02, Expires: -
```

配置脚本

- DeviceA

```
#
sysname DeviceA
#
multicast routing-enable
#
```

```
interface 100GE1/0/1
 undo portswitch
 ip address 10.110.1.1 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
 pim sm
#
ospf 1
 area 0.0.0.0
  network 10.110.1.0 0.0.0.255
  network 192.168.1.0 0.0.0.255
#
return
```

- DeviceB

```
#
sysname DeviceB
#
multicast routing-enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.2.1 255.255.255.0
 pim sm
#
ospf 1
 area 0.0.0.0
   network 192.168.2.0 0.0.0.255
#
return
```

- DeviceC

```
#
sysname DeviceC
#
multicast routing-enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.3.1 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
 pim sm
#
interface 100GE 1/0/3
 undo portswitch
 ip address 10.110.2.1 255.255.255.0
 pim sm
 igmp enable
#
interface LoopBack0
 ip address 1.1.1.1 255.255.255.255
 pim sm
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
 pim sm
#
ospf 1
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.3.0 0.0.0.255
  network 10.110.2.0 0.0.0.255
  network 1.1.1.1 0.0.0.0
```

```
 network 2.2.2.2 0.0.0.0
#
pim
 c-bsr LoopBack0
 c-rp LoopBack0
 anycast-rp 1.1.1.1
  local-address 2.2.2.2
  peer 3.3.3.3
#
return
```

- DeviceD

```
#
sysname DeviceD
#
multicast routing-enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.2.2 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.3.1 255.255.255.0
 pim sm
 igmp enable
#
interface 100GE 1/0/3
 undo portswitch
 ip address 192.168.3.2 255.255.255.0
 pim sm
#
interface LoopBack0
 ip address 1.1.1.1 255.255.255.255
 pim sm
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.0
 pim sm
#
ospf 1
 area 0.0.0.0
  network 192.168.2.0 0.0.0.255
  network 192.168.3.0 0.0.0.255
  network 10.110.3.0 0.0.0.255
  network 3.3.3.3 0.0.0.0
  network 1.1.1.1 0.0.0.0
#
pim
 c-bsr LoopBack0
 c-rp LoopBack0
 anycast-rp 1.1.1.1
  local-address 3.3.3.3
  peer 2.2.2.2
#
return
```

2.6.2 MSDP

2.6.2.1 举例：配置 Anycast RP

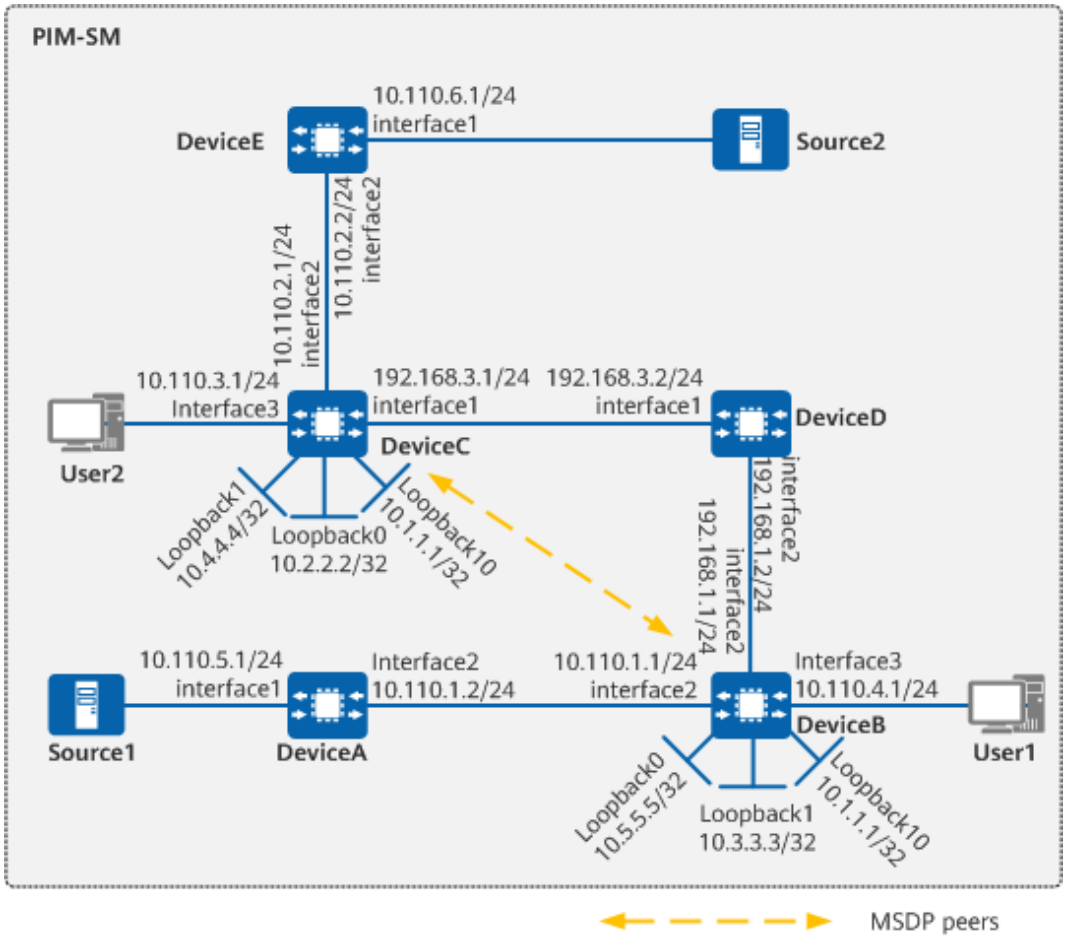
组网需求

如图2-19所示，PIM-SM域拥有多个组播源和多个接收者。要求在PIM-SM域内建立MSDP对等体实现RP负荷分担。

图 2-19 配置 PIM-SM 域间组播组网图

说明

本示例中的interface1、interface2、interface3分别代表接口100GE1/0/1，100GE1/0/2和100GE1/0/3。



配置思路

配置基于MSDP的Anycast RP，接收者向拓扑距离最近的RP发起加入，组播源向拓扑距离最近的RP发起注册，从而实现RP负载分担。

1. 使能组播功能，并配置各设备接口IP地址，在PIM-SM域内配置OSPF协议实现互联。
2. 在各接口上使能PIM-SM功能，在主机侧接口上使能IGMP功能。
3. 在DeviceB和DeviceC的Loopback10接口地址相同，配置C-RP。在Loopback1接口上配置C-BSR。

4. 在DeviceB和DeviceC的Loopback0接口上配置MSDP对等体。根据RPF规则，接收源RP发来的SA消息。

操作步骤

步骤1 使能组播功能，在PIM-SM域内配置各接口的IP地址和掩码以及Loopback接口，配置各设备间采用OSPF进行互连

配置DeviceB。DeviceB和DeviceC上需要配置相同的Loopback10接口地址。其他设备的配置过程与DeviceB相似，具体配置过程参见配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*HUAWEI] commit
[~DeviceB] multicast routing-enable
[*DeviceB] interface 100ge 1/0/1
[*DeviceB-100GE1/0/1] undo portswitch
[*DeviceB-100GE1/0/1] ip address 192.168.1.1 24
[*DeviceB-100GE1/0/1] quit
[*DeviceB] interface 100ge 1/0/2
[*DeviceB-100GE1/0/2] undo portswitch
[*DeviceB-100GE1/0/2] ip address 10.110.1.1 24
[*DeviceB-100GE1/0/2] quit
[*DeviceB] interface 100ge 1/0/3
[*DeviceB-100GE1/0/3] undo portswitch
[*DeviceB-100GE1/0/3] ip address 10.110.4.1 24
[*DeviceB-100GE1/0/3] quit
[~DeviceB] interface loopback 0
[*DeviceB-LoopBack0] ip address 10.5.5.5 255.255.255.255
[*DeviceB-LoopBack0] quit
[~DeviceB] interface loopback 10
[*DeviceB-LoopBack10] ip address 10.1.1.1 255.255.255.255
[*DeviceB-LoopBack10] quit
[~DeviceB] interface loopback 1
[*DeviceB-LoopBack1] ip address 10.3.3.3 255.255.255.255
[*DeviceB-LoopBack1] quit
[*DeviceB] ospf 1
[*DeviceB-ospf-1] area 0.0.0.0
[*DeviceB-ospf-1-area-0.0.0.0] network 10.5.5.5 0.0.0.0
[*DeviceB-ospf-1-area-0.0.0.0] network 10.3.3.3 0.0.0.0
[*DeviceB-ospf-1-area-0.0.0.0] network 10.1.1.1 0.0.0.0
[*DeviceB-ospf-1-area-0.0.0.0] network 10.110.1.0 0.0.0.255
[*DeviceB-ospf-1-area-0.0.0.0] network 10.110.4.0 0.0.0.255
[*DeviceB-ospf-1-area-0.0.0.0] network 192.168.1.0 0.0.0.255
[*DeviceB-ospf-1-area-0.0.0.0] quit
[*DeviceB-ospf-1] quit
[*DeviceB] commit
```

步骤2 配置PIM-SM功能

在所有设备的各接口上使能PIM-SM功能，在主机侧接口使能IGMP功能。其他设备的配置过程与DeviceB相似，配置过程参见配置脚本。

```
[~DeviceB] interface 100ge 1/0/3
[*DeviceB-100GE1/0/3] pim sm
[*DeviceB-100GE1/0/8] igmp enable
[*DeviceB-100GE1/0/3] quit
[*DeviceB] interface 100ge 1/0/2
[*DeviceB-100GE1/0/2] pim sm
[*DeviceB-100GE1/0/2] quit
[*DeviceB] interface 100ge 1/0/1
[*DeviceB-100GE1/0/1] pim sm
[*DeviceB-100GE1/0/1] quit
[*DeviceB] commit
```

步骤3 配置Loopback1、Loopback10接口，C-BSR、C-RP的位置

在DeviceB和DeviceC上的Loopback1上配置C-BSR，在Loopback10上配置C-RP。DeviceC上的配置过程与DeviceB相似，配置过程参见配置脚本。

```
[~DeviceB] interface loopback 1
[*DeviceB-LoopBack1] pim sm
[*DeviceB-LoopBack1] quit
[*DeviceB] interface loopback 10
[*DeviceB-LoopBack10] pim sm
[*DeviceB-LoopBack10] quit
[*DeviceB] pim
[*DeviceB-pim] c-bsr loopback 1
[*DeviceB-pim] c-rp loopback 10
[*DeviceB-pim] quit
[*DeviceB] commit
```

步骤4 配置Loopback0接口和MSDP对等体

在DeviceB上的Loopback0接口上配置MSDP对等体。

```
[~DeviceB] interface loopback 0
[*DeviceB-LoopBack0] pim sm
[*DeviceB-LoopBack0] quit
[*DeviceB] msdp
[*DeviceB-msdp] originating-rp loopback0
[*DeviceB-msdp] peer 10.2.2.2 connect-interface loopback0
[*DeviceB-msdp] quit
[*DeviceB] commit
```

在DeviceC上的Loopback0接口上配置MSDP对等体。

```
[~DeviceC] interface loopback 0
[*DeviceC-LoopBack0] ip address 10.2.2.2 255.255.255.255
[*DeviceC-LoopBack0] pim sm
[*DeviceC-LoopBack0] quit
[*DeviceC] msdp
[*DeviceC-msdp] originating-rp loopback0
[*DeviceC-msdp] peer 10.5.5.5 connect-interface loopback0
[*DeviceC-msdp] quit
[*DeviceC] commit
```

----结束

检查配置结果

执行命令**display msdp brief**查看设备之间MSDP对等体建立情况。以DeviceB和DeviceC的显示结果为例。

```
[~DeviceB] display msdp brief
MSDP Peer Brief Information of VPN instance: public net
--------------------------------------------------------------------------------
 Configured   Up         Listen      Connect     Shutdown    Down
        1    1            0           0           0       0
--------------------------------------------------------------------------------
 Peer's Address    State    Up/Down time    AS         SA Count   Reset
Count
 10.2.2.2           Up     00:10:17       ?(unknown)       0          0

[~DeviceC] display msdp brief
MSDP Peer Brief Information of VPN instance: public net
--------------------------------------------------------------------------------
 Configured   Up         Listen      Connect     Shutdown    Down
        1    1            0           0           0       0
--------------------------------------------------------------------------------
 Peer's Address    State    Up/Down time    AS         SA Count   Reset
Count
 10.5.5.5           Up     00:10:18       ?(unknown)       0          0
```

执行命令**display pim routing-table**查看DeviceB和DeviceC上的PIM路由。PIM-SM域内组播源S1（10.110.5.100/24）向组播组G（225.1.1.1）发送组播信息，用户

User1加入组播组G，接收发往组G的组播数据。通过比较DeviceB和DeviceC上PIM路由的显示信息，可知当前有效RP是DeviceB：S1向DeviceB注册，User1向DeviceB发起加入。

```
[~DeviceB] display pim routing-table
VPN-Instance: public net
 Total 1 (*, G) entry; 1 (S, G) entry

 (*, 225.1.1.1)
     RP: 10.1.1.1 (local)
     Protocol: pim-sm, Flag: WC
     UpTime: 00:08:49
     Upstream interface: Register, Refresh time: 00:28:49
         Upstream neighbor: NULL
         RPF prime neighbor: NULL
     Downstream interface(s) information:
     Total number of downstreams: 1
         1: 100GE1/0/3
             Protocol: igmp, UpTime: 00:08:49, Expires: -

 (10.110.5.1, 225.1.1.1)
     RP: 10.1.1.1 (local)
     Protocol: pim-sm, Flag: SPT 2MSDP ACT
     UpTime: 00:07:26
     Upstream interface: 100GE1/0/2, Refresh time: 00:28:49
         Upstream neighbor: 10.110.1.2
         RPF prime neighbor: 10.110.1.2
     Downstream interface(s) information:
     Total number of downstreams: 1
         1: 100GE1/0/3
             Protocol: pim-sm, UpTime: 00:07:26, Expires: -
[~DeviceC] display pim routing-table
```

无输出信息。

User1退出组播组G，S1停止向组播组G发送组播数据。使用**reset pim routing-table**清除DeviceB上的PIM路由表项。

```
[~DeviceB] reset pim routing-table group 225.1.1.1 mask 255.255.255.255 source 10.110.5.100 interface 100GE 1/0/3
```

用户User2加入组播组G，S2（10.110.6.100/24）开始向组播组G发送组播数据。通过比较DeviceB和DeviceC上PIM路由的显示信息，可知当前有效RP是DeviceC：S2向DeviceC注册，User2向DeviceC发起加入。

```
[~DeviceB] display pim routing-table
```

无输出信息。

```
[~DeviceC] display pim routing-table
VPN-Instance: public net
Total 1 (*, G) entry; 1 (S, G) entry

(*, 225.1.1.1)
    RP: 10.1.1.1 (local)
    Protocol: pim-sm, Flag: WC
    UpTime: 00:07:23
    Upstream interface: Register, Refresh time: 00:07:23
        Upstream neighbor: NULL
        RPF prime neighbor: NULL
    Downstream interface(s) information:
    Total number of downstreams: 1
        1: 100GE1/0/3,
            Protocol: igmp, UpTime: 00:07:23, Expires:-

 (10.110.6.100, 225.1.1.1)
    RP: 10.1.1.1 (local)
    Protocol: pim-sm, Flag: SPT 2MSDP ACT
```

```
UpTime: 00:05:20
Upstream interface: 100GE1/0/2
    Upstream neighbor: 10.110.2.2
    RPF prime neighbor: 10.110.2.2
Downstream interface(s) information:
Total number of downstreams: 1
    1: 100GE1/0/3
        Protocol: pim-sm, UpTime: 00:05:20, Expires: -
```

配置脚本

- DeviceA

```
#
sysname DeviceA
#
multicast routing-enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.110.5.1 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.1.2 255.255.255.0
 pim sm
#
ospf 1
 area 0.0.0.0
  network 10.110.1.0 0.0.0.255
  network 10.110.5.0 0.0.0.255
#
return
```

- DeviceB

```
#
sysname DeviceB
#
multicast routing-enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
 pim sm
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.1.1 255.255.255.0
 pim sm
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.110.4.1 255.255.255.0
 pim sm
 igmp enable
#
interface LoopBack0
 ip address 10.5.5.5 255.255.255.255
 pim sm
#
interface LoopBack1
 ip address 10.3.3.3 255.255.255.255
 pim sm
#
interface LoopBack10
 ip address 10.1.1.1 255.255.255.255
 pim sm
#
```

```
ospf 1
 area 0.0.0.0
  network 10.5.5.5 0.0.0.0
  network 10.3.3.3 0.0.0.0
  network 10.1.1.1 0.0.0.0
  network 10.110.1.0 0.0.0.255
  network 10.110.4.0 0.0.0.255
  network 192.168.1.0 0.0.0.255
#
pim
 c-bsr LoopBack1
 c-rp LoopBack10
#
msdp
 originating-rp LoopBack0
 peer 10.2.2.2 connect-interface LoopBack0
#
return
```

- DeviceC

```
#
sysname DeviceC
#
multicast routing-enable
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.2.1 255.255.255.0
 pim sm
#
interface 100GE1/0/3
 undo portswitch
 ip address 10.110.3.1 255.255.255.0
 pim sm
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.3.1 255.255.255.0
 pim sm
 igmp enable
#
interface LoopBack0
 ip address 10.2.2.2 255.255.255.255
 pim sm
#
interface LoopBack1
 ip address 10.4.4.4 255.255.255.255
 pim sm
#
interface LoopBack10
 ip address 10.1.1.1 255.255.255.255
 pim sm
#
ospf 1
 area 0.0.0.0
  network 10.2.2.2 0.0.0.0
  network 10.4.4.4 0.0.0.0
  network 10.1.1.1 0.0.0.0
  network 10.110.2.0 0.0.0.255
  network 10.110.3.0 0.0.0.255
  network 192.168.3.0 0.0.0.255
#
pim
 c-bsr LoopBack1
 c-rp LoopBack10
#
msdp
 originating-rp LoopBack0
 peer 10.5.5.5 connect-interface LoopBack0
```

```
#
return
```

- DeviceD

```
#
sysname DeviceD
#
multicast routing-enable
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
 pim sm
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.3.2 255.255.255.0
 pim sm
#
ospf 1
 area 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.3.0 0.0.0.255
#
return
```

- DeviceE

```
#
sysname DeviceE
#
multicast routing-enable
#
interface 100GE1/0/2
 undo portswitch
 ip address 10.110.2.2 255.255.255.0
 pim sm
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.110.6.1 255.255.255.0
 pim sm
#
ospf 1
 area 0.0.0.0
  network 10.110.2.0 0.0.0.255
  network 10.110.6.0 0.0.0.255
#
return
```

2.7 VPN

2.7.1 IPv4 L3VPN

2.7.1.1 举例：配置本地 IPv4 L3VPN 互访

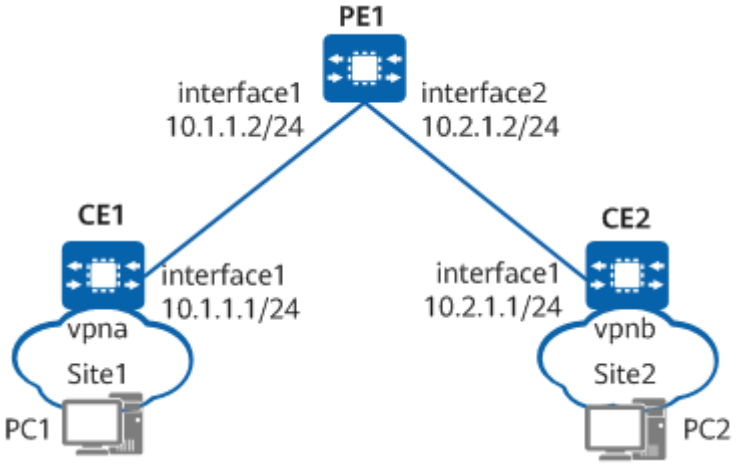
组网需求

如图2-20，CE1和CE2均连接到PE1上，其中CE1属于vpna，CE2属于vpnb。当前由于业务需求，Site1和Site2之间需要互访。可以配置本地VPN互访功能，满足上述需求。

图 2-20 本地 IPv4 VPN 互访组网图

说明

本例中interface1、interface2分别代表100GE1/0/1和100GE1/0/7。



配置思路

本例按如下思路配置VPN互访：

1. 在PE1上配置VPN实例，为VPN实例配置不同的VPN-Target属性，实现不同VPN的隔离。
2. 在PE1上配置与CE相连的接口与VPN实例绑定，接入VPN用户。
3. 在PE1上配置将到本地CE的直连路由引入VPN路由表，同时在CE上配置访问另一CE设备的路由，以实现本地互访。

操作步骤

步骤1 在PE设备上配置VPN实例，将CE接入PE。

配置PE1。

```
<HUAWEI> system-view
[~HUAWEI] sysname PE1
[*HUAWEI] commit
[~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 export-extcommunity
[*PE1-vpn-instance-vpna-af-ipv4] vpn-target 111:1 222:2 import-extcommunity
[*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 export-extcommunity
[*PE1-vpn-instance-vpnb-af-ipv4] vpn-target 222:2 111:1 import-extcommunity
[*PE1-vpn-instance-vpnb-af-ipv4] quit
[*PE1-vpn-instance-vpnb] quit
[*PE1] interface 100ge 1/0/1
[*PE1-100GE1/0/1] undo portswitch
[*PE1-100GE1/0/1] ip binding vpn-instance vpna
[*PE1-100GE1/0/1] ip address 10.1.1.2 24
[*PE1-100GE1/0/1] quit
[*PE1] interface 100ge 1/0/7
```

```
[*PE1-100GE1/0/7] undo portswitch
[*PE1-100GE1/0/7] ip binding vpn-instance vpnb
[*PE1-100GE1/0/7] ip address 10.2.1.2 24
[*PE1-100GE1/0/7] quit
[*PE1] commit
```

配置CE1的接口IP地址。

```
<HUAWEI> system-view
[~HUAWEI] sysname CE1
[*HUAWEI] commit
[~CE1] interface 100ge 1/0/1
[~CE1-100GE1/0/1] undo portswitch
[*CE1-100GE1/0/1] ip address 10.1.1.1 24
[*CE1-100GE1/0/1] commit
[~CE1-100GE1/0/1] quit
```

配置CE2的接口IP地址。

```
<HUAWEI> system-view
[~HUAWEI] sysname CE2
[*HUAWEI] commit
[~CE2] interface 100ge 1/0/1
[~CE2-100GE1/0/1] undo portswitch
[*CE2-100GE1/0/1] ip address 10.2.1.1 24
[*CE2-100GE1/0/1] commit
[~CE2-100GE1/0/1] quit
```

PE能Ping通自己接入的CE。以PE1和CE1为例：

```
[~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=2 ms
    Reply from 10.1.1.1: bytes=56 Sequence=2 ttl=255 time=2 ms
    Reply from 10.1.1.1: bytes=56 Sequence=3 ttl=255 time=2 ms
    Reply from 10.1.1.1: bytes=56 Sequence=4 ttl=255 time=2 ms
    Reply from 10.1.1.1: bytes=56 Sequence=5 ttl=255 time=2 ms

  --- 10.1.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 2/2/2 ms
```

步骤2 配置BGP，将到本地CE的直连路由引入VPN路由表。

配置PE1。

```
[~PE1] bgp 100
[*PE1-bgp] ipv4-family vpn-instance vpna
[*PE1-bgp-vpna] import-route direct
[*PE1-bgp-vpna] quit
[*PE1-bgp] ipv4-family vpn-instance vpnb
[*PE1-bgp-vpnb] import-route direct
[*PE1-bgp-vpnb] quit
[*PE1-bgp] quit
[*PE1] commit
```

步骤3 配置CE上的静态路由。

配置CE1。

```
[~CE1] ip route-static 10.2.1.0 24 10.1.1.2
[*CE1] commit
```

配置CE2。


```
[~CE2] ip route-static 10.1.1.0 24 10.2.1.2
[*CE2] commit
```

----结束

检查配置结果

配置完成后，在PE1上执行**display ip routing-table vpn-instance**可以看到不同VPN路由相互引入，以vpna为例：

```
[~PE1] display ip routing-table vpn-instance vpna
Route Flags: R - relay, D - download to fib, T - to vpn-instance, B - black hole route
------------------------------------------------------------------------------
Routing Table : vpna
         Destinations : 7        Routes : 7

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

        10.1.1.0/24  Direct  0    0           D   10.1.1.2        100GE1/0/1
        10.1.1.2/32  Direct  0    0           D   127.0.0.1       100GE1/0/1
      10.1.1.255/32  Direct  0    0           D   127.0.0.1       100GE1/0/1
        10.2.1.0/24  BGP     255  0           RD  10.2.1.2        100GE1/0/7
        10.2.1.2/32  BGP     255  0           RD  127.0.0.1       100GE1/0/7
       127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
255.255.255.255/32  Direct  0    0           D   127.0.0.1       InLoopBack0
```

CE1和CE2能够相互Ping通。

```
[~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=254 time=8 ms
    Reply from 10.2.1.1: bytes=56 Sequence=2 ttl=254 time=3 ms
    Reply from 10.2.1.1: bytes=56 Sequence=3 ttl=254 time=2 ms
    Reply from 10.2.1.1: bytes=56 Sequence=4 ttl=254 time=3 ms
    Reply from 10.2.1.1: bytes=56 Sequence=5 ttl=254 time=2 ms

  --- 10.2.1.1 ping statistics ---
    5 packet(s) transmitted
    5 packet(s) received
    0.00% packet loss
    round-trip min/avg/max = 2/3/8 ms
```

配置脚本

- PE1

```
#
sysname PE1
#
ip vpn-instance vpna
 ipv4-family
  route-distinguisher 100:1
  vpn-target 111:1 export-extcommunity
  vpn-target 111:1 import-extcommunity
  vpn-target 222:2 import-extcommunity
#
ip vpn-instance vpnb
 ipv4-family
  route-distinguisher 100:2
  vpn-target 222:2 export-extcommunity
  vpn-target 222:2 import-extcommunity
  vpn-target 111:1 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip binding vpn-instance vpna
 ip address 10.1.1.2 255.255.255.0
#
```

```
interface 100GE1/0/7
 undo portswitch
 ip binding vpn-instance vpnb
 ip address 10.2.1.2 255.255.255.0
#
bgp 100
 #
 ipv4-family unicast
 #
 ipv4-family vpn-instance vpna
  import-route direct
 #
 ipv4-family vpn-instance vpnb
  import-route direct
#
return
```

- CE1

```
#
sysname CE1
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.1.1.1 255.255.255.0
#
ip route-static 10.2.1.0 255.255.255.0 10.1.1.2
#
return
```

- CE2

```
#
sysname CE2
#
interface 100GE1/0/1
 undo portswitch
 ip address 10.2.1.1 255.255.255.0
#
ip route-static 10.1.1.0 255.255.255.0 10.2.1.2
#
return
```

2.8 VXLAN

2.8.1 VXLAN

2.8.1.1 举例：配置通过端到端 VXLAN 实现 DCI 互联

组网需求

如图2-21所示，某企业在不同的数据中心中都拥有自己的VM，服务器1上的VMa1属于VLAN 10，服务器2上的VMb2属于VLAN 20，且位于不同网段。现需要通过VXLAN分布式网关，在数据中心A的Leaf1和数据中心B的Leaf4上配置BGP EVPN协议创建VXLAN隧道，实现数据中心A内VMa1和数据中心B内VMb2之间端到端的互相通信。

图 2-21 配置端到端 VXLAN 组网图

说明

本例中interface1、interface2和interface3分别代表100GE1/0/1、100GE1/0/2、100GE1/0/3。

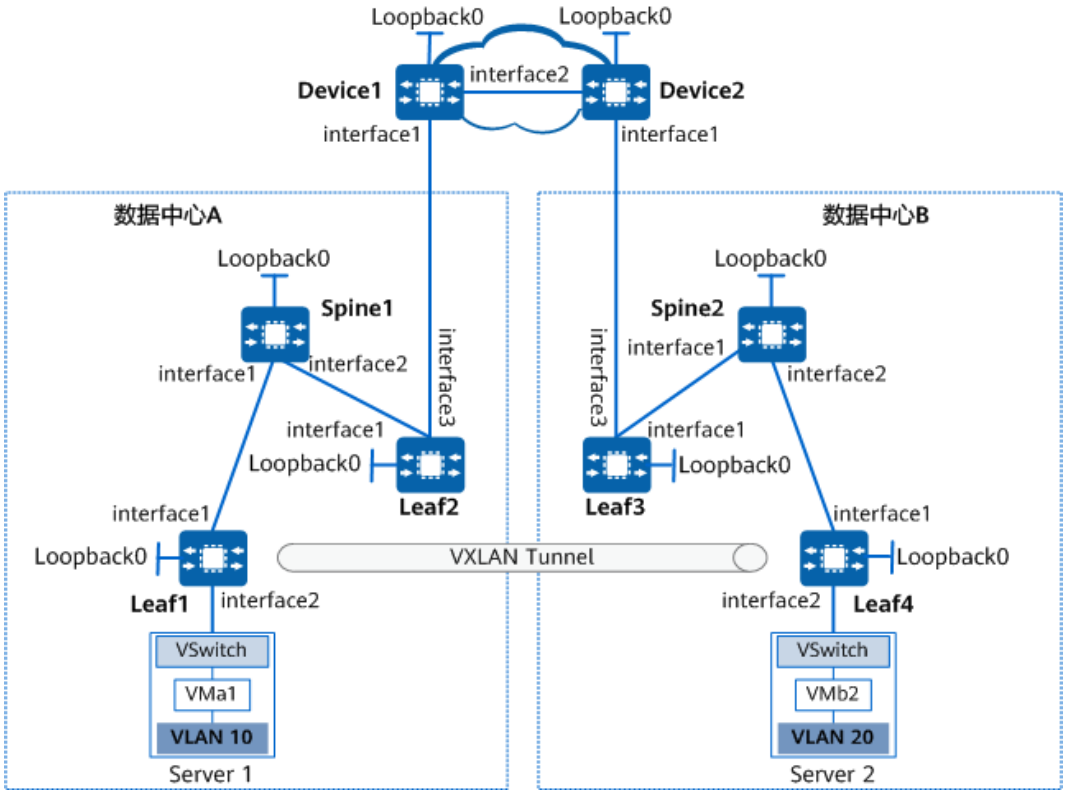


表 2-1 接口的 IP 地址

设备	接口	IP地址	设备	接口	IP地址
Device 1	100GE1/0/1	192.168.50.1/24	Device 2	100GE1/0/1	192.168.60.1/24
	100GE1/0/2	192.168.1.1/24		100GE1/0/2	192.168.1.2/24
	LoopBack0	1.1.1.1/32		LoopBack0	2.2.2.2/32
Spine1	100GE1/0/1	192.168.10.1/24	Spine2	100GE1/0/1	192.168.30.1/24
	100GE1/0/2	192.168.20.1/24		100GE1/0/2	192.168.40.1/24
	LoopBack0	3.3.3.3/32		LoopBack0	4.4.4.4/32
Leaf1	100GE1/0/1	192.168.10.2/24	Leaf4	100GE1/0/1	192.168.40.2/24
	100GE1/0/2	-		100GE1/0/2	-
	LoopBack0	5.5.5.5/32		LoopBack0	8.8.8.8/32
Leaf2	100GE1/0/1	192.168.20.2/24	Leaf3	100GE1/0/1	192.168.30.2/24

设备	接口	IP地址	设备	接口	IP地址
	100GE1/0/3	192.168.50.2/24		100GE1/0/3	192.168.60.2/24
	LoopBack0	6.6.6.6/32		LoopBack0	7.7.7.7/32

配置思路

采用如下的思路配置端到端VXLAN：

1. 配置各节点接口的IP地址。
2. 配置路由协议，实现各节点之间的互通。
3. 配置VXLAN业务接入点。
4. 配置VXLAN隧道。
5. 配置VXLAN三层网关。
6. 配置VXLAN网关之间发布的路由类型。

操作步骤

步骤1 配置各节点接口的IP地址。

配置Device1。其他设备的配置过程与Device1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Device1
[*HUAWEI] commit
[~Device1] interface loopback 0
[*Device1-LoopBack0] ip address 1.1.1.1 32
[*Device1-LoopBack0] quit
[*Device1] interface 100ge 1/0/1
[*Device1-100GE1/0/1] undo portswitch
[*Device1-100GE1/0/1] ip address 192.168.50.1 24
[*Device1-100GE1/0/1] quit
[*Device1] interface 100ge 1/0/2
[*Device1-100GE1/0/2] undo portswitch
[*Device1-100GE1/0/2] ip address 192.168.1.1 24
[*Device1-100GE1/0/2] quit
[*Device1] commit
```

步骤2 在数据中心内配置OSPF，在数据中心间配置BGP，实现路由互通。

配置Device1。Device2的配置过程与Device1类似，在此不再赘述，具体请参考配置脚本。

```
[~Device1] bgp 10
[*Device1-bgp] peer 192.168.1.2 as-number 10
[*Device1-bgp] peer 192.168.50.2 as-number 20
[*Device1-bgp] ipv4-family unicast
[*Device1-bgp-af-ipv4] peer 192.168.1.2 enable
[*Device1-bgp-af-ipv4] peer 192.168.1.2 next-hop-local
[*Device1-bgp-af-ipv4] peer 192.168.50.2 enable
[*Device1-bgp-af-ipv4] quit
[*Device1-bgp] quit
[*Device1] commit
```

配置Spine1。Spine2的配置过程与Spine1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Spine1
[*HUAWEI] commit
[~Spine1] ospf 1
[*Spine1-ospf-1] area 0
[*Spine1-ospf-1-area-0.0.0.0] network 3.3.3.3 0.0.0.0
[*Spine1-ospf-1-area-0.0.0.0] network 192.168.10.0 0.0.0.255
[*Spine1-ospf-1-area-0.0.0.0] network 192.168.20.0 0.0.0.255
[*Spine1-ospf-1-area-0.0.0.0] quit
[*Spine1-ospf-1] quit
[*Spine1] commit
```

配置Leaf1。Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf1
[*HUAWEI] commit
[~Leaf1] ospf 1
[*Leaf1-ospf-1] area 0
[*Leaf1-ospf-1-area-0.0.0.0] network 5.5.5.5 0.0.0.0
[*Leaf1-ospf-1-area-0.0.0.0] network 192.168.10.0 0.0.0.255
[*Leaf1-ospf-1-area-0.0.0.0] quit
[*Leaf1-ospf-1] quit
[*Leaf1] commit
```

配置Leaf2。Leaf3的配置过程与Leaf2类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf2
[*HUAWEI] commit
[~Leaf2] ospf 1
[*Leaf2-ospf-1] import-route bgp
[*Leaf2-ospf-1] area 0
[*Leaf2-ospf-1-area-0.0.0.0] network 6.6.6.6 0.0.0.0
[*Leaf2-ospf-1-area-0.0.0.0] network 192.168.20.0 0.0.0.255
[*Leaf2-ospf-1-area-0.0.0.0] quit
[*Leaf2-ospf-1] quit
[*Leaf2] commit
[~Leaf2] bgp 20
[*Leaf2-bgp] peer 192.168.50.1 as-number 10
[*Leaf2-bgp] ipv4-family unicast
[*Leaf2-bgp-af-ipv4] network 5.5.5.5 255.255.255.255
[*Leaf2-bgp-af-ipv4] network 6.6.6.6 255.255.255.255
[*Leaf2-bgp-af-ipv4] peer 192.168.50.1 enable
[*Leaf2-bgp-af-ipv4] quit
[*Leaf2-bgp] quit
[*Leaf2] commit
```

步骤3 配置VXLAN业务接入点。

配置Leaf1。

```
[~Leaf1] bridge-domain 10
[*Leaf1-bd10] quit
[*Leaf1] interface 100GE 1/0/2.1 mode l2
[*Leaf1-100GE1/0/2.1] encapsulation dot1q vid 10
[*Leaf1-100GE1/0/2.1] bridge-domain 10
[*Leaf1-100GE1/0/2.1] quit
[*Leaf1] commit
```

Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

步骤4 配置VXLAN隧道。

1. 在Leaf1、Leaf2、Leaf3和Leaf4上使能EVPN作为VXLAN控制平面。

   # 配置Leaf1。

   ```
   [~Leaf1] evpn-overlay enable
   [*Leaf1] commit
   ```


Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

2. 在Leaf1和Leaf2之间、在Leaf3和Leaf4之间配置IBGP EVPN对等体关系。

   # 配置Leaf1。

   ```
   [~Leaf1] bgp 100 instance evpn1
   [*Leaf1-bgp-instance-evpn1] peer 6.6.6.6 as-number 100
   [*Leaf1-bgp-instance-evpn1] peer 6.6.6.6 connect-interface LoopBack 0
   [*Leaf1-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf1-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf1-bgp-instance-evpn1-af-evpn] quit
   [*Leaf1-bgp-instance-evpn1] quit
   [*Leaf1] commit
   ```

   # 配置Leaf2。

   ```
   [~Leaf2] bgp 100 instance evpn1
   [*Leaf2-bgp-instance-evpn1] peer 5.5.5.5 as-number 100
   [*Leaf2-bgp-instance-evpn1] peer 5.5.5.5 connect-interface LoopBack 0
   [*Leaf2-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 next-hop-invariable
   [*Leaf2-bgp-instance-evpn1-af-evpn] quit
   [*Leaf2-bgp-instance-evpn1] quit
   [*Leaf2] commit
   ```

   # 配置Leaf3。

   ```
   [~Leaf3] bgp 200 instance evpn1
   [*Leaf3-bgp-instance-evpn1] peer 8.8.8.8 as-number 200
   [*Leaf3-bgp-instance-evpn1] peer 8.8.8.8 connect-interface LoopBack 0
   [*Leaf3-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 next-hop-invariable
   [*Leaf3-bgp-instance-evpn1-af-evpn] quit
   [*Leaf3-bgp-instance-evpn1] quit
   [*Leaf3] commit
   ```

   # 配置Leaf4。

   ```
   [~Leaf4] bgp 200 instance evpn1
   [*Leaf4-bgp-instance-evpn1] peer 7.7.7.7 as-number 200
   [*Leaf4-bgp-instance-evpn1] peer 7.7.7.7 connect-interface LoopBack 0
   [*Leaf4-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf4-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf4-bgp-instance-evpn1-af-evpn] quit
   [*Leaf4-bgp-instance-evpn1] quit
   [*Leaf4] commit
   ```

3. 在Leaf2和Leaf3之间配置EBGP EVPN对等体关系。

   # 配置Leaf2。

   ```
   [~Leaf2] bgp 100 instance evpn1
   [*Leaf2-bgp-instance-evpn1] peer 7.7.7.7 as-number 200
   [*Leaf2-bgp-instance-evpn1] peer 7.7.7.7 connect-interface LoopBack 0
   [*Leaf2-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf2-bgp-instance-evpn1-af-evpn] undo policy vpn-target
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 next-hop-invariable
   [*Leaf2-bgp-instance-evpn1-af-evpn] quit
   [*Leaf2-bgp-instance-evpn1] quit
   [*Leaf2] commit
   ```


配置Leaf3。

```
[~Leaf3] bgp 200 instance evpn1
[*Leaf3-bgp-instance-evpn1] peer 6.6.6.6 as-number 100
[*Leaf3-bgp-instance-evpn1] peer 6.6.6.6 connect-interface LoopBack0
[*Leaf3-bgp-instance-evpn1] l2vpn-family evpn
[*Leaf3-bgp-instance-evpn1-af-evpn] undo policy vpn-target
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 next-hop-invariable
[*Leaf3-bgp-instance-evpn1-af-evpn] quit
[*Leaf3-bgp-instance-evpn1] quit
[*Leaf3] commit
```

4. 配置EVPN实例。

   # 配置Leaf1。

   ```
   [~Leaf1] ip vpn-instance vpn1
   [*Leaf1-vpn-instance-vpn1] vxlan vni 5010
   [*Leaf1-vpn-instance-vpn1] ipv4-family
   [*Leaf1-vpn-instance-vpn1-af-ipv4] route-distinguisher 20:1
   [*Leaf1-vpn-instance-vpn1-af-ipv4] vpn-target 100:5010 evpn
   [*Leaf1-vpn-instance-vpn1-af-ipv4] quit
   [*Leaf1-vpn-instance-vpn1] quit
   [*Leaf1] bridge-domain 10
   [*Leaf1-bd10] vxlan vni 10
   [*Leaf1-bd10] evpn
   [*Leaf1-bd10-evpn] route-distinguisher 10:1
   [*Leaf1-bd10-evpn] vpn-target 100:10
   [*Leaf1-bd10-evpn] vpn-target 100:5010 export-extcommunity
   [*Leaf1-bd10-evpn] quit
   [*Leaf1-bd10] quit
   [*Leaf1] commit
   ```

   Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

5. 在Leaf上使能头端复制功能。

   # 在配置Leaf1。

   ```
   [~Leaf1] interface nve 1
   [*Leaf1-Nve1] source 5.5.5.5
   [*Leaf1-Nve1] vni 10 head-end peer-list protocol bgp
   [*Leaf1-Nve1] quit
   [*Leaf1] commit
   ```

   Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

步骤5 配置VXLAN三层网关。

配置Leaf1。

```
[~Leaf1] interface vbdif 10
[*Leaf1-Vbdif10] ip binding vpn-instance vpn1
[*Leaf1-Vbdif10] ip address 10.1.1.1 24
[*Leaf1-Vbdif10] arp collect host enable
[*Leaf1-Vbdif10] vxlan anycast-gateway enable
[*Leaf1-Vbdif10] quit
[*Leaf1] commit
```

Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

步骤6 配置VXLAN网关之间发布的路由类型。

配置Leaf1。

```
[~Leaf1] bgp 100 instance evpn1
[*Leaf1-bgp-instance-evpn1] l2vpn-family evpn
[*Leaf1-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise irb
```



```
[*Leaf1-bgp-instance-evpn1-af-evpn] quit
[*Leaf1-bgp-instance-evpn1] quit
[*Leaf1] commit
```

配置Leaf2。

```
[~Leaf2] bgp 100 instance evpn1
[*Leaf2-bgp-instance-evpn1] l2vpn-family evpn
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 advertise irb
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 advertise irb
[*Leaf2-bgp-instance-evpn1-af-evpn] quit
[*Leaf2-bgp-instance-evpn1] quit
[*Leaf2] commit
```

Leaf4的配置过程与Leaf1类似，Leaf3的配置过程与Leaf2类似，在此不再赘述，具体请参考配置脚本。

----结束

检查配置结果

上述配置成功后，在Leaf上执行**display vxlan tunnel**命令，可以看到建立的VXLAN隧道信息。以Leaf1的显示为例：

```
[~Leaf1] display vxlan tunnel
Number of vxlan tunnel : 1
Tunnel ID   Source              Destination         State  Type     Uptime
--------------------------------------------------------------------------------
4026531842  5.5.5.5             8.8.8.8             up     dynamic  00:10:16
```

配置完成后，VMa1和VMb2之间可以互相通信。

配置脚本

- Spine1的配置文件

```
#
sysname Spine1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.20.1 255.255.255.0
#
interface LoopBack0
 ip address 3.3.3.3 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 192.168.10.0 0.0.0.255
  network 192.168.20.0 0.0.0.255
#
return
```

- Leaf1的配置文件

```
#
sysname Leaf1
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:1
  vpn-target 100:5010 export-extcommunity evpn
```

```
 vpn-target 100:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:1
  vpn-target 100:10 export-extcommunity
  vpn-target 100:5010 export-extcommunity
  vpn-target 100:10 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ip address 10.1.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack0
 ip address 5.5.5.5 255.255.255.255
#
interface Nve1
 source 5.5.5.5
 vni 10 head-end peer-list protocol bgp
#
bgp 100 instance evpn1
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack0
 #
 l2vpn-family evpn
  policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 advertise irb
#
ospf 1
 area 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.10.0 0.0.0.255
#
return
```

- Leaf2的配置文件

```
#
sysname Leaf2
#
evpn-overlay enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.20.2 255.255.255.0
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.50.2 255.255.255.0
#
interface LoopBack0
 ip address 6.6.6.6 255.255.255.255
#
bgp 20
 peer 192.168.50.1 as-number 10
 #
 ipv4-family unicast
  network 5.5.5.5 255.255.255.255
```

```
 network 6.6.6.6 255.255.255.255
 peer 192.168.50.1 enable
#
bgp 100 instance evpn1
 peer 5.5.5.5 as-number 100
 peer 5.5.5.5 connect-interface LoopBack0
 peer 7.7.7.7 as-number 200
 peer 7.7.7.7 connect-interface LoopBack0
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 5.5.5.5 enable
  peer 5.5.5.5 advertise irb
  peer 5.5.5.5 next-hop-invariable
  peer 7.7.7.7 enable
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 next-hop-invariable
#
ospf 1
 import-route bgp
 area 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.20.0 0.0.0.255
#
return
```

- Spine2的配置文件

```
#
sysname Spine2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.40.1 255.255.255.0
#
interface LoopBack0
 ip address 4.4.4.4 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 192.168.30.0 0.0.0.255
  network 192.168.40.0 0.0.0.255
#
return
```

- Leaf3的配置文件

```
#
sysname Leaf3
#
evpn-overlay enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.2 255.255.255.0
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.60.2 255.255.255.0
#
interface LoopBack0
 ip address 7.7.7.7 255.255.255.255
#
bgp 30
 peer 192.168.60.1 as-number 10
 #
 ipv4-family unicast
```



```
 network 7.7.7.7 255.255.255.255
 network 8.8.8.8 255.255.255.255
 peer 192.168.60.1 enable
#
bgp 200 instance evpn1
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack0
 peer 8.8.8.8 as-number 200
 peer 8.8.8.8 connect-interface LoopBack0
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 advertise irb
  peer 6.6.6.6 next-hop-invariable
  peer 8.8.8.8 enable
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 next-hop-invariable
#
ospf 1
 import-route bgp
 area 0.0.0.0
  network 7.7.7.7 0.0.0.0
  network 192.168.30.0 0.0.0.255
#
return
```

- Leaf4的配置文件

```
#
sysname Leaf4
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:4
  vpn-target 100:5010 export-extcommunity evpn
  vpn-target 100:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 10:4
  vpn-target 100:20 export-extcommunity
  vpn-target 100:5010 export-extcommunity
  vpn-target 100:20 import-extcommunity
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ip address 10.2.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.40.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface LoopBack0
 ip address 8.8.8.8 255.255.255.255
#
interface Nve1
 source 8.8.8.8
 vni 20 head-end peer-list protocol bgp
#
bgp 200 instance evpn1
```

```
 peer 7.7.7.7 as-number 200
 peer 7.7.7.7 connect-interface LoopBack0
 #
 l2vpn-family evpn
  policy vpn-target
  peer 7.7.7.7 enable
  peer 7.7.7.7 advertise irb
#
ospf 1
 area 0.0.0.0
  network 8.8.8.8 0.0.0.0
  network 192.168.40.0 0.0.0.255
#
return
```

- Device1的配置文件

```
#
sysname Device1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.50.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
#
interface LoopBack0
 ip address 1.1.1.1 255.255.255.255
#
bgp 10
 peer 192.168.1.2 as-number 10
 peer 192.168.50.2 as-number 20
 #
 ipv4-family unicast
  peer 192.168.1.2 enable
  peer 192.168.1.2 next-hop-local
  peer 192.168.50.2 enable
#
return
```

- Device2的配置文件

```
#
sysname Device2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.60.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
#
interface LoopBack0
 ip address 2.2.2.2 255.255.255.255
#
bgp 10
 peer 192.168.1.1 as-number 10
 peer 192.168.60.2 as-number 30
 #
 ipv4-family unicast
  peer 192.168.1.1 enable
  peer 192.168.1.1 next-hop-local
  peer 192.168.60.2 enable
#
return
```

2.8.1.2 举例：配置通过 VLAN hand-off 实现 DCI 互联示例

组网需求

如图2-22所示，分别在数据中心A、数据中心B内配置BGP EVPN协议创建VXLAN隧道，实现各数据中心内部VM之间的通信，Leaf2和Leaf3通过二层子接口方式接入DCI-VTEP1和DCI-VTEP2，DCI-VTEP1和DCI-VTEP2之间配置EVPN协议创建VXLAN隧道，实现数据中心之间的通信。Leaf2/Leaf3将收到的数据中心侧的VXLAN报文进行解封装，然后发送到DCI-VTEP，DCI-VTEP将收到的VLAN报文重新封装成VXLAN报文后发送给对端DCI-VTEP，实现VXLAN隧道对跨数据中心的报文端到端的承载，保证跨数据中心VM之间的通信。

图 2-22 配置通过 VLAN hand-off 实现 DCI 互联组网图

说明

本例中interface1、interface2和interface3分别代表100GE1/0/1、100GE1/0/2、100GE1/0/3。

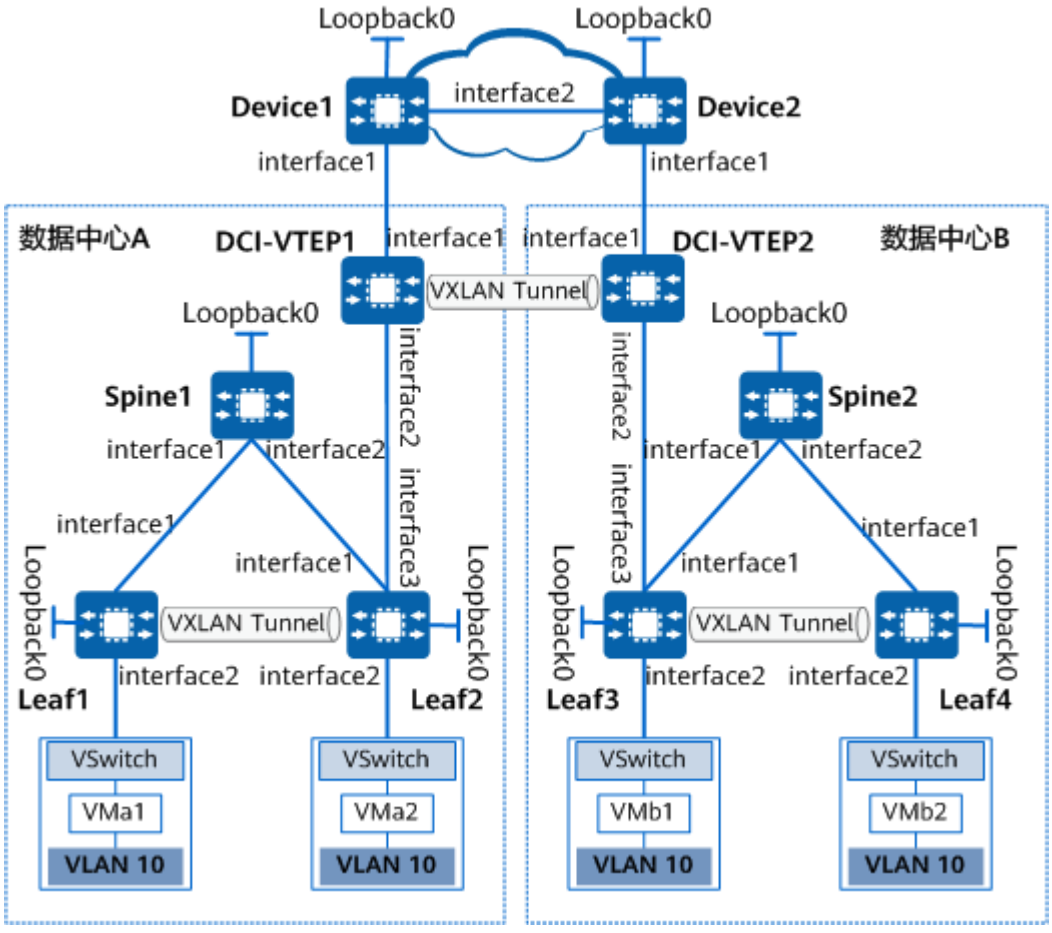


表 2-2 接口的 IP 地址

设备	接口	IP地址	设备	接口	IP地址
Device 1	100GE1/0/1	192.168.50.1/24	Device 2	100GE1/0/1	192.168.60.1/24

设备	接口	IP地址	设备	接口	IP地址
	100GE1/0/2	192.168.1.1/24		100GE1/0/2	192.168.1.2/24
	LoopBack0	1.1.1.1/32		LoopBack0	2.2.2.2/32
DCI-VTEP1	10GE1/0/1	192.168.50.2/24	DCI-VTEP2	10GE1/0/1	192.168.60.2/24
	10GE1/0/2	-		10GE1/0/2	-
	LoopBack0	9.9.9.9/32		LoopBack0	10.10.10.10/32
Spine1	100GE1/0/1	192.168.10.1/24	Spine2	100GE1/0/1	192.168.30.1/24
	100GE1/0/2	192.168.20.1/24		100GE1/0/2	192.168.40.1/24
	LoopBack0	3.3.3.3/32		LoopBack0	4.4.4.4/32
Leaf1	100GE1/0/1	192.168.10.2/24	Leaf4	100GE1/0/1	192.168.40.2/24
	100GE1/0/2	-		100GE1/0/2	-
	LoopBack0	5.5.5.5/32		LoopBack0	8.8.8.8/32
Leaf2	100GE1/0/1	192.168.20.2/24	Leaf3	100GE1/0/1	192.168.30.2/24
	100GE1/0/2	-		100GE1/0/2	-
	100GE1/0/3	-		100GE1/0/3	-
	LoopBack0	6.6.6.6/32		LoopBack0	7.7.7.7/32

配置思路

采用如下的思路配置通过VLAN hand-off实现DCI互联：

1. 配置各节点接口的IP地址。
2. 配置路由协议，实现各节点之间的互通。
3. 在数据中心A和数据中心B内配置BGP EVPN协议创建VXLAN隧道；在数据中心A和数据中心B内分别创建IBGP邻居。
4. 在DCI-VTEP之间配置BGP EVPN协议创建VXLAN隧道。
5. 在Leaf2、Leaf3、DCI-VTEP1和DCI-VTEP2上配置通过二层子接口方式接入DCI隧道。

操作步骤

步骤1 配置各节点接口的IP地址。

配置Device1。其他设备的配置过程与Device1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Device1
[*HUAWEI] commit
[~Device1] interface loopback 0
[*Device1-LoopBack0] ip address 1.1.1.1 32
[*Device1-LoopBack0] quit
[*Device1] interface 100ge 1/0/1
[*Device1-100GE1/0/1] undo portswitch
[*Device1-100GE1/0/1] ip address 192.168.50.1 24
[*Device1-100GE1/0/1] quit
[*Device1] interface 100ge 1/0/2
[*Device1-100GE1/0/2] undo portswitch
[*Device1-100GE1/0/2] ip address 192.168.1.1 24
[*Device1-100GE1/0/2] quit
[*Device1] commit
```

步骤2 配置路由协议，实现路由互通。

配置Spine1。Spine2、Device1、Device2的配置过程与Spine1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Spine1
[*HUAWEI] commit
[~Spine1] ospf 1
[*Spine1-ospf-1] area 0
[*Spine1-ospf-1-area-0.0.0.0] network 3.3.3.3 0.0.0.0
[*Spine1-ospf-1-area-0.0.0.0] network 192.168.10.0 0.0.0.255
[*Spine1-ospf-1-area-0.0.0.0] network 192.168.20.0 0.0.0.255
[*Spine1-ospf-1-area-0.0.0.0] quit
[*Spine1-ospf-1] quit
[*Spine1] commit
```

配置Leaf1。Leaf2、Leaf3、Leaf4、DCI-VTEP1、DCI-VTEP2的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf1
[*HUAWEI] commit
[~Leaf1] ospf 1
[*Leaf1-ospf-1] area 0
[*Leaf1-ospf-1-area-0.0.0.0] network 5.5.5.5 0.0.0.0
[*Leaf1-ospf-1-area-0.0.0.0] network 192.168.10.0 0.0.0.255
[*Leaf1-ospf-1-area-0.0.0.0] quit
[*Leaf1-ospf-1] quit
[*Leaf1] bgp 100
[*Leaf1-bgp] ipv4-family unicast
[*Leaf1-bgp-af-ipv4] peer 6.6.6.6 enable
[*Leaf1-bgp-af-ipv4] quit
[*Leaf1-bgp] quit
[*Leaf1] commit
```

步骤3 在数据中心A和数据中心B内配置BGP EVPN协议，创建VXLAN隧道。

1. 配置VXLAN业务接入点。

   # 配置Leaf1。

   ```
   [~Leaf1] bridge-domain 10
   [*Leaf1-bd10] quit
   [*Leaf1] interface 100GE 1/0/2.1 mode l2
   [*Leaf1-100GE1/0/2.1] encapsulation dot1q vid 10
   [*Leaf1-100GE1/0/2.1] bridge-domain 10
   [*Leaf1-100GE1/0/2.1] quit
   [*Leaf1] commit
   ```

Leaf2、Leaf3、Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

2. 在Leaf1、Leaf2、Leaf3和Leaf4上使能EVPN作为VXLAN控制平面。

   # 配置Leaf1。

   ```
   [~Leaf1] evpn-overlay enable
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

3. 在Leaf1和Leaf2之间、在Leaf3和Leaf4之间配置IBGP EVPN对等体关系。

   # 配置Leaf1。

   ```
   [~Leaf1] bgp 100
   [*Leaf1-bgp] peer 6.6.6.6 as-number 100
   [*Leaf1-bgp] peer 6.6.6.6 connect-interface LoopBack 0
   [*Leaf1-bgp] l2vpn-family evpn
   [*Leaf1-bgp-af-evpn] peer 6.6.6.6 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf1-bgp-af-evpn] quit
   [*Leaf1-bgp] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

4. 配置EVPN实例。

   # 配置Leaf1。

   ```
   [*Leaf1] bridge-domain 10
   [*Leaf1-bd10] vxlan vni 10
   [*Leaf1-bd10] evpn
   [*Leaf1-bd10-evpn] route-distinguisher 10:1
   [*Leaf1-bd10-evpn] vpn-target 11:1
   [*Leaf1-bd10-evpn] quit
   [*Leaf1-bd10] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3、Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

5. 在Leaf上使能头端复制功能。

   # 配置Leaf1。

   ```
   [~Leaf1] interface nve 1
   [*Leaf1-Nve1] source 5.5.5.5
   [*Leaf1-Nve1] vni 10 head-end peer-list protocol bgp
   [*Leaf1-Nve1] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3、Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

6. 在Leaf1和Leaf2之间、Leaf3和Leaf4之间配置发布IRB类型的路由。

   # 配置Leaf1。

   ```
   [~Leaf1] bgp 100
   [~Leaf1-bgp] l2vpn-family evpn
   [~Leaf1-bgp-af-evpn] peer 6.6.6.6 advertise irb
   [*Leaf1-bgp-af-evpn] quit
   [*Leaf1-bgp] quit
   [*Leaf1] commit
   ```


Leaf2、Leaf3、Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

步骤4 在DCI-VTEP之间配置BGP EVPN协议，创建VXLAN隧道。

1. 在DCI-VTEP1和DCI-VTEP2上使能EVPN作VXLAN控制平面功能。

   # 配置DCI-VTEP1。

   ```
   <HUAWEI> system-view
   [~HUAWEI] sysname DCI-VTEP1
   [*HUAWEI] commit
   [~DCI-VTEP1] evpn-overlay enable
   [*DCI-VTEP1] commit
   ```

   DCI-VTEP2的配置过程与DCI-VTEP1类似，在此不再赘述，具体请参考配置脚本。

2. 在DCI-VTEP1和DCI-VTEP2之间配置EBGP EVPN对等体关系。

   # 配置DCI-VTEP1。

   ```
   [~DCI-VTEP1] bgp 100
   [*DCI-VTEP1-bgp] peer 10.10.10.10 as-number 200
   [*DCI-VTEP1-bgp] peer 10.10.10.10 connect-interface LoopBack 0
   [*DCI-VTEP1-bgp] peer 10.10.10.10 ebgp-max-hop 255
   [*DCI-VTEP1-bgp] l2vpn-family evpn
   [*DCI-VTEP1-bgp-af-evpn] peer 10.10.10.10 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*DCI-VTEP1-bgp-af-evpn] quit
   [*DCI-VTEP1-bgp] quit
   [*DCI-VTEP1] commit
   ```

   DCI-VTEP2的配置过程与DCI-VTEP1类似，在此不再赘述，具体请参考配置脚本。

3. 在DCI-VTEP上配置EVPN实例

   # 配置DCI-VTEP1。

   ```
   [~DCI-VTEP1] bridge-domain 10
   [*DCI-VTEP1-bd10] vxlan vni 10
   [*DCI-VTEP1-bd10] evpn
   [*DCI-VTEP1-bd10-evpn] route-distinguisher 10:5
   [*DCI-VTEP1-bd10-evpn] vpn-target 33:3
   [*DCI-VTEP1-bd10-evpn] quit
   [*DCI-VTEP1-bd10] quit
   [*DCI-VTEP1] commit
   ```

   DCI-VTEP2的配置过程与DCI-VTEP1类似，在此不再赘述，具体请参考配置脚本。

4. 在DCI-VTEP上使能头端复制功能

   # 配置DCI-VTEP1。

   ```
   [~DCI-VTEP1] interface nve 1
   [*DCI-VTEP1-Nve1] source 9.9.9.9
   [*DCI-VTEP1-Nve1] vni 10 head-end peer-list protocol bgp
   [*DCI-VTEP1-Nve1] quit
   [*DCI-VTEP1] commit
   ```

   DCI-VTEP2的配置过程与DCI-VTEP1类似，在此不再赘述，具体请参考配置脚本。

5. 在DCI-VTEP1和DCI-VTEP2之间配置发布IRB类型的路由

   # 配置DCI-VTEP1。

   ```
   [~DCI-VTEP1] bgp 100
   [~DCI-VTEP1-bgp] l2vpn-family evpn
   [~DCI-VTEP1-bgp-af-evpn] peer 10.10.10.10 advertise irb
   [*DCI-VTEP1-bgp-af-evpn] quit
   [*DCI-VTEP1-bgp] quit
   [*DCI-VTEP1] commit
   ```

DCI-VTEP2的配置过程与DCI-VTEP1类似，在此不再赘述，具体请参考配置脚本。

步骤5 配置VLAN接入VXLAN隧道

配置Leaf2。

```
[~Leaf2] interface 10ge 1/0/3.1 mode l2
[*Leaf2-100GE1/0/3.1] encapsulation dot1q vid 10
[*Leaf2-10GE1/0/3.1] bridge-domain 10
[*Leaf2-100GE1/0/3.1] quit
[*Leaf2] commit
```

Leaf3、DCI-VTEP1、DCI-VTEP2的配置过程与Leaf2类似，在此不再赘述，具体请参考配置脚本。

----结束

检查配置结果

上述配置成功后，在Leaf上执行**display vxlan tunnel**命令，可以看到建立的VXLAN隧道信息。以Leaf1的显示为例：

```
[~Leaf1] display vxlan tunnel
Number of vxlan tunnel : 1
Tunnel ID   Source              Destination         State  Type     Uptime
--------------------------------------------------------------------------------
4026531842  5.5.5.5             8.8.8.8             up     dynamic  00:10:16
```

配置完成后，VMa1和VMb2之间可以互相通信。

配置脚本

- Spine1的配置文件

```
#
sysname Spine1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.20.1 255.255.255.0
#
interface LoopBack0
 ip address 3.3.3.3 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 192.168.10.0 0.0.0.255
  network 192.168.20.0 0.0.0.255
#
return
```

- Leaf1的配置文件

```
#
sysname Leaf1
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:1
  vpn-target 11:1 export-extcommunity
```

```
 vpn-target 11:1 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack0
 ip address 5.5.5.5 255.255.255.255
#
interface Nve1
 source 5.5.5.5
 vni 10 head-end peer-list protocol bgp
#
bgp 100
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 6.6.6.6 enable
 #
 l2vpn-family evpn
  policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 advertise irb
#
ospf 1
 area 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.10.0 0.0.0.255
#
return
```

- Leaf2的配置文件

```
#
sysname Leaf2
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:2
  vpn-target 11:1 export-extcommunity
  vpn-target 11:1 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.20.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface 100GE1/0/3.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack0
 ip address 6.6.6.6 255.255.255.255
#
interface Nve1
 source 6.6.6.6
 vni 10 head-end peer-list protocol bgp
#
bgp 100
 peer 5.5.5.5 as-number 100
```

```
 peer 5.5.5.5 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 5.5.5.5 enable
#
 l2vpn-family evpn
  policy vpn-target
  peer 5.5.5.5 enable
  peer 5.5.5.5 advertise irb
#
ospf 1
 area 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.20.0 0.0.0.255
#
return
```

- Spine2的配置文件

```
#
sysname Spine2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.40.1 255.255.255.0
#
interface LoopBack0
 ip address 4.4.4.4 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 192.168.30.0 0.0.0.255
  network 192.168.40.0 0.0.0.255
#
return
```

- Leaf3的配置文件

```
#
sysname Leaf3
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:3
  vpn-target 22:2 export-extcommunity
  vpn-target 22:2 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface 100GE1/0/3.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack0
 ip address 7.7.7.7 255.255.255.255
#
interface Nve1
 source 7.7.7.7
```

```
 vni 10 head-end peer-list protocol bgp
#
bgp 200
 peer 8.8.8.8 as-number 200
 peer 8.8.8.8 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 8.8.8.8 enable
#
 l2vpn-family evpn
  policy vpn-target
  peer 8.8.8.8 enable
  peer 8.8.8.8 advertise irb
#
ospf 1
  area 0.0.0.0
  network 7.7.7.7 0.0.0.0
  network 192.168.30.0 0.0.0.255
#
return
```

- Leaf4的配置文件

```
#
sysname Leaf4
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:4
  vpn-target 22:2 export-extcommunity
  vpn-target 22:2 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.40.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack0
 ip address 8.8.8.8 255.255.255.255
#
interface Nve1
 source 8.8.8.8
 vni 10 head-end peer-list protocol bgp
#
bgp 200
 peer 7.7.7.7 as-number 200
 peer 7.7.7.7 connect-interface LoopBack0
 #
 ipv4-family unicast
  peer 7.7.7.7 enable
 #
 l2vpn-family evpn
  policy vpn-target
  peer 7.7.7.7 enable
  peer 7.7.7.7 advertise irb
#
ospf 1
 area 0.0.0.0
  network 8.8.8.8 0.0.0.0
  network 192.168.40.0 0.0.0.255
#
return
```

- DCI-VTEP1的配置文件

```
#
sysname DCI-VTEP1
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:5
  vpn-target 33:3 export-extcommunity
  vpn-target 33:3 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.50.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack0
 ip address 9.9.9.9 255.255.255.255
#
interface Nve1
 source 9.9.9.9
 vni 10 head-end peer-list protocol bgp
#
bgp 100
 peer 10.10.10.10 as-number 200
 peer 10.10.10.10 connect-interface LoopBack0
 peer 10.10.10.10 ebgp-max-hop 255
 #
 ipv4-family unicast
  peer 10.10.10.10 enable
 #
 l2vpn-family evpn
  policy vpn-target
  peer 10.10.10.10 enable
  peer 10.10.10.10 advertise irb
#
ospf 1
 area 0.0.0.0
  network 9.9.9.9 0.0.0.0
  network 192.168.50.0 0.0.0.255
#
return
```

- DCI-VTEP2的配置文件

```
#
sysname DCI-VTEP2
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 10
  evpn
  route-distinguisher 11:6
  vpn-target 33:3 export-extcommunity
  vpn-target 33:3 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.60.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack0
 ip address 10.10.10.10 255.255.255.255
```



```
#
interface Nve1
 source 10.10.10.10
 vni 10 head-end peer-list protocol bgp
#
bgp 200
 peer 9.9.9.9 as-number 100
 peer 9.9.9.9 connect-interface LoopBack0
 peer 9.9.9.9 ebgp-max-hop 255
 #
 ipv4-family unicast
  peer 9.9.9.9 enable
 #
 l2vpn-family evpn
  policy vpn-target
  peer 9.9.9.9 enable
  peer 9.9.9.9 advertise irb
#
ospf 1
 area 0.0.0.0
  network 10.10.10.10 0.0.0.0
  network 192.168.60.0 0.0.0.255
#
return
```

- Device1的配置文件

```
#
sysname Device1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.50.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
#
interface LoopBack0
 ip address 1.1.1.1 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 1.1.1.1 0.0.0.0
  network 192.168.1.0 0.0.0.255
  network 192.168.50.0 0.0.0.255
#
return
```

- Device2的配置文件

```
#
sysname Device2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.60.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
#
interface LoopBack0
 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 192.168.1.0 0.0.0.255
  network 192.168.60.0 0.0.0.255
```

```
#
return
```

2.8.1.3 举例：配置 AS 域内的 Segment VXLAN 实现三层互通

组网需求

如图2-23所示，数据中心A和数据中心B规划在相同的BGP AS域，在数据中心内部配置BGP EVPN协议创建分布式网关VXLAN隧道，实现同一数据中心VMa1和VMa2之间、VMb1和VMb2之间的互相通信，通过在Leaf2和Leaf3之间配置BGP EVPN协议创建VXLAN隧道，实现数据中心A和数据中心B之间的互相通信（例如VMa1和VMb2之间互相通信）。由于Leaf2或者Leaf3收到IBGP EVPN对等体发送的EVPN路由后，不会再向其他IBGP EVPN对等体发送，因此需要将Leaf2和Leaf3配置为路由反射器。

图 2-23 配置 AS 域内的 Segment VXLAN 实现三层互通示意图

说明

本例中interface1、interface2和interface3分别代表100GE1/0/1、100GE1/0/2、100GE1/0/3。

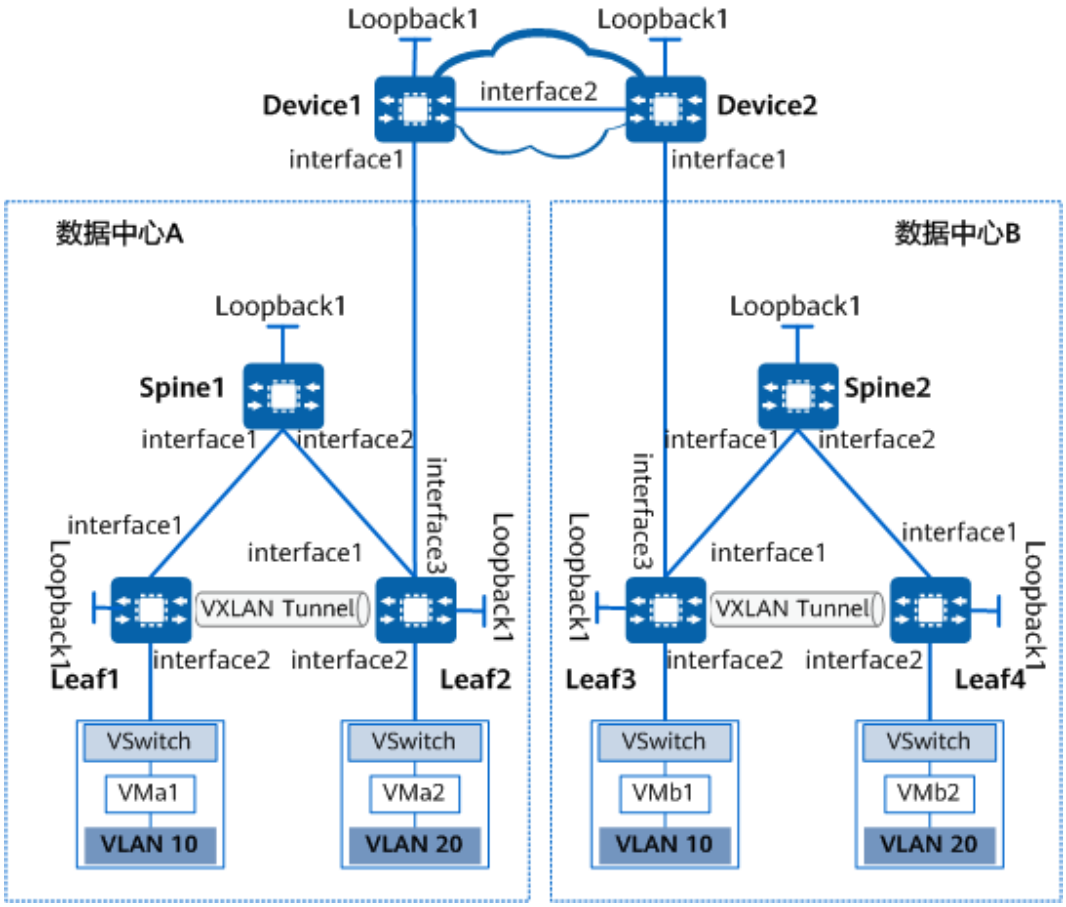


表 2-3 接口的 IP 地址

设备	接口	IP地址	设备	接口	IP地址
Device 1	100GE1/0/1	192.168.50.1/24	Device 2	100GE1/0/1	192.168.60.1/24
	100GE1/0/2	192.168.1.1/24		100GE1/0/2	192.168.1.2/24
	LoopBack1	1.1.1.1/32		LoopBack1	2.2.2.2/32
Spine1	100GE1/0/1	192.168.10.1/24	Spine2	100GE1/0/1	192.168.30.1/24
	100GE1/0/2	192.168.20.1/24		100GE1/0/2	192.168.40.1/24
	LoopBack1	3.3.3.3/32		LoopBack1	4.4.4.4/32
Leaf1	100GE1/0/1	192.168.10.2/24	Leaf4	100GE1/0/1	192.168.40.2/24
	100GE1/0/2	-		100GE1/0/2	-
	LoopBack1	5.5.5.5/32		LoopBack1	8.8.8.8/32
Leaf2	100GE1/0/1	192.168.20.2/24	Leaf3	100GE1/0/1	192.168.30.2/24
	100GE1/0/2	-		100GE1/0/2	-
	100GE1/0/3	192.168.50.2/24		100GE1/0/3	192.168.60.2/24
	LoopBack1	6.6.6.6/32		LoopBack1	7.7.7.7/32

配置思路

采用如下的思路配置AS域内的Segment VXLAN实现三层互通：

1. 配置各节点接口的IP地址。
2. 配置路由协议，实现各节点之间的互通。
3. 在数据中心A和数据中心B内配置BGP EVPN方式建立VXLAN隧道，实现同一数据中心内的互通。
4. 在Leaf2和Leaf3上配置BGP EVPN协议创建VXLAN隧道，实现数据中心A和数据中心B之间的互通。
5. 配置Leaf2和Leaf3为路由反射器。
6. 在各Leaf上配置路由策略并应用。

操作步骤

步骤1 配置各节点接口的IP地址。

配置Device1。其他设备的配置过程与Device1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Device1
[*HUAWEI] commit
[~Device1] interface loopback 1
[*Device1-LoopBack0] ip address 1.1.1.1 32
[*Device1-LoopBack0] quit
[*Device1] interface 100ge 1/0/1
[*Device1-100GE1/0/1] undo portswitch
[*Device1-100GE1/0/1] ip address 192.168.50.1 24
[*Device1-100GE1/0/1] quit
[*Device1] interface 100ge 1/0/2
[*Device1-100GE1/0/2] undo portswitch
[*Device1-100GE1/0/2] ip address 192.168.1.1 24
[*Device1-100GE1/0/2] quit
[*Device1] commit
```

步骤2 配置路由协议，实现各节点之间的互通。

配置Spine1。Spine2、Leaf1、Leaf4的配置过程与Spine1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Spine1
[*HUAWEI] commit
[~Spine1] ospf 1
[*Spine1-ospf-1] area 0
[*Spine1-ospf-1-area-0.0.0.0] network 3.3.3.3 0.0.0.0
[*Spine1-ospf-1-area-0.0.0.0] network 192.168.10.0 0.0.0.255
[*Spine1-ospf-1-area-0.0.0.0] network 192.168.20.0 0.0.0.255
[*Spine1-ospf-1-area-0.0.0.0] quit
[*Spine1-ospf-1] quit
[*Spine1] commit
```

配置Leaf2。Leaf3、Device1、Device2的配置过程与Leaf2类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf2
[*HUAWEI] commit
[~Leaf2] ospf 1
[*Leaf2-ospf-1] area 0
[*Leaf2-ospf-1-area-0.0.0.0] network 6.6.6.6 0.0.0.0
[*Leaf2-ospf-1-area-0.0.0.0] network 192.168.20.0 0.0.0.255
[*Leaf2-ospf-1-area-0.0.0.0] quit
[*Leaf2-ospf-1] quit
[*Leaf2] bgp 20
[*Leaf2-bgp] peer 192.168.50.1 as-number 10
[*Leaf2-bgp] ipv4-family unicast
[*Leaf2-bgp-af-ipv4] network 6.6.6.6 255.255.255.255
[*Leaf2-bgp-af-ipv4] peer 192.168.50.1 enable
[*Leaf2-bgp-af-ipv4] quit
[*Leaf2-bgp] quit
[*Leaf2] commit
```

步骤3 数据中心A和数据中心B内配置BGP EVPN方式建立VXLAN隧道。

1. 配置VXLAN业务接入点。

   # 配置Leaf1。

   ```
   [~Leaf1] bridge-domain 10
   [*Leaf1-bd10] quit
   [*Leaf1] interface 100GE 1/0/2.1 mode l2
   [*Leaf1-100GE1/0/2.1] encapsulation dot1q vid 10
   [*Leaf1-100GE1/0/2.1] bridge-domain 10
   ```

```
[*Leaf1-100GE1/0/2.1] quit
[*Leaf1] commit
```

Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

2. 在Leaf上使能EVPN作为VXLAN控制平面。

   # 配置Leaf1。

   ```
   [~Leaf1] evpn-overlay enable
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

3. 在Leaf1和Leaf2之间、Leaf3和Leaf4之间配置BGP EVPN对等体关系。

   # 配置Leaf1。

   ```
   [~Leaf1] bgp 100 instance evpn1
   [*Leaf1-bgp-instance-evpn1] peer 6.6.6.6 as-number 100
   [*Leaf1-bgp-instance-evpn1] peer 6.6.6.6 connect-interface LoopBack 1
   [*Leaf1-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf1-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf1-bgp-instance-evpn1-af-evpn] quit
   [*Leaf1-bgp-instance-evpn1] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

4. 在Leaf1上配置EVPN实例。

   # 配置Leaf1。

   ```
   [~Leaf1] ip vpn-instance vpn1
   [*Leaf1-vpn-instance-vpn1] vxlan vni 5010
   [*Leaf1-vpn-instance-vpn1] ipv4-family
   [*Leaf1-vpn-instance-vpn1-af-ipv4] route-distinguisher 20:1
   [*Leaf1-vpn-instance-vpn1-af-ipv4] vpn-target 100:5010 evpn
   [*Leaf1-vpn-instance-vpn1-af-ipv4] quit
   [*Leaf1-vpn-instance-vpn1] quit
   [*Leaf1] bridge-domain 10
   [*Leaf1-bd10] vxlan vni 10
   [*Leaf1-bd10] evpn
   [*Leaf1-bd10-evpn] route-distinguisher 10:1
   [*Leaf1-bd10-evpn] vpn-target 100:10
   [*Leaf1-bd10-evpn] vpn-target 100:5010 export-extcommunity
   [*Leaf1-bd10-evpn] quit
   [*Leaf1-bd10] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

5. 在各Leaf上使能头端复制功能。

   # 在配置Leaf1。

   ```
   [~Leaf1] interface nve 1
   [*Leaf1-Nve1] source 5.5.5.5
   [*Leaf1-Nve1] vni 10 head-end peer-list protocol bgp
   [*Leaf1-Nve1] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

6. 在Leaf上配置VXLAN三层网关。

   # 配置Leaf1。

   ```
   [~Leaf1] interface vbdif 10
   [*Leaf1-Vbdif10] ip binding vpn-instance vpn1
   [*Leaf1-Vbdif10] ip address 10.10.1.1 24
   [*Leaf1-Vbdif10] arp collect host enable
   [*Leaf1-Vbdif10] vxlan anycast-gateway enable
   [*Leaf1-Vbdif10] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

7. 配置VXLAN网关之间发布的路由类型。

   # 配置Leaf1。

   ```
   [~Leaf1] bgp 100 instance evpn1
   [*Leaf1-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf1-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise irb
   [*Leaf1-bgp-instance-evpn1-af-evpn] quit
   [*Leaf1-bgp-instance-evpn1] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

步骤4 在Leaf2和Leaf3上配置BGP EVPN协议创建VXLAN隧道。

1. 在Leaf上配置BGP EVPN对等体关系。

   # 配置Leaf2。

   ```
   [~Leaf2] bgp 100 instance evpn1
   [*Leaf2-bgp-instance-evpn1] peer 7.7.7.7 as-number 100
   [*Leaf2-bgp-instance-evpn1] peer 7.7.7.7 connect-interface LoopBack1
   [*Leaf2-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 advertise irb
   [*Leaf2-bgp-instance-evpn1-af-evpn] quit
   [*Leaf2-bgp-instance-evpn1] quit
   [*Leaf2] commit
   ```

   # 配置Leaf3。

   ```
   [~Leaf3] bgp 100 instance evpn1
   [*Leaf3-bgp-instance-evpn1] peer 6.6.6.6 as-number 100
   [*Leaf3-bgp-instance-evpn1] peer 6.6.6.6 connect-interface LoopBack1
   [*Leaf3-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise irb
   [*Leaf3-bgp-instance-evpn1-af-evpn] quit
   [*Leaf3-bgp-instance-evpn1] quit
   [*Leaf3] commit
   ```

2. 配置EVPN路由中的IRB路由、IP前缀路由的重生成功能。

   # 配置Leaf2。

   ```
   [~Leaf2] bgp 100 instance evpn1
   [~Leaf2-bgp-instance-evpn1] l2vpn-family evpn
   [~Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 import reoriginate
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 advertise route-reoriginated evpn mac-ip
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 advertise route-reoriginated evpn ip
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 import reoriginate
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 advertise route-reoriginated evpn mac-ip
   ```

```
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 advertise route-reoriginated evpn ip
[*Leaf2-bgp-instance-evpn1-af-evpn] quit
[*Leaf2-bgp-instance-evpn1] quit
[*Leaf2] commit
```

配置Leaf3。

```
[~Leaf3] bgp 100 instance evpn1
[*Leaf3-bgp-instance-evpn1] l2vpn-family evpn
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 import reoriginate
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 advertise route-reoriginated evpn mac-ip
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 advertise route-reoriginated evpn ip
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 import reoriginate
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise route-reoriginated evpn mac-ip
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise route-reoriginated evpn ip
[*Leaf3-bgp-instance-evpn1-af-evpn] quit
[*Leaf3-bgp-instance-evpn1] quit
[*Leaf3] commit
```

步骤5 配置Leaf2为路由反射器，指定Leaf1和Leaf3作为反射器的客户机；配置Leaf3为路由反射器，指定Leaf4和Leaf2作为反射器的客户机。

配置Leaf2。

```
[~Leaf2] bgp 100 instance evpn1
[~Leaf2-bgp-instance-evpn1] l2vpn-family evpn
[~Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 reflect-client
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 reflect-client
[*Leaf2-bgp-instance-evpn1-af-evpn] undo policy vpn-target
[*Leaf2-bgp-instance-evpn1-af-evpn] quit
[*Leaf2-bgp-instance-evpn1] quit
[*Leaf2] commit
```

配置Leaf3。

```
[~Leaf3] bgp 100 instance evpn1
[~Leaf3-bgp-instance-evpn1] l2vpn-family evpn
[~Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 reflect-client
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 reflect-client
[*Leaf3-bgp-instance-evpn1-af-evpn] undo policy vpn-target
[*Leaf3-bgp-instance-evpn1-af-evpn] quit
[*Leaf3-bgp-instance-evpn1] quit
[*Leaf3] commit
```

步骤6 在各Leaf上配置路由策略并应用。

配置Leaf1，只允许接收下一跳是本端DCI Leaf（即Leaf2）的路由并进行应用。Leaf4上的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] ip ip-prefix DCI index 10 permit 6.6.6.6 32
[*Leaf1] route-policy DCI permit node 10
[*Leaf1-route-policy] if-match ip next-hop ip-prefix DCI
[*Leaf1-route-policy] quit
[*Leaf1] bgp 100 instance evpn1
[*Leaf1-bgp-instance-evpn1] l2vpn-family evpn
[*Leaf1-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 route-policy DCI import
[*Leaf1-bgp-instance-evpn1-af-evpn] quit
[*Leaf1-bgp-instance-evpn1] quit
[*Leaf1] commit
```

配置Leaf2，只允许接收下一跳是对端DCI Leaf（即Leaf3）的路由并进行应用。Leaf3上的配置与Leaf2类似，这里不再赘述。

```
[~Leaf2] ip ip-prefix DCI index 10 permit 7.7.7.7 32
[*Leaf2] route-policy DCI permit node 10
[*Leaf2-route-policy] if-match ip next-hop ip-prefix DCI
[*Leaf2-route-policy] quit
[*Leaf2] bgp 100 instance evpn1
[*Leaf2-bgp-instance-evpn1] l2vpn-family evpn
```



```
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 route-policy DCI import
[*Leaf2-bgp-instance-evpn1-af-evpn] quit
[*Leaf2-bgp-instance-evpn1] quit
[*Leaf2] commit
```

----结束

检查配置结果

在Leaf上执行**display vxlan tunnel**命令，可查看到VXLAN隧道的信息。以Leaf2显示为例。

```
[~Leaf2] display vxlan tunnel
Number of vxlan tunnel : 2
Tunnel ID   Source              Destination         State  Type     Uptime
--------------------------------------------------------------------------------
4026531841  6.6.6.6             5.5.5.5             up     dynamic  0035h21m
4026531842  6.6.6.6             7.7.7.7             up     dynamic  0035h25m
```

配置完成后，VMa1和VMb2之间可以互相通信。

配置脚本

- Spine1的配置文件

```
#
sysname Spine1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.20.1 255.255.255.0
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 192.168.10.0 0.0.0.255
  network 192.168.20.0 0.0.0.255
#
return
```

- Leaf1的配置文件

```
#
sysname Leaf1
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:1
  vpn-target 100:5010 export-extcommunity evpn
  vpn-target 100:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:1
  vpn-target 100:10 export-extcommunity
  vpn-target 100:5010 export-extcommunity
  vpn-target 100:10 import-extcommunity
#
```



```
interface Vbdif10
 ip binding vpn-instance vpn1
 ip address 10.10.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack1
 ip address 5.5.5.5 255.255.255.255
#
interface Nve1
 source 5.5.5.5
 vni 10 head-end peer-list protocol bgp
#
bgp 100 instance evpn1
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack1
 #
 l2vpn-family evpn
  policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 route-policy DCI import
  peer 6.6.6.6 advertise irb
#
ospf 1
 area 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.10.0 0.0.0.255
#
route-policy DCI permit node 10
 if-match ip next-hop ip-prefix DCI
#
ip ip-prefix DCI index 10 permit 6.6.6.6 32
#
return
```

- Leaf2的配置文件

```
#
sysname Leaf2
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:2
  vpn-target 100:5010 export-extcommunity evpn
  vpn-target 300:5010 export-extcommunity evpn
  vpn-target 100:5010 import-extcommunity evpn
  vpn-target 300:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 10:2
  vpn-target 100:20 export-extcommunity
  vpn-target 100:5010 export-extcommunity
  vpn-target 300:5010 export-extcommunity
  vpn-target 100:20 import-extcommunity
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ip address 10.20.1.1 255.255.255.0
```

```
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.20.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.50.2 255.255.255.0
#
interface LoopBack1
 ip address 6.6.6.6 255.255.255.255
#
interface Nve1
 source 6.6.6.6
 vni 20 head-end peer-list protocol bgp
#
bgp 20
 peer 192.168.50.1 as-number 10
 #
 ipv4-family unicast
  network 6.6.6.6 255.255.255.255
  peer 192.168.50.1 enable
#
bgp 100 instance evpn1
 peer 5.5.5.5 as-number 100
 peer 5.5.5.5 connect-interface LoopBack1
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack1
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 5.5.5.5 enable
  peer 5.5.5.5 advertise irb
  peer 5.5.5.5 reflect-client
  peer 5.5.5.5 import reoriginate
  peer 5.5.5.5 advertise route-reoriginated evpn mac-ip
  peer 5.5.5.5 advertise route-reoriginated evpn ip
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy DCI import
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 reflect-client
  peer 7.7.7.7 import reoriginate
  peer 7.7.7.7 advertise route-reoriginated evpn mac-ip
  peer 7.7.7.7 advertise route-reoriginated evpn ip
#
ospf 1
 area 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.20.0 0.0.0.255
#
route-policy DCI permit node 10
 if-match ip next-hop ip-prefix DCI
#
ip ip-prefix DCI index 10 permit 7.7.7.7 32
#
return
```

- Spine2的配置文件

```
#
sysname Spine2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.1 255.255.255.0
```

```
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.40.1 255.255.255.0
#
interface LoopBack1
 ip address 4.4.4.4 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 192.168.30.0 0.0.0.255
  network 192.168.40.0 0.0.0.255
#
return
```

- Leaf3的配置文件

```
#
sysname Leaf3
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:3
  vpn-target 200:5010 export-extcommunity evpn
  vpn-target 300:5010 export-extcommunity evpn
  vpn-target 200:5010 import-extcommunity evpn
  vpn-target 300:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:3
  vpn-target 200:10 export-extcommunity
  vpn-target 200:5010 export-extcommunity
  vpn-target 300:5010 export-extcommunity
  vpn-target 200:10 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ip address 10.30.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.60.2 255.255.255.0
#
interface LoopBack1
 ip address 7.7.7.7 255.255.255.255
#
interface Nve1
 source 7.7.7.7
 vni 10 head-end peer-list protocol bgp
#
bgp 30
 peer 192.168.60.1 as-number 10
 #
 ipv4-family unicast
  network 7.7.7.7 255.255.255.255
```



```
 peer 192.168.60.1 enable
#
bgp 100 instance evpn1
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack1
 peer 8.8.8.8 as-number 100
 peer 8.8.8.8 connect-interface LoopBack1
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 route-policy DCI import
  peer 6.6.6.6 advertise irb
  peer 6.6.6.6 reflect-client
  peer 6.6.6.6 import reoriginate
  peer 6.6.6.6 advertise route-reoriginated evpn mac-ip
  peer 6.6.6.6 advertise route-reoriginated evpn ip
  peer 8.8.8.8 enable
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 reflect-client
  peer 8.8.8.8 import reoriginate
  peer 8.8.8.8 advertise route-reoriginated evpn mac-ip
  peer 8.8.8.8 advertise route-reoriginated evpn ip
#
ospf 1
 area 0.0.0.0
  network 7.7.7.7 0.0.0.0
  network 192.168.30.0 0.0.0.255
#
route-policy DCI permit node 10
 if-match ip next-hop ip-prefix DCI
#
ip ip-prefix DCI index 10 permit 6.6.6.6 32
#
return
```

- Leaf4的配置文件

```
#
sysname Leaf4
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:4
  vpn-target 200:5010 export-extcommunity evpn
  vpn-target 200:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 10:4
  vpn-target 200:20 export-extcommunity
  vpn-target 200:5010 export-extcommunity
  vpn-target 200:20 import-extcommunity
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ip address 10.40.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.40.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
```



```
#
interface LoopBack1
 ip address 8.8.8.8 255.255.255.255
#
interface Nve1
 source 8.8.8.8
 vni 20 head-end peer-list protocol bgp
#
bgp 100 instance evpn1
 peer 7.7.7.7 as-number 100
 peer 7.7.7.7 connect-interface LoopBack1
 #
 l2vpn-family evpn
  policy vpn-target
  peer 7.7.7.7 enable
  peer 7.7.7.7 route-policy DCI import
  peer 7.7.7.7 advertise irb
#
ospf 1
 area 0.0.0.0
  network 8.8.8.8 0.0.0.0
  network 192.168.40.0 0.0.0.255
#
route-policy DCI permit node 10
 if-match ip next-hop ip-prefix DCI
#
ip ip-prefix DCI index 10 permit 7.7.7.7 32
#
return
```

- Device1的配置文件

```
#
sysname Device1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.50.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
#
interface LoopBack1
 ip address 1.1.1.1 255.255.255.255
#
bgp 10
 peer 192.168.1.2 as-number 10
 peer 192.168.50.2 as-number 20
 #
 ipv4-family unicast
  peer 192.168.1.2 enable
  peer 192.168.1.2 next-hop-local
  peer 192.168.50.2 enable
#
return
```

- Device2的配置文件

```
#
sysname Device2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.60.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
```



```
#
bgp 10
 peer 192.168.1.1 as-number 10
 peer 192.168.60.2 as-number 30
 #
 ipv4-family unicast
  peer 192.168.1.1 enable
  peer 192.168.1.1 next-hop-local
  peer 192.168.60.2 enable
#
return
```

2.8.1.4 举例：配置跨 AS 的 Segment VXLAN 实现三层互通

组网需求

如图2-24所示，数据中心A和数据中心B规划在不同的BGP AS域，在数据中心内部配置BGP EVPN协议创建分布式网关VXLAN隧道，实现同一数据中心VMa1和VMa2之间、VMb1和VMb2之间的互相通信，通过在Leaf2和Leaf3之间配置BGP EVPN协议创建VXLAN隧道，实现数据中心A和数据中心B之间的互相通信（例如VMa1和VMb2之间互相通信）。

图 2-24 配置跨 AS 的 Segment VXLAN 实现三层互通示意图

说明

本例中interface1、interface2和interface3分别代表100GE1/0/1、100GE1/0/2、100GE1/0/3。

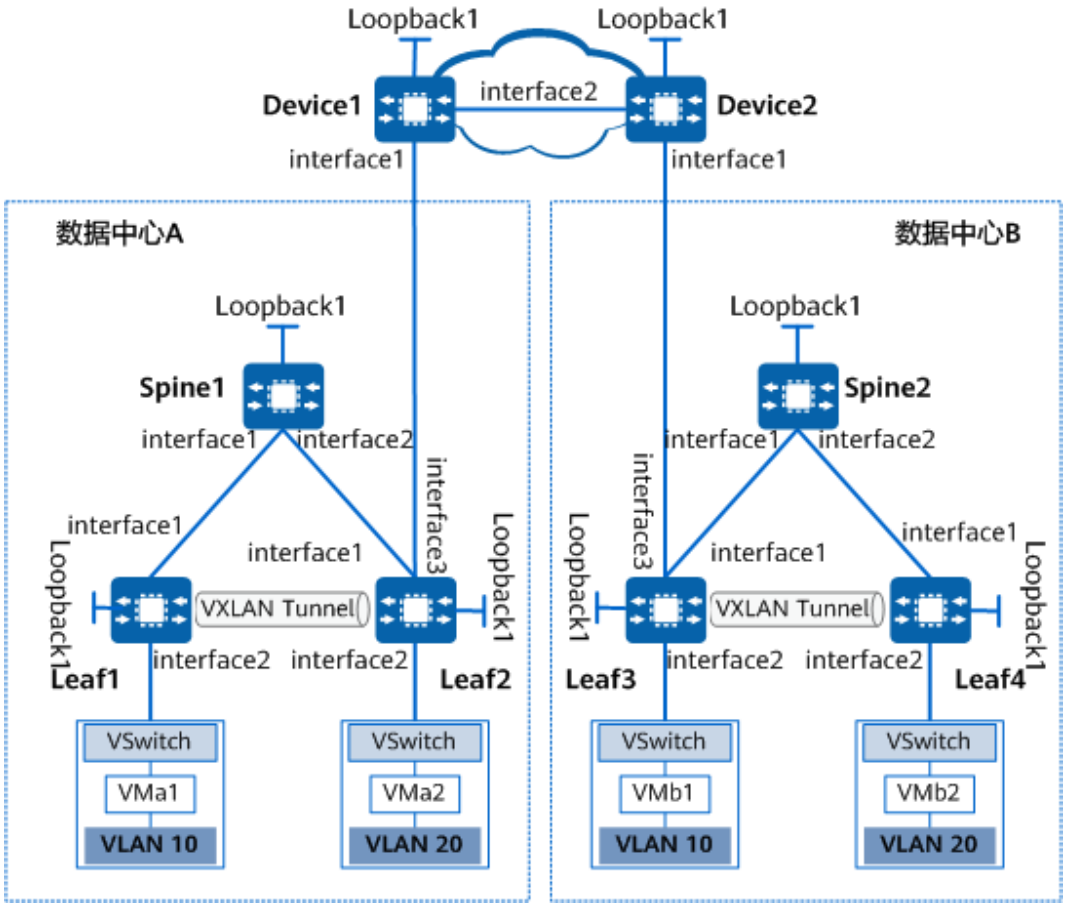


表 2-4 接口的 IP 地址

设备	接口	IP地址	设备	接口	IP地址
Device 1	100GE1/0/1	192.168.50.1/24	Device 2	100GE1/0/1	192.168.60.1/24
	100GE1/0/2	192.168.1.1/24		100GE1/0/2	192.168.1.2/24
	LoopBack1	1.1.1.1/32		LoopBack1	2.2.2.2/32
Spine1	100GE1/0/1	192.168.10.1/24	Spine2	100GE1/0/1	192.168.30.1/24
	100GE1/0/2	192.168.20.1/24		100GE1/0/2	192.168.40.1/24
	LoopBack1	3.3.3.3/32		LoopBack1	4.4.4.4/32
Leaf1	100GE1/0/1	192.168.10.2/24	Leaf4	100GE1/0/1	192.168.40.2/24
	100GE1/0/2	-		100GE1/0/2	-
	LoopBack1	5.5.5.5/32		LoopBack1	8.8.8.8/32
Leaf2	100GE1/0/1	192.168.20.2/24	Leaf3	100GE1/0/1	192.168.30.2/24
	100GE1/0/2	-		100GE1/0/2	-
	100GE1/0/3	192.168.50.2/24		100GE1/0/3	192.168.60.2/24
	LoopBack1	6.6.6.6/32		LoopBack1	7.7.7.7/32

配置思路

采用如下的思路配置跨AS的Segment VXLAN实现三层互通：

1. 配置各节点接口的IP地址。
2. 配置路由协议，实现各节点之间的互通。
3. 在数据中心A和数据中心B内配置BGP EVPN方式建立VXLAN隧道，实现同一数据中心内的互通。
4. 在Leaf2和Leaf3上配置BGP EVPN协议创建VXLAN隧道，实现数据中心A和数据中心B之间的互通。

操作步骤

步骤1 配置各节点接口的IP地址。

配置Device1。其他设备的配置过程与Device1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Device1
```

```
[*HUAWEI] commit
[~Device1] interface loopback 1
[*Device1-LoopBack0] ip address 1.1.1.1 32
[*Device1-LoopBack0] quit
[*Device1] interface 100ge 1/0/1
[*Device1-100GE1/0/1] undo portswitch
[*Device1-100GE1/0/1] ip address 192.168.50.1 24
[*Device1-100GE1/0/1] quit
[*Device1] interface 100ge 1/0/2
[*Device1-100GE1/0/2] undo portswitch
[*Device1-100GE1/0/2] ip address 192.168.1.1 24
[*Device1-100GE1/0/2] quit
[*Device1] commit
```

步骤2 配置路由协议，实现各节点之间的互通。

配置Spine1。Spine2、Leaf1、Leaf4的配置过程与Spine1类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Spine1
[*HUAWEI] commit
[~Spine1] ospf 1
[*Spine1-ospf-1] area 0
[*Spine1-ospf-1-area-0.0.0.0] network 3.3.3.3 0.0.0.0
[*Spine1-ospf-1-area-0.0.0.0] network 192.168.10.0 0.0.0.255
[*Spine1-ospf-1-area-0.0.0.0] network 192.168.20.0 0.0.0.255
[*Spine1-ospf-1-area-0.0.0.0] quit
[*Spine1-ospf-1] quit
[*Spine1] commit
```

配置Leaf2。Leaf3、Device1、Device2的配置过程与Leaf2类似，在此不再赘述，具体请参考配置脚本。

```
<HUAWEI> system-view
[~HUAWEI] sysname Leaf2
[*HUAWEI] commit
[~Leaf2] ospf 1
[*Leaf2-ospf-1] area 0
[*Leaf2-ospf-1-area-0.0.0.0] network 6.6.6.6 0.0.0.0
[*Leaf2-ospf-1-area-0.0.0.0] network 192.168.20.0 0.0.0.255
[*Leaf2-ospf-1-area-0.0.0.0] quit
[*Leaf2-ospf-1] quit
[*Leaf2] bgp 20
[*Leaf2-bgp] peer 192.168.50.1 as-number 10
[*Leaf2-bgp] ipv4-family unicast
[*Leaf2-bgp-af-ipv4] network 6.6.6.6 255.255.255.255
[*Leaf2-bgp-af-ipv4] peer 192.168.50.1 enable
[*Leaf2-bgp-af-ipv4] quit
[*Leaf2-bgp] quit
[*Leaf2] commit
```

步骤3 数据中心A和数据中心B内配置BGP EVPN方式建立VXLAN隧道。

1. 配置VXLAN业务接入点。

   # 配置Leaf1。

   ```
   [~Leaf1] bridge-domain 10
   [*Leaf1-bd10] quit
   [*Leaf1] interface 100GE 1/0/2.1 mode l2
   [*Leaf1-100GE1/0/2.1] encapsulation dot1q vid 10
   [*Leaf1-100GE1/0/2.1] bridge-domain 10
   [*Leaf1-100GE1/0/2.1] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

2. 在Leaf上使能EVPN作为VXLAN控制平面。

   # 配置Leaf1。

   ```
   [~Leaf1] evpn-overlay enable
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

3. 在Leaf1和Leaf2之间、Leaf3和Leaf4之间配置BGP EVPN对等体关系。

   # 配置Leaf1。

   ```
   [~Leaf1] bgp 100 instance evpn1
   [*Leaf1-bgp-instance-evpn1] peer 6.6.6.6 as-number 100
   [*Leaf1-bgp-instance-evpn1] peer 6.6.6.6 connect-interface LoopBack 1
   [*Leaf1-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf1-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf1-bgp-instance-evpn1-af-evpn] quit
   [*Leaf1-bgp-instance-evpn1] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

4. 在Leaf1上配置EVPN实例。

   # 配置Leaf1。

   ```
   [~Leaf1] ip vpn-instance vpn1
   [*Leaf1-vpn-instance-vpn1] vxlan vni 5010
   [*Leaf1-vpn-instance-vpn1] ipv4-family
   [*Leaf1-vpn-instance-vpn1-af-ipv4] route-distinguisher 20:1
   [*Leaf1-vpn-instance-vpn1-af-ipv4] vpn-target 100:5010 evpn
   [*Leaf1-vpn-instance-vpn1-af-ipv4] quit
   [*Leaf1-vpn-instance-vpn1] quit
   [*Leaf1] bridge-domain 10
   [*Leaf1-bd10] vxlan vni 10
   [*Leaf1-bd10] evpn
   [*Leaf1-bd10-evpn] route-distinguisher 10:1
   [*Leaf1-bd10-evpn] vpn-target 100:10
   [*Leaf1-bd10-evpn] vpn-target 100:5010 export-extcommunity
   [*Leaf1-bd10-evpn] quit
   [*Leaf1-bd10] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

5. 在各Leaf上使能头端复制功能。

   # 在配置Leaf1。

   ```
   [~Leaf1] interface nve 1
   [*Leaf1-Nve1] source 5.5.5.5
   [*Leaf1-Nve1] vni 10 head-end peer-list protocol bgp
   [*Leaf1-Nve1] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

6. 在Leaf上配置VXLAN三层网关。

   # 配置Leaf1。

   ```
   [~Leaf1] interface vbdif 10
   [*Leaf1-Vbdif10] ip binding vpn-instance vpn1
   ```

```
[*Leaf1-Vbdif10] ip address 10.10.1.1 24
[*Leaf1-Vbdif10] arp collect host enable
[*Leaf1-Vbdif10] vxlan anycast-gateway enable
[*Leaf1-Vbdif10] quit
[*Leaf1] commit
```

Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

7. 配置VXLAN网关之间发布的路由类型。

   # 配置Leaf1。

   ```
   [~Leaf1] bgp 100 instance evpn1
   [*Leaf1-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf1-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise irb
   [*Leaf1-bgp-instance-evpn1-af-evpn] quit
   [*Leaf1-bgp-instance-evpn1] quit
   [*Leaf1] commit
   ```

   Leaf2、Leaf3和Leaf4的配置过程与Leaf1类似，在此不再赘述，具体请参考配置脚本。

步骤4 在Leaf2和Leaf3上配置BGP EVPN协议创建VXLAN隧道。

1. 在Leaf上配置BGP EVPN对等体关系。

   # 配置Leaf2。

   ```
   [~Leaf2] bgp 100 instance evpn1
   [*Leaf2-bgp-instance-evpn1] peer 7.7.7.7 as-number 200
   [*Leaf2-bgp-instance-evpn1] peer 7.7.7.7 connect-interface LoopBack1
   [*Leaf2-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 advertise irb
   [*Leaf2-bgp-instance-evpn1-af-evpn] quit
   [*Leaf2-bgp-instance-evpn1] quit
   [*Leaf2] commit
   ```

   # 配置Leaf3。

   ```
   [~Leaf3] bgp 200 instance evpn1
   [*Leaf3-bgp-instance-evpn1] peer 6.6.6.6 as-number 100
   [*Leaf3-bgp-instance-evpn1] peer 6.6.6.6 connect-interface LoopBack1
   [*Leaf3-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise irb
   [*Leaf3-bgp-instance-evpn1-af-evpn] quit
   [*Leaf3-bgp-instance-evpn1] quit
   [*Leaf3] commit
   ```

2. 配置EVPN路由中的IRB路由、IP前缀路由的重生成功能。

   # 配置Leaf2。

   ```
   [~Leaf2] bgp 100 instance evpn1
   [~Leaf2-bgp-instance-evpn1] l2vpn-family evpn
   [~Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 import reoriginate
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 advertise route-reoriginated evpn mac-ip
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 5.5.5.5 advertise route-reoriginated evpn ip
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 import reoriginate
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 advertise route-reoriginated evpn mac-ip
   [*Leaf2-bgp-instance-evpn1-af-evpn] peer 7.7.7.7 advertise route-reoriginated evpn ip
   [*Leaf2-bgp-instance-evpn1-af-evpn] quit
   [*Leaf2-bgp-instance-evpn1] quit
   [*Leaf2] commit
   ```

   # 配置Leaf3。


```
[~Leaf3] bgp 200 instance evpn1
[*Leaf3-bgp-instance-evpn1] l2vpn-family evpn
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 import reoriginate
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 advertise route-reoriginated evpn mac-ip
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 8.8.8.8 advertise route-reoriginated evpn ip
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 import reoriginate
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise route-reoriginated evpn mac-ip
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 6.6.6.6 advertise route-reoriginated evpn ip
[*Leaf3-bgp-instance-evpn1-af-evpn] quit
[*Leaf3-bgp-instance-evpn1] quit
[*Leaf3] commit
```

----结束

检查配置结果

在Leaf上执行**display vxlan tunnel**命令，可查看到VXLAN隧道的信息。以Leaf2显示为例。

```
[~Leaf2] display vxlan tunnel
Number of vxlan tunnel : 2
Tunnel ID   Source            Destination       State  Type     Uptime
--------------------------------------------------------------------------------
4026531841  6.6.6.6           5.5.5.5           up     dynamic  0035h21m
4026531842  6.6.6.6           7.7.7.7           up     dynamic  0035h25m
```

配置完成后，VMa1和VMb2之间可以互相通信。

配置脚本

- Spine1的配置文件

```
#
sysname Spine1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.20.1 255.255.255.0
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.255
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 192.168.10.0 0.0.0.255
  network 192.168.20.0 0.0.0.255
#
return
```

- Leaf1的配置文件

```
#
sysname Leaf1
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:1
  vpn-target 100:5010 export-extcommunity evpn
  vpn-target 100:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 10
```



```
 vxlan vni 10
 evpn
  route-distinguisher 10:1
  vpn-target 100:10 export-extcommunity
  vpn-target 100:5010 export-extcommunity
  vpn-target 100:10 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ip address 10.10.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack1
 ip address 5.5.5.5 255.255.255.255
#
interface Nve1
 source 5.5.5.5
 vni 10 head-end peer-list protocol bgp
#
bgp 100 instance evpn1
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack1
 #
 l2vpn-family evpn
  policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 advertise irb
#
ospf 1
 area 0.0.0.0
  network 5.5.5.5 0.0.0.0
  network 192.168.10.0 0.0.0.255
#
return
```

- Leaf2的配置文件

```
#
sysname Leaf2
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:2
  vpn-target 100:5010 export-extcommunity evpn
  vpn-target 300:5010 export-extcommunity evpn
  vpn-target 100:5010 import-extcommunity evpn
  vpn-target 300:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 10:2
  vpn-target 100:20 export-extcommunity
  vpn-target 100:5010 export-extcommunity
  vpn-target 300:5010 export-extcommunity
  vpn-target 100:20 import-extcommunity
#
interface Vbdif20
 ip binding vpn-instance vpn1
```



```
 ip address 10.20.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.20.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.50.2 255.255.255.0
#
interface LoopBack1
 ip address 6.6.6.6 255.255.255.255
#
interface Nve1
 source 6.6.6.6
 vni 20 head-end peer-list protocol bgp
#
bgp 20
 peer 192.168.50.1 as-number 10
 #
 ipv4-family unicast
  network 6.6.6.6 255.255.255.255
  peer 192.168.50.1 enable
#
bgp 100 instance evpn1
 peer 5.5.5.5 as-number 100
 peer 5.5.5.5 connect-interface LoopBack1
 peer 7.7.7.7 as-number 200
 peer 7.7.7.7 connect-interface LoopBack1
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 5.5.5.5 enable
  peer 5.5.5.5 advertise irb
  peer 5.5.5.5 import reoriginate
  peer 5.5.5.5 advertise route-reoriginated evpn mac-ip
  peer 5.5.5.5 advertise route-reoriginated evpn ip
  peer 7.7.7.7 enable
  peer 7.7.7.7 advertise irb
  peer 7.7.7.7 import reoriginate
  peer 7.7.7.7 advertise route-reoriginated evpn mac-ip
  peer 7.7.7.7 advertise route-reoriginated evpn ip
#
ospf 1
 area 0.0.0.0
  network 6.6.6.6 0.0.0.0
  network 192.168.20.0 0.0.0.255
#
return
```

- Spine2的配置文件

```
#
sysname Spine2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.40.1 255.255.255.0
#
interface LoopBack1
 ip address 4.4.4.4 255.255.255.255
```

```
#
ospf 1
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 192.168.30.0 0.0.0.255
  network 192.168.40.0 0.0.0.255
#
return
```

- Leaf3的配置文件

```
#
sysname Leaf3
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:3
  vpn-target 200:5010 export-extcommunity evpn
  vpn-target 300:5010 export-extcommunity evpn
  vpn-target 200:5010 import-extcommunity evpn
  vpn-target 300:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:3
  vpn-target 200:10 export-extcommunity
  vpn-target 200:5010 export-extcommunity
  vpn-target 300:5010 export-extcommunity
  vpn-target 200:10 import-extcommunity
#
interface Vbdif10
 ip binding vpn-instance vpn1
 ip address 10.30.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface 100GE1/0/3
 undo portswitch
 ip address 192.168.60.2 255.255.255.0
#
interface LoopBack1
 ip address 7.7.7.7 255.255.255.255
#
interface Nve1
 source 7.7.7.7
 vni 10 head-end peer-list protocol bgp
#
bgp 30
 peer 192.168.60.1 as-number 10
 #
 ipv4-family unicast
  network 7.7.7.7 255.255.255.255
  peer 192.168.60.1 enable
#
bgp 200 instance evpn1
 peer 6.6.6.6 as-number 100
 peer 6.6.6.6 connect-interface LoopBack1
 peer 8.8.8.8 as-number 200
 peer 8.8.8.8 connect-interface LoopBack1
```



```
 #
 l2vpn-family evpn
  undo policy vpn-target
  peer 6.6.6.6 enable
  peer 6.6.6.6 advertise irb
  peer 6.6.6.6 import reoriginate
  peer 6.6.6.6 advertise route-reoriginated evpn mac-ip
  peer 6.6.6.6 advertise route-reoriginated evpn ip
  peer 8.8.8.8 enable
  peer 8.8.8.8 advertise irb
  peer 8.8.8.8 import reoriginate
  peer 8.8.8.8 advertise route-reoriginated evpn mac-ip
  peer 8.8.8.8 advertise route-reoriginated evpn ip
#
ospf 1
 area 0.0.0.0
  network 7.7.7.7 0.0.0.0
  network 192.168.30.0 0.0.0.255
#
return
```

- Leaf4的配置文件

```
#
sysname Leaf4
#
evpn-overlay enable
#
ip vpn-instance vpn1
 ipv4-family
  route-distinguisher 20:4
  vpn-target 200:5010 export-extcommunity evpn
  vpn-target 200:5010 import-extcommunity evpn
 vxlan vni 5010
#
bridge-domain 20
 vxlan vni 20
 evpn
  route-distinguisher 10:4
  vpn-target 200:20 export-extcommunity
  vpn-target 200:5010 export-extcommunity
  vpn-target 200:20 import-extcommunity
#
interface Vbdif20
 ip binding vpn-instance vpn1
 ip address 10.40.1.1 255.255.255.0
 vxlan anycast-gateway enable
 arp collect host enable
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.40.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 20
 bridge-domain 20
#
interface LoopBack1
 ip address 8.8.8.8 255.255.255.255
#
interface Nve1
 source 8.8.8.8
 vni 20 head-end peer-list protocol bgp
#
bgp 200 instance evpn1
 peer 7.7.7.7 as-number 200
 peer 7.7.7.7 connect-interface LoopBack1
 #
 l2vpn-family evpn
  policy vpn-target
  peer 7.7.7.7 enable
```

```
 peer 7.7.7.7 advertise irb
#
ospf 1
 area 0.0.0.0
  network 8.8.8.8 0.0.0.0
  network 192.168.40.0 0.0.0.255
#
return
```

- Device1的配置文件

```
#
sysname Device1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.50.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.1 255.255.255.0
#
interface LoopBack1
 ip address 1.1.1.1 255.255.255.255
#
bgp 10
 peer 192.168.1.2 as-number 10
 peer 192.168.50.2 as-number 20
 #
 ipv4-family unicast
  peer 192.168.1.2 enable
  peer 192.168.1.2 next-hop-local
  peer 192.168.50.2 enable
#
return
```

- Device2的配置文件

```
#
sysname Device2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.60.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.1.2 255.255.255.0
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
#
bgp 10
 peer 192.168.1.1 as-number 10
 peer 192.168.60.2 as-number 30
 #
 ipv4-family unicast
  peer 192.168.1.1 enable
  peer 192.168.1.1 next-hop-local
  peer 192.168.60.2 enable
#
return
```

2.8.1.5 举例：配置 Segment VXLAN 实现二层互通（映射 VNI 模式）

组网需求

如图2-25所示，在数据中心A和数据中心B内部分别配置BGP EVPN方式建立VXLAN隧道，通过在Leaf2和Leaf3之间配置BGP EVPN方式建立VXLAN隧道。当VM1和VM2之

间需要通信时，需要实现数据中心A和数据中心B之间的二层互通。本例中，数据中心A内部的VXLAN隧道采用的VNI是10，数据中心B内部的VXLAN隧道采用的VNI是20，此时，在Leaf2和Leaf3上配置到达对端的VXLAN隧道时，需要配置Segment VXLAN功能进行VNI的转换。

图 2-25 配置 Segment VXLAN 实现二层互通组网图

说明

本例中interface1、interface2分别代表100GE1/0/1、100GE1/0/2。

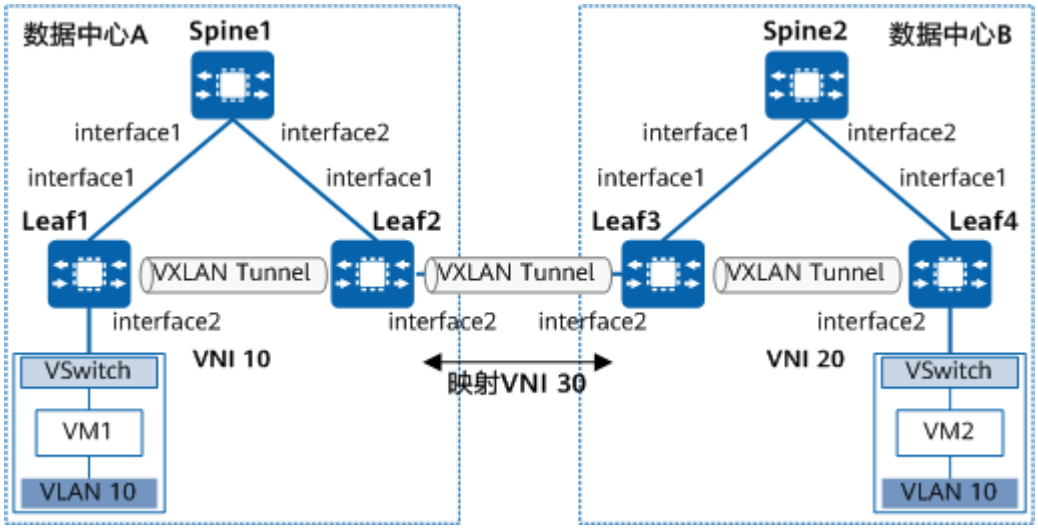


表 2-5 接口的 IP 地址

设备	接口	IP地址	设备	接口	IP地址
Spine1	100GE1/0/1	192.168.10.1/24	Spine2	100GE1/0/1	192.168.30.1/24
	100GE1/0/2	192.168.20.1/24		100GE1/0/2	192.168.40.1/24
Leaf1	100GE1/0/1	192.168.10.2/24	Leaf4	100GE1/0/1	192.168.40.2/24
	100GE1/0/2	-		100GE1/0/2	-
	LoopBack1	1.1.1.1/32		LoopBack1	4.4.4.4/32
Leaf2	100GE1/0/1	192.168.20.2/24	Leaf3	100GE1/0/1	192.168.30.2/24
	100GE1/0/2	192.168.50.1/24		100GE1/0/2	192.168.50.2/24
	LoopBack1	2.2.2.2/32		LoopBack1	3.3.3.3/32

配置思路

采用如下的思路配置Segment VXLAN实现二层互通（VNI映射模式）：

1. 配置各节点IP地址。
2. 配置路由协议实现各节点之间的互通。
3. 在数据中心A和数据中心B内配置BGP EVPN方式建立VXLAN隧道。
4. 在Leaf2和Leaf3上配置EBGP EVPN方式建立数据中心之间的VXLAN隧道。
5. 在Leaf2和Leaf3上配置Segment VXLAN。

操作步骤

步骤1 配置各设备接口IP地址

按图2-25分别配置所有设备上的接口IP地址。

步骤2 配置路由协议

在数据中心内配置IGP，本示例使用OSPF。在数据中心间配置EBGP。详细配置方法请参考配置脚本。

步骤3 数据中心A和数据中心B内配置BGP EVPN方式建立VXLAN隧道

1. 在Leaf1和Leaf4上配置业务接入点

   # 配置Leaf1。Leaf4的配置与Leaf1类似，这里不再赘述。

   ```
   [~Leaf1] bridge-domain 10
   [*Leaf1-bd10] quit
   [*Leaf1] interface 100ge 1/0/2.1 mode l2
   [*Leaf1-100GE1/0/2.1] encapsulation dot1q vid 10
   [*Leaf1-100GE1/0/2.1] bridge-domain 10
   [*Leaf1-100GE1/0/2.1] quit
   [*Leaf1] commit
   ```

2. 在各Leaf上使能EVPN作VXLAN控制平面功能

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述。

   ```
   [~Leaf1] evpn-overlay enable
   [*Leaf1] commit
   ```

3. 在数据中心A的Leaf1和Leaf2之间、数据中心B的Leaf3和Leaf4之间配置BGP EVPN对等体关系

   # 在Leaf1上配置BGP EVPN对等体关系。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述。

   ```
   [~Leaf1] bgp 100 instance evpn1
   [*Leaf1-bgp-instance-evpn1] peer 2.2.2.2 as-number 100
   [*Leaf1-bgp-instance-evpn1] peer 2.2.2.2 connect-interface LoopBack1
   [*Leaf1-bgp-instance-evpn1] l2vpn-family evpn
   [*Leaf1-bgp-instance-evpn1-af-evpn] peer 2.2.2.2 enable
   Warning: This operation will reset the peer session. Continue? [Y/N]: y
   [*Leaf1-bgp-instance-evpn1-af-evpn] quit
   [*Leaf1-bgp-instance-evpn1] quit
   [*Leaf1] commit
   ```

4. 在Leaf1和Leaf4上配置EVPN实例

   # 配置Leaf1。Leaf4的配置与Leaf1类似，这里不再赘述。

   ```
   [~Leaf1] bridge-domain 10
   [~Leaf1-bd10] vxlan vni 10
   [*Leaf1-bd10] evpn
   [*Leaf1-bd10-evpn] route-distinguisher 10:1
   [*Leaf1-bd10-evpn] vpn-target 300:30
   [*Leaf1-bd10-evpn] quit
   ```

```
[*Leaf1-bd10] quit
[*Leaf1] commit
```

5. 在各Leaf上使能头端复制功能

   # 配置Leaf1。Leaf2、Leaf3、Leaf4的配置与Leaf1类似，这里不再赘述。

```
[~Leaf1] interface nve 1
[*Leaf1-Nve1] source 1.1.1.1
[*Leaf1-Nve1] vni 10 head-end peer-list protocol bgp
[*Leaf1-Nve1] quit
[*Leaf1] commit
```

步骤4 在Leaf2和Leaf3之间配置EBGP EVPN对等体关系

配置Leaf2。

```
[~Leaf2] bgp 100 instance evpn1
[*Leaf2-bgp-instance-evpn1] peer 3.3.3.3 as-number 200
[*Leaf2-bgp-instance-evpn1] peer 3.3.3.3 connect-interface LoopBack1
[*Leaf2-bgp-instance-evpn1] peer 3.3.3.3 ebgp-max-hop 255
[*Leaf2-bgp-instance-evpn1] l2vpn-family evpn
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 3.3.3.3 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf2-bgp-instance-evpn1-af-evpn] quit
[*Leaf2-bgp-instance-evpn1] quit
[*Leaf2] commit
```

配置Leaf3。

```
[~Leaf3] bgp 200 instance evpn1
[*Leaf3-bgp-instance-evpn1] peer 2.2.2.2 as-number 100
[*Leaf3-bgp-instance-evpn1] peer 2.2.2.2 connect-interface LoopBack1
[*Leaf3-bgp-instance-evpn1] peer 2.2.2.2 ebgp-max-hop 255
[*Leaf3-bgp-instance-evpn1] l2vpn-family evpn
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 2.2.2.2 enable
Warning: This operation will reset the peer session. Continue? [Y/N]: y
[*Leaf3-bgp-instance-evpn1-af-evpn] quit
[*Leaf3-bgp-instance-evpn1] quit
[*Leaf3] commit
```

步骤5 在Leaf2和Leaf3上配置Segment VXLAN功能

1. 配置BGP EVPN对等体所属的水平分割组

   # 配置Leaf2。

```
[~Leaf2] bgp 100 instance evpn1
[~Leaf2-bgp-instance-evpn1] l2vpn-family evpn
[~Leaf2-bgp-instance-evpn1-af-evpn] peer 3.3.3.3 split-group sg1
[*Leaf2-bgp-instance-evpn1-af-evpn] commit
```

   # 配置Leaf3。

```
[~Leaf3] bgp 200 instance evpn1
[~Leaf3-bgp-instance-evpn1] l2vpn-family evpn
[~Leaf3-bgp-instance-evpn1-af-evpn] peer 2.2.2.2 split-group sg1
[*Leaf3-bgp-instance-evpn1-af-evpn] commit
```

2. 配置EVPN路由中的MAC路由的重生成功能

   # 配置Leaf2。

```
[~Leaf2-bgp-instance-evpn1-af-evpn] peer 1.1.1.1 import reoriginate
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 1.1.1.1 advertise route-reoriginated evpn mac
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 3.3.3.3 import reoriginate
[*Leaf2-bgp-instance-evpn1-af-evpn] peer 3.3.3.3 advertise route-reoriginated evpn mac
[*Leaf2-bgp-instance-evpn1-af-evpn] quit
[*Leaf2-bgp-instance-evpn1] quit
[*Leaf2] commit
```

配置Leaf3。

```
[~Leaf3-bgp-instance-evpn1-af-evpn] peer 4.4.4.4 import reoriginate
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 4.4.4.4 advertise route-reoriginated evpn mac
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 2.2.2.2 import reoriginate
[*Leaf3-bgp-instance-evpn1-af-evpn] peer 2.2.2.2 advertise route-reoriginated evpn mac
[*Leaf3-bgp-instance-evpn1-af-evpn] quit
[*Leaf3-bgp-instance-evpn1] quit
[*Leaf3] commit
```

3. 配置关联BD的映射VNI，并指定该映射VNI所属的水平分割组

配置Leaf2。

```
[~Leaf2] bridge-domain 10
[~Leaf2-bd10] vxlan vni 30 split-group sg1
[*Leaf2-bd10] quit
[*Leaf2] commit
```

配置Leaf3。

```
[~Leaf3] bridge-domain 10
[~Leaf3-bd10] vxlan vni 30 split-group sg1
[*Leaf3-bd10] quit
[*Leaf3] commit
```

步骤6 在Leaf2和Leaf3上配置EVPN实例

配置Leaf2。

```
[~Leaf2] bridge-domain 10
[~Leaf2-bd10] vxlan vni 10
[*Leaf2-bd10] evpn
[*Leaf2-bd10-evpn] route-distinguisher 10:2
[*Leaf2-bd10-evpn] vpn-target 300:30
[*Leaf2-bd10-evpn] quit
[*Leaf2-bd10] quit
[*Leaf2] commit
```

配置Leaf3。

```
[~Leaf3] bridge-domain 10
[~Leaf3-bd10] vxlan vni 20
[*Leaf3-bd10] evpn
[*Leaf3-bd10-evpn] route-distinguisher 10:3
[*Leaf3-bd10-evpn] vpn-target 300:30
[*Leaf3-bd10-evpn] quit
[*Leaf3-bd10] quit
[*Leaf3] commit
```

步骤7 在Leaf2和Leaf3上配置映射VNI的头端复制功能

配置Leaf2。

```
[~Leaf2] interface nve 1
[*Leaf2-Nve1] vni 30 head-end peer-list protocol bgp
[*Leaf2-Nve1] quit
[*Leaf2] commit
```

配置Leaf3。

```
[~Leaf3] interface nve 1
[*Leaf3-Nve1] vni 30 head-end peer-list protocol bgp
[*Leaf3-Nve1] quit
[*Leaf3] commit
```

----结束

检查配置结果

上述配置成功后，在Leaf上执行**display vxlan tunnel**命令可查看到VXLAN隧道的信息；执行**display vxlan peer**命令可查看到VXLAN的邻居信息。以Leaf2显示为例。

```
[~Leaf2] display vxlan tunnel
Number of vxlan tunnel : 2
Tunnel ID   Source              Destination         State  Type     Uptime
------------------------------------------------------------------------------
4026531924  2.2.2.2             1.1.1.1             up     dynamic  00:39:19
4026531925  2.2.2.2             3.3.3.3             up     dynamic  00:39:09
[~Leaf2] display vxlan peer
Number of peers : 2
Vni ID    Source              Destination         Type      Out Vni ID
------------------------------------------------------------------------------
10        2.2.2.2             1.1.1.1             dynamic   10
30        2.2.2.2             3.3.3.3             dynamic   30
```

配置完成后，VM1和VM2之间可以二层互通。

配置脚本

- Spine1的配置文件

```
#
sysname Spine1
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.20.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.10.0 0.0.0.255
  network 192.168.20.0 0.0.0.255
#
return
```

- Leaf1的配置文件

```
#
sysname Leaf1
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 10
 evpn
  route-distinguisher 10:1
  vpn-target 300:30 export-extcommunity
  vpn-target 300:30 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.10.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack1
 ip address 1.1.1.1 255.255.255.255
#
interface Nve1
 source 1.1.1.1
```

```
 vni 10 head-end peer-list protocol bgp
#
bgp 100 instance evpn1
 peer 2.2.2.2 as-number 100
 peer 2.2.2.2 connect-interface LoopBack1
 #
 l2vpn-family evpn
  policy vpn-target
  peer 2.2.2.2 enable
#
ospf 1
 area 0.0.0.0
  network 1.1.1.1 0.0.0.0
  network 192.168.10.0 0.0.0.255
#
return
```

- Leaf2的配置文件

```
#
sysname Leaf2
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 10
 vxlan vni 30 split-group sg1
 evpn
  route-distinguisher 10:2
  vpn-target 300:30 export-extcommunity
  vpn-target 300:30 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.20.2 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.50.1 255.255.255.0
#
interface LoopBack1
 ip address 2.2.2.2 255.255.255.255
#
interface Nve1
 source 2.2.2.2
 vni 10 head-end peer-list protocol bgp
 vni 30 head-end peer-list protocol bgp
#
bgp 10
 peer 192.168.50.2 as-number 20
 #
 ipv4-family unicast
  network 2.2.2.2 255.255.255.255
  peer 192.168.50.2 enable
#
bgp 100 instance evpn1
 peer 1.1.1.1 as-number 100
 peer 1.1.1.1 connect-interface LoopBack1
 peer 3.3.3.3 as-number 200
 peer 3.3.3.3 ebgp-max-hop 255
 peer 3.3.3.3 connect-interface LoopBack1
 #
 l2vpn-family evpn
  policy vpn-target
  peer 1.1.1.1 enable
  peer 1.1.1.1 import reoriginate
  peer 1.1.1.1 advertise route-reoriginated evpn mac
  peer 3.3.3.3 enable
  peer 3.3.3.3 split-group sg1
  peer 3.3.3.3 import reoriginate
  peer 3.3.3.3 advertise route-reoriginated evpn mac
```

```
#
ospf 1
 area 0.0.0.0
  network 2.2.2.2 0.0.0.0
  network 192.168.20.0 0.0.0.255
#
return
```

- Spine2的配置文件

```
#
sysname Spine2
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.1 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.40.1 255.255.255.0
#
ospf 1
 area 0.0.0.0
  network 192.168.30.0 0.0.0.255
  network 192.168.40.0 0.0.0.255
#
return
```

- Leaf3的配置文件

```
#
sysname Leaf3
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 20
 vxlan vni 30 split-group sg1
 evpn
  route-distinguisher 10:3
  vpn-target 300:30 export-extcommunity
  vpn-target 300:30 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.30.2 255.255.255.0
#
interface 100GE1/0/2
 undo portswitch
 ip address 192.168.50.2 255.255.255.0
#
interface LoopBack1
 ip address 3.3.3.3 255.255.255.255
#
interface Nve1
 source 3.3.3.3
 vni 20 head-end peer-list protocol bgp
 vni 30 head-end peer-list protocol bgp
#
bgp 20
 peer 192.168.50.1 as-number 10
 #
 ipv4-family unicast
  network 3.3.3.3 255.255.255.255
  peer 192.168.50.1 enable
#
bgp 200 instance evpn1
 peer 2.2.2.2 as-number 100
 peer 2.2.2.2 ebgp-max-hop 255
 peer 2.2.2.2 connect-interface LoopBack1
 peer 4.4.4.4 as-number 200
 peer 4.4.4.4 connect-interface LoopBack1
```

```
 #
 l2vpn-family evpn
  policy vpn-target
  peer 2.2.2.2 enable
  peer 2.2.2.2 split-group sg1
  peer 2.2.2.2 import reoriginate
  peer 2.2.2.2 advertise route-reoriginated evpn mac
  peer 4.4.4.4 enable
  peer 4.4.4.4 import reoriginate
  peer 4.4.4.4 advertise route-reoriginated evpn mac
#
ospf 1
 area 0.0.0.0
  network 3.3.3.3 0.0.0.0
  network 192.168.30.0 0.0.0.255
#
return
```

- Leaf4的配置文件

```
#
sysname Leaf4
#
evpn-overlay enable
#
bridge-domain 10
 vxlan vni 20
 evpn
  route-distinguisher 10:4
  vpn-target 300:30 export-extcommunity
  vpn-target 300:30 import-extcommunity
#
interface 100GE1/0/1
 undo portswitch
 ip address 192.168.40.2 255.255.255.0
#
interface 100GE1/0/2.1 mode l2
 encapsulation dot1q vid 10
 bridge-domain 10
#
interface LoopBack1
 ip address 4.4.4.4 255.255.255.255
#
interface Nve1
 source 4.4.4.4
 vni 20 head-end peer-list protocol bgp
#
bgp 200 instance evpn1
 peer 3.3.3.3 as-number 200
 peer 3.3.3.3 connect-interface LoopBack1
 #
 l2vpn-family evpn
  policy vpn-target
  peer 3.3.3.3 enable
#
ospf 1
 area 0.0.0.0
  network 4.4.4.4 0.0.0.0
  network 192.168.40.0 0.0.0.255
#
return
```

2.9 用户接入与认证

2.9.1 AAA

2.9.1.1 举例：配置 AAA 本地认证和授权

组网需求

如图2-26所示，企业希望管理员使用AAA本地认证，通过STelnet登录设备：

1. 管理员输入正确的用户名和密码才能通过STelnet登录设备。
2. 管理员通过STelnet登录设备后，授权管理员级别为3级。

图 2-26 配置 AAA 本地认证和授权示例

说明

本例中interface1代表100GE1/0/1。

配置思路

1. 配置STelnet登录：配置VTY界面的认证方式为AAA、使能STelnet服务、配置SSH用户的认证方式和服务方式。
2. 配置AAA本地认证：创建用户名和密码、配置用户的接入类型、配置用户级别。

配置注意事项

配置前请确保各设备之间路由可达。

操作步骤

步骤1 配置接口的IP地址。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*DeviceA] commit
[~DeviceA] vlan batch 10
[*DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] portswitch
[*DeviceA-100GE1/0/1] port link-type trunk
[*DeviceA-100GE1/0/1] port trunk allow-pass vlan 10
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface vlanif 10
[*DeviceA-Vlanif10] ip address 192.168.10.1 24
[*DeviceA-Vlanif10] quit
[*DeviceA] commit
```

步骤2 配置STelnet登录。

在服务器端生成本地密钥对。

```
[~DeviceA] rsa local-key-pair create
The key name will be:Host
The range of public key size is (2048, 3072).
NOTE: Key pair generation will take a short while.
Please input the modulus [default = 3072]:3072
```

配置VTY用户界面0～4的认证方式为AAA认证、支持的协议为SSH。

```
[*DeviceA] user-interface vty 0 4
[*DeviceA-ui-vty0-4] authentication-mode aaa
[*DeviceA-ui-vty0-4] protocol inbound ssh
[*DeviceA-ui-vty0-4] quit
```

开启设备的SSH服务器功能。

```
[*DeviceA] stelnet server enable
[*DeviceA] ssh server-source -i vlanif 10
```

配置所有SSH用户的认证方式为Password认证、服务方式为STelnet。

```
[*DeviceA] ssh authentication-type default password
[*DeviceA] commit
```

步骤3 配置AAA本地认证。

```
[~DeviceA] aaa
[~DeviceA-aaa] local-user user1-huawei password irreversible-cipher Huawei@123
[*DeviceA-aaa] local-user user1-huawei service-type ssh
[*DeviceA-aaa] local-user user1-huawei privilege level 3
[*DeviceA-aaa] quit
[*DeviceA] commit
```

----结束

检查配置结果

管理员通过STelnet客户端，输入正确的用户名和密码后，能够登录DeviceA。

配置脚本

```
#
sysname DeviceA
#
aaa
 local-user user1-huawei password irreversible-cipher $1d$OwseVRh@LH}ZeTBm$1nH4$ab>d(N{-%0!
ab48y=Ic*xEUR4pVhR2"9-~,$
 local-user user1-huawei privilege level 3
 local-user user1-huawei service-type ssh
#
vlan batch 10
#
interface Vlanif10
 ip address 192.168.10.1 24
#
interface 100GE1/0/1
 port link-type trunk
 port trunk allow-pass vlan 10
#
stelnet server enable
ssh server-source -i Vlanif 10
#
user-interface vty 0 4
 authentication-mode aaa
#
return
```

2.9.1.2 举例：配置 HWTACACS 认证、授权和计费

组网需求

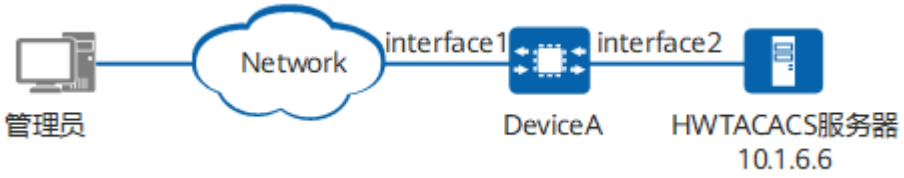
如图2-27所示，网络中部署了HWTACACS服务器，企业希望管理员使用HWTACACS认证，通过STelnet登录设备：

1. 管理员输入正确的用户名和密码才能通过STelnet登录设备。
2. 管理员通过STelnet登录设备后，授权管理员级别为3级，并且能够控制管理员不可执行部分命令并记录管理员执行过的命令。

图 2-27 配置 HWTACACS 认证、授权和计费示例

说明

本例中interface1和interface2分别代表100GE1/0/1和100GE1/0/2。



配置思路

1. 在DeviceA上配置STelnet登录：配置VTY界面的认证方式为AAA、使能STelnet服务、配置SSH用户的认证方式和服务方式。
2. 在DeviceA上配置HWTACACS认证：创建HWTACACS服务器模板，配置AAA方案、记录方案，并使能命令行授权功能。
3. 配置HWTACACS服务器。

配置注意事项

配置前请确保各设备之间路由可达。

操作步骤

步骤1 配置接口的IP地址。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*DeviceA] commit
[~DeviceA] vlan batch 10 20
[*DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] portswitch
[*DeviceA-100GE1/0/1] port link-tpye trunk
[*DeviceA-100GE1/0/1] port trunk allow-pass vlan 10
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] portswitch
[*DeviceA-100GE1/0/2] port link-tpye trunk
[*DeviceA-100GE1/0/2] port trunk allow-pass vlan 20
[*DeviceA-100GE1/0/2] quit
[*DeviceA] interface vlanif 10
[*DeviceA-Vlanif10] ip address 10.1.1.2 255.255.255.0
[*DeviceA-Vlanif10] quit
[*DeviceA] interface vlanif 20
[*DeviceA-Vlanif20] ip address 10.1.6.2 255.255.255.0
[*DeviceA-Vlanif20] quit
[*DeviceA] commit
```

步骤2 配置STelnet登录。

在服务器端生成本地密钥对。

```
[~DeviceA] rsa local-key-pair create
The key name will be:Host
```

```
The range of public key size is (2048, 3072).
NOTE: Key pair generation will take a short while.
Please input the modulus [default = 3072]:3072
```

配置VTY用户界面0～4的认证方式为AAA认证、支持的协议为SSH。

```
[*DeviceA] user-interface vty 0 4
[*DeviceA-ui-vty0-4] authentication-mode aaa
[*DeviceA-ui-vty0-4] protocol inbound ssh
[*DeviceA-ui-vty0-4] quit
```

开启设备的SSH服务器功能。

```
[*DeviceA] stelnet server enable
[*DeviceA] ssh server-source -i vlanif 10
```

配置所有SSH用户的认证方式为Password认证、服务方式为STelnet。

```
[*DeviceA] ssh authentication-type default password
[*DeviceA] commit
```

步骤3 配置HWTACACS认证、授权和计费。

配置HWTACACS服务器模板template1，实现设备与HWTACACS服务器的通信。

```
[~DeviceA] hwtacacs-server template template1
[*DeviceA-hwtacacs-template1] hwtacacs-server authentication 10.1.6.6 49
[*DeviceA-hwtacacs-template1] hwtacacs-server authorization 10.1.6.6 49
[*DeviceA-hwtacacs-template1] hwtacacs-server accounting 10.1.6.6 49
[*DeviceA-hwtacacs-template1] hwtacacs-server shared-key cipher Huawei@123456789
[*DeviceA-hwtacacs-template1] quit
[*DeviceA] commit
```

配置认证方案sch1，指定认证方式为HWTACACS认证。

```
[~DeviceA] aaa
[~DeviceA-aaa] authentication-scheme sch1
[*DeviceA-aaa-authen-sch1] authentication-mode hwtacacs
[*DeviceA-aaa-authen-sch1] quit
```

配置授权方案sch2，指定授权方式为HWTACACS授权，并使能3级用户的命令行授权功能。

```
[*DeviceA-aaa] authorization-scheme sch2
[*DeviceA-aaa-author-sch2] authorization-mode hwtacacs
[*DeviceA-aaa-author-sch2] authorization-cmd 3 hwtacacs
[*DeviceA-aaa-author-sch2] quit
```

配置记录方案sch0，记录用户执行的命令行。

```
[*DeviceA-aaa] recording-scheme sch0
[*DeviceA-aaa-recording-sch0] recording-mode hwtacacs template1
[*DeviceA-aaa-recording-sch0] quit
[*DeviceA-aaa] cmd recording-scheme sch0
```

配置计费方案sch3，指定计费方式为HWTACACS计费。

```
[*DeviceA-aaa] accounting-scheme sch3
[*DeviceA-aaa-accounting-sch3] accounting-mode hwtacacs
[*DeviceA-aaa-accounting-sch3] quit
```

在域huawei.com下引用HWTACACS服务器模板和AAA方案。

```
[*DeviceA-aaa] domain huawei.com
[*DeviceA-aaa-domain-huawei.com] hwtacacs-server template1
[*DeviceA-aaa-domain-huawei.com] authentication-scheme sch1
[*DeviceA-aaa-domain-huawei.com] authorization-scheme sch2
[*DeviceA-aaa-domain-huawei.com] accounting-scheme sch3
[*DeviceA-aaa-domain-huawei.com] quit
[*DeviceA-aaa] quit
```


配置域huawei.com为全局默认管理域。

```
[*DeviceA] domain huawei.com admin
[*DeviceA] commit
```

步骤4 配置HWTACACS服务器，以配置Secure ACS为例

配置步骤包括：添加设备、添加用户、配置用户级别为3级、配置命令行授权（不允许执行命令**reset hwtacacs-server statistics all**）。

在**Reports and Activity** -> **TACACS+ Administration**中可以查看所有用户（包括非HWTACACS认证用户）命令执行成功或失败的日志记录。

----结束

检查配置结果

- 管理员通过STelnet客户端，输入正确的用户名和密码后，能够登录DeviceA。
- 用户登录后，执行命令**reset hwtacacs-server statistics all**，提示Error: Failed to pass the authorization.表示命令行授权成功。

```
[~DeviceA] quit
<~DeviceA> reset hwtacacs-server statistics all
 Error: Failed to pass the authorization.
```

配置脚本

DeviceA的配置脚本

```
#
sysname DeviceA
#
hwtacacs-server template template1
 hwtacacs-server authentication 10.1.6.6
 hwtacacs-server authorization 10.1.6.6
 hwtacacs-server accounting 10.1.6.6
 hwtacacs-server shared-key cipher %+%##!!!!!!!!!"!!!!"!!!!*!!!!SKvr${[Fs."u,S-6a-X1'[X=L"cpF!5Oz`1!!!!!2jp5!!!!!!
A!!!!Ix>cM8i{y6!);(8Dr9:dK`&BHfE(H2=.:SH{@pT%+%#
#
aaa
 authentication-scheme sch1
  authentication-mode hwtacacs
 authorization-scheme sch2
  authorization-mode hwtacacs
  authorization-cmd 3 hwtacacs
 accounting-scheme sch3
  accounting-mode hwtacacs
 recording-scheme sch0
  recording-mode hwtacacs template1
 cmd recording-scheme sch0
 domain huawei.com
  authentication-scheme sch1
  accounting-scheme sch3
  authorization-scheme sch2
  hwtacacs-server template1
#
domain huawei.com admin
#
vlan batch 10 20
#
interface Vlanif10
 ip address 10.1.1.2 255.255.255.0
#
interface Vlanif20
 ip address 10.1.6.2 255.255.255.0
#
```



```
interface 100GE1/0/1
 port link-type trunk
 port trunk allow-pass vlan 10
#
interface 100GE1/0/2
 port link-type trunk
 port trunk allow-pass vlan 20
#
stelnet server enable
ssh server-source -i Vlanif10
#
user-interface vty 0 4
 authentication-mode aaa
#
return
```

2.9.1.3 举例：配置 RADIUS 认证、授权和计费

组网需求

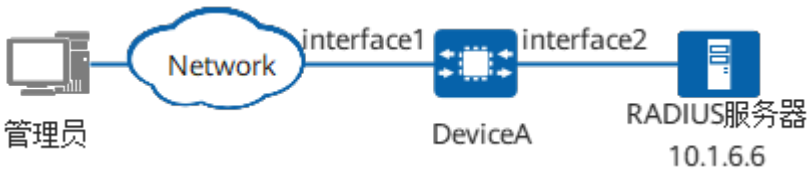
如图2-28所示，网络中部署了RADIUS服务器，企业希望管理员使用RADIUS认证方式，通过STelnet登录设备：

1. 管理员输入正确的用户名和密码才能通过STelnet登录设备。
2. 管理员通过STelnet登录设备后，授权管理员级别为3级。

图 2-28 配置 RADIUS 认证、授权和计费示例

说明

本例中interface1和interface2分别代表100GE1/0/1和100GE1/0/2。



配置思路

1. 在DeviceA上配置STelnet登录：配置VTY界面的认证方式为AAA、使能STelnet服务、配置SSH用户的认证方式和服务方式。
2. 在DeviceA上配置RADIUS认证：创建RADIUS服务器模板和AAA认证方案并在域下引用。
3. 配置RADIUS服务器。

配置注意事项

- 配置前请确保各设备之间路由可达。
- 请确保RADIUS服务器模板内的共享密钥和RADIUS服务器上的配置保持一致。
- 域被配置成全局默认管理域之后，管理用户的用户名中携带该域名或者不携带域名时，会使用全局默认管理域下的AAA配置信息。
- 如果RADIUS服务器不支持包含域名的用户名，可以在RADIUS服务器模板视图下，配置命令**undo radius-server user-name domain-included**使设备向RADIUS服务器发送的报文中的用户名不包含域名。

- 配置命令**undo radius-server user-name domain-included**后，设备仅会修改发送报文中的用户名格式，不会影响用户所属的域。例如，配置该命令后，用户名为“username@huawei.com”的用户仍使用huawei.com域下的AAA配置信息。
- 使用RADIUS扩展属性HW-Exec-Privilege（26-29）来授权管理员用户的优先级时，有效值范围是0～3。取值大于等于4为无效值。

操作步骤

步骤1 配置接口的IP地址。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*DeviceA] commit
[~DeviceA] vlan batch 10 20
[*DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] portswitch
[*DeviceA-100GE1/0/1] port link-tpye trunk
[*DeviceA-100GE1/0/1] port trunk allow-pass vlan 10
[*DeviceA-100GE1/0/1] quit
[*DeviceA] interface 100ge 1/0/2
[*DeviceA-100GE1/0/2] portswitch
[*DeviceA-100GE1/0/2] port link-tpye trunk
[*DeviceA-100GE1/0/2] port trunk allow-pass vlan 20
[*DeviceA-100GE1/0/2] quit
[*DeviceA] interface vlanif 10
[*DeviceA-Vlanif10] ip address 10.1.1.2 255.255.255.0
[*DeviceA-Vlanif10] quit
[*DeviceA] interface vlanif 20
[*DeviceA-Vlanif20] ip address 10.1.6.2 255.255.255.0
[*DeviceA-Vlanif20] quit
[*DeviceA] commit
```

步骤2 配置STelnet登录。

在服务器端生成本地密钥对。

```
[~DeviceA] rsa local-key-pair create
The key name will be:Host
The range of public key size is (2048, 3072).
NOTE: Key pair generation will take a short while.
Please input the modulus [default = 3072]:3072
```

配置VTY用户界面0～4的认证方式为AAA认证、支持的协议为SSH。

```
[*DeviceA] user-interface vty 0 4
[*DeviceA-ui-vty0-4] authentication-mode aaa
[*DeviceA-ui-vty0-4] protocol inbound ssh
[*DeviceA-ui-vty0-4] quit
```

开启设备的SSH服务器功能。

```
[*DeviceA] stelnet server enable
[*DeviceA] ssh server-source -i vlanif 10
```

配置所有SSH用户的认证方式为Password认证、服务方式为STelnet。

```
[*DeviceA] ssh authentication-type default password
[*DeviceA] commit
```

步骤3 配置RADIUS认证、授权和计费。

配置RADIUS服务器模板，实现与RADIUS服务器的通信。

```
[~DeviceA] radius-server template 1
[*DeviceA-radius-1] radius-server authentication 10.1.6.6 1812
[*DeviceA-radius-1] radius-server accounting 10.1.6.6 1813
```

```
[*DeviceA-radius-1] radius-server shared-key cipher Huawei@123456789
[*DeviceA-radius-1] quit
[*DeviceA] commit
```

配置AAA认证方案，指定认证方式为RADIUS。

```
[~DeviceA] aaa
[~DeviceA-aaa] authentication-scheme auth1
[*DeviceA-aaa-authen-auth1] authentication-mode radius
[*DeviceA-aaa-authen-auth1] quit
```

配置计费方案acc1，指定计费方式为RADIUS计费。

```
[*DeviceA-aaa] accounting-scheme acc1
[*DeviceA-aaa-accounting-acc1] accounting-mode radius
[*DeviceA-aaa-accounting-acc1] quit
```

在域下引用AAA方案、RADIUS服务器模板。

```
[*DeviceA-aaa] domain huawei.com
[*DeviceA-aaa-domain-huawei.com] authentication-scheme auth1
[*DeviceA-aaa-domain-huawei.com] accounting-scheme acc1
[*DeviceA-aaa-domain-huawei.com] radius-server 1
[*DeviceA-aaa-domain-huawei.com] quit
[*DeviceA-aaa] quit
[*DeviceA] commit
```

步骤4 配置管理员所属域为全局默认管理域，这样管理员通过STelnet登录设备时就不需要输入域名。

```
[~DeviceA] domain huawei.com admin
[*DeviceA] commit
```

步骤5 配置RADIUS服务器。

配置步骤包括：添加设备、添加用户、配置授权用户级别为3。

----结束

检查配置结果

管理员通过STelnet客户端，输入正确的用户名和密码后，能够登录DeviceA。

配置脚本

DeviceA的配置脚本

```
#
sysname DeviceA
#
radius-server template 1
 radius-server shared-key cipher %+%##!!!!!!!!!"!!!!"!!!!*!!!!SKvr${[Fs.3t@/5k|BENhEu>W(3\~XG!!D;!!!!!2jp5!!!!!!
A!!!!3"pK8qv!}9M#(4$jGWvQF/R[CNe/+:W^jk8HUe&W%+%#
 radius-server authentication 10.1.6.6 1812 weight 80
 radius-server accounting 10.1.6.6 1813 weight 80
#
aaa
 authentication-scheme auth1
  authentication-mode radius
 accounting-scheme acc1
  accounting-mode radius
 domain huawei.com
  authentication-scheme auth1
  accounting-scheme acc1
  radius-server 1
#
domain huawei.com admin
```



```
#
vlan batch 10 20
#
interface Vlanif10
 ip address 10.1.1.2 255.255.255.0
#
interface Vlanif20
 ip address 10.1.6.2 255.255.255.0
#
interface 100GE1/0/1
 port link-type trunk
 port trunk allow-pass vlan 10
#
interface 100GE1/0/2
 port link-type trunk
 port trunk allow-pass vlan 20
#
stelnet server enable
ssh server-source -i Vlanif10
#
user-interface vty 0 4
 authentication-mode aaa
#
return
```

2.10 安全

2.10.1 本机防攻击

2.10.1.1 举例：配置 CPU 防攻击

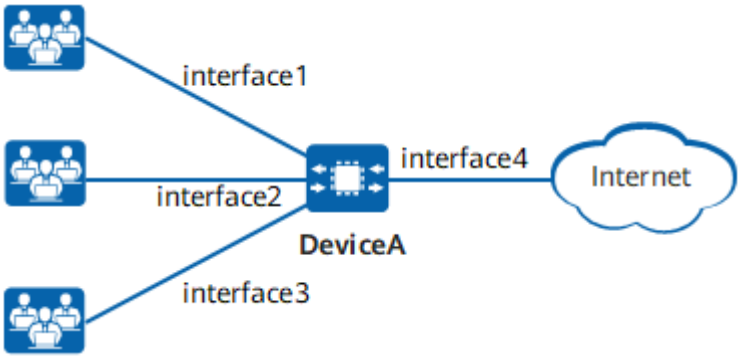
组网需求

如图2-29所示，大量用户通过DeviceA访问Internet，管理员发现攻击者发送大量的ARP Request报文，影响CPU的正常工作，希望能够减小ARP报文对CPU处理正常业务的影响。

图 2-29 配置本机防攻击示例组网图

说明

本例中interface1，interface2，interface3和interface4分别代表100GE1/0/1，100GE1/0/2，100GE1/0/3，100GE1/0/4。



操作步骤

步骤1 配置防攻击策略。

创建防攻击策略。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*DeviceA] commit
[~DeviceA] cpu-defend policy test1
```

配置ARP Request报文上送CPU的速率限制。

```
[*DeviceA-cpu-defend-policy-test1] car packet-type arp-request pps 128
[*DeviceA-cpu-defend-policy-test1] quit
[*DeviceA] commit
```

步骤2 全局应用防攻击策略。

```
[~DeviceA] cpu-defend-policy test1
[*DeviceA] commit
```

----结束

检查配置结果

查看配置的防攻击策略的信息。

```
[~DeviceA] display cpu-defend policy test1
=============================================
Policy name: test1
Policy applys on slot: <1>
Car packet-type arp-request(pps) : 128
=============================================
```

查看配置的CAR的信息。

```
[~DeviceA] display cpu-defend configuration all
Car configurations on slot 1 :
------------------------------------------------------------------------
PacketType          Status     Current(pps) Default(pps)     Queue
------------------------------------------------------------------------
arp-miss            Enabled          1536        1536        13
arp-reply           Enabled          2048        2048        23
arp-request         Enabled           128        2048        23
arp-request-uc      Enabled          2048        2048        23
......
```

配置文件

DeviceA的配置文件

```
#
sysname DeviceA
#
cpu-defend policy test1
 car packet-type arp-request pps 128
#
cpu-defend-policy test1
#
return
```

2.10.2 风暴抑制

2.10.2.1 举例：配置接口入方向的流量抑制

组网需求

如图2-30所示，DeviceA作为二层网络到三层设备的衔接点，需要通过接口入方向的流量抑制功能限制二层网络转发的广播、未知组播和未知单播报文，防止产生广播风暴。

图 2-30 配置接口入方向的流量抑制组网图

说明

本例中interface1代表100GE1/0/1。

操作步骤

步骤1 进入接口视图。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*DeviceA] commit
[~DeviceA] interface 100ge 1/0/1
[~DeviceA-100GE1/0/1] portswitch
```

步骤2 配置广播流量抑制，按承诺信息速率CIR进行抑制，限制广播报文的最大速率为100kbit/s。

```
[*DeviceA-100GE1/0/1] storm suppression broadcast cir 100
```

步骤3 配置未知组播流量抑制，按百分比（即报文速率和接口速率的比值）抑制，百分比值为80%。

```
[*DeviceA-100GE1/0/1] storm suppression multicast 80
```

步骤4 配置未知单播流量抑制，按承诺信息速率CIR进行抑制，限制未知单播报文的最大速率为100kbit/s。

```
[*DeviceA-100GE1/0/1] storm suppression unknown-unicast cir 100
[*DeviceA-100GE1/0/1] quit
[*DeviceA] commit
```

----结束

检查配置结果

查看接口入方向流量抑制的配置信息。

```
[~DeviceA] display storm suppression broadcast interface 100ge 1/0/1
-------------------------------------------------------------------------------
                    Configured                      Current
interface       percent(%) cir(kbps) cbs(bytes)    pps percent(%) cir(kbps) cbs(bytes)     pps
-------------------------------------------------------------------------------
100GE1/0/1              --      100      18800     --        --      100      18800      --
-------------------------------------------------------------------------------
[~DeviceA] display storm suppression multicast interface 100ge 1/0/1
-------------------------------------------------------------------------------
                    Configured                      Current
interface       percent(%) cir(kbps) cbs(bytes)    pps percent(%) cir(kbps) cbs(bytes)     pps
-------------------------------------------------------------------------------
```

```
100GE1/0/1              80      --      --      --      80      --      --      --
------------------------------------------------------------------------------
[~DeviceA] display storm suppression unknown-unicast interface 100ge 1/0/1
------------------------------------------------------------------------------
                        Configured                      Current
interface       percent(%) cir(kbps) cbs(bytes)    pps percent(%) cir(kbps) cbs(bytes)    pps
------------------------------------------------------------------------------
100GE1/0/1              --      100     18800     --        --      100     18800     --
------------------------------------------------------------------------------
```

其中，**Configured**字段显示已配置的流量抑制百分比值、承诺信息速率和承诺突发尺寸，**Current**字段显示实际生效的流量抑制百分比值、承诺信息速率和承诺突发尺寸。可以看出，DeviceA的接口100GE1/0/1在入方向限制广播报文的最大速率为100kbit/s，限制未知组播报文速率和接口速率的比值不超过80%，限制未知单播报文的最大速率为100kbit/s。

配置脚本

DeviceA

```
#
sysname DeviceA
#
interface 100GE1/0/1
 storm suppression broadcast cir 100 kbps
 storm suppression multicast 80
 storm suppression unknown-unicast cir 100 kbps
#
return
```

2.10.2.2 举例：配置风暴控制

组网需求

如图2-31所示，DeviceA作为二层网络到三层设备的衔接点，需要通过风暴控制限制二层网络转发的广播、未知组播和未知单播报文，防止产生广播风暴。

图 2-31 配置风暴控制组网图

说明

本例中interface1代表100GE1/0/1。

配置思路

采用如下思路配置风暴控制：

- 通过在100GE1/0/1接口视图下配置风暴功能，限制二层网络转发的广播、未知组播和未知单播报文产生广播风暴。
- 使能在风暴控制时记录日志的功能，以便及时提醒网络管理员采取措施来保护设备。

操作步骤

步骤1 进入接口视图。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*DeviceA] commit
[~DeviceA] interface 100ge 1/0/1
[~DeviceA-100GE1/0/1] portswitch
```

步骤2 配置广播、未知组播和未知单播报文的风暴控制高低阈值。在风暴控制检测时间间隔内，当接口接收广播、未知组播或未知单播报文的平均速率大于2000pps时，则对该接口对应类型的报文进行风暴控制；当接口接收广播、未知组播或未知单播报文的平均速率小于1000pps时，则将该接口对应类型的报文恢复到正常转发状态。

```
[*DeviceA-100GE1/0/1] storm control broadcast min-rate 1000 max-rate 2000
[*DeviceA-100GE1/0/1] storm control multicast min-rate 1000 max-rate 2000
[*DeviceA-100GE1/0/1] storm control unknown-unicast min-rate 1000 max-rate 2000
```

步骤3 配置风暴控制的动作为阻塞报文。

```
[*DeviceA-100GE1/0/1] storm control action block
```

步骤4 配置风暴控制的检测时间间隔为90秒。

```
[*DeviceA-100GE1/0/1] storm control interval 90
```

步骤5 使能在风暴控制时记录日志的功能。

```
[*DeviceA-100GE1/0/1] storm control enable log
[*DeviceA-100GE1/0/1] quit
[*DeviceA] commit
```

----结束

检查配置结果

查看风暴控制的配置信息。

```
[~DeviceA] display storm control interface 100ge 1/0/1
-------------------------------------------------------------------------------
NOTE:
BC = Broadcast; MC = Multicast; UUC = Unknown Unicast
Int = Interval value (unit: seconds)
-------------------------------------------------------------------------------
PortName     Type   MaxRate Mode Action    Punish-   Trap Log  Int Last
                                 Status              Punish-Time
-------------------------------------------------------------------------------
100GE1/0/1   BC       2000 Pps  Block     Normal    Off  On    90 --
100GE1/0/1   MC       2000 Pps  Block     Normal    Off  On    90 --
100GE1/0/1   UUC      2000 Pps  Block     Normal    Off  On    90 --
```

其中，**Punish-Status**字段显示当前接口的报文状态，**Last Punish-Time**字段显示上一次实施风暴控制惩罚的时间。可以看出，DeviceA的接口100GE1/0/1的广播、未知组播和未知单播报文状态正常，且没有出现实施风暴控制惩罚，说明广播、未知组播和未知单播报文在检测时间间隔内的平均速率没有超过设定值，网络状态良好。

配置脚本

DeviceA的配置脚本

```
#
sysname DeviceA
#
interface 100GE1/0/1
 storm control broadcast min-rate 1000 max-rate 2000
 storm control multicast min-rate 1000 max-rate 2000
```

```
 storm control unknown-unicast min-rate 1000 max-rate 2000
 storm control interval 90
 storm control action block
 storm control enable log
#
return
```

2.11 QoS

2.11.1 报文过滤

2.11.1.1 举例：配置基于 MQC 的报文过滤

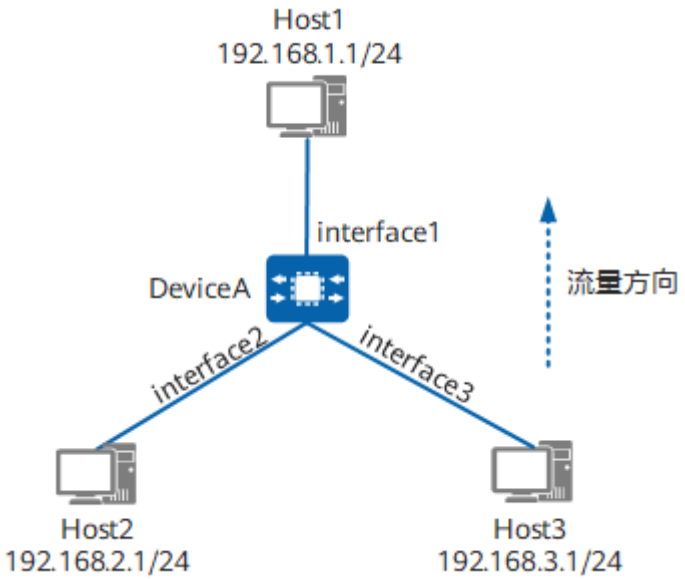
组网需求

如图2-32所示，Host1、2、3通过DeviceA实现互访。现要求Host1的用户可以通过DeviceA收到Host2发送的流量，但是不能收到Host3发送的流量。

图 2-32 配置报文过滤组网图

说明

本例中interface1，interface2，interface3分别代表100GE1/0/1，100GE1/0/2和100GE1/0/3。



操作步骤

步骤1 配置ACL规则

在DeviceA上创建ACL3001，匹配源IP为192.168.3.1且目的IP为192.168.1.1的流，即从Host3到Host1的流量。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
```

```
[*HUAWEI] commit
[~DeviceA] acl 3001
[*DeviceA-acl4-advance-3001] rule permit ip destination 192.168.1.1 24 source 192.168.3.1 24
[*DeviceA-acl4-advance-3001] quit
[*DeviceA] commit
```

步骤2 配置流分类

在DeviceA上创建流分类c1，匹配规则为ACL3001。

```
[~DeviceA] traffic classifier c1
[*DeviceA-classifier-c1] if-match acl 3001
[*DeviceA-classifier-c1] quit
[*DeviceA] commit
```

步骤3 配置流行为

在DeviceA上创建流行为b1，并配置禁止动作。

```
[~DeviceA] traffic behavior b1
[*DeviceA-behavior-b1] deny
[*DeviceA-behavior-b1] quit
[*DeviceA] commit
```

步骤4 配置流策略并应用到接口100GE1/0/1的出方向上

在DeviceA上创建流策略p1，将流分类和对应的流行为进行绑定。

```
[~DeviceA] traffic policy p1
[*DeviceA-trafficpolicy-p1] classifier c1 behavior b1
[*DeviceA-trafficpolicy-p1] quit
[*DeviceA] commit
```

将流策略p1应用到接口100GE1/0/1的出方向上。

```
[~DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] traffic-policy p1 outbound
[*DeviceA-100GE1/0/1] quit
[*DeviceA] commit
[~DeviceA] quit
```

----结束

检查配置结果

查看ACL规则的配置信息。

```
<DeviceA> display acl 3001
Advanced ACL 3001, 1 rule
ACL's step is 5
 rule 5 permit ip source 192.168.3.0 0.0.0.255 destination 192.168.1.0 0.0.0.255
 (0 times matched)
```

查看流分类的配置信息。

```
<DeviceA> display traffic classifier c1
  Traffic Classifier Information:
    Classifier: c1
      Type: OR
      Rule(s):
        if-match acl 3001
```

查看流策略的配置信息。

```
<DeviceA> display traffic policy p1
  Traffic Policy Information:
    Policy: p1
      Classifier: c1
```



```
    Type: OR
  Behavior: b1
    Deny
```

配置脚本

DeviceA的配置脚本

```
#
sysname DeviceA
#
acl number 3001
 rule 5 permit ip source 192.168.3.0 0.0.0.255 destination 192.168.1.0 0.0.0.255
#
traffic classifier c1 type or
 if-match acl 3001
#
traffic behavior b1
 deny
#
traffic policy p1
 classifier c1 behavior b1 precedence 5
#
interface 100GE1/0/1
 traffic-policy p1 outbound
#
return
```

2.12 系统监控

2.12.1 镜像

2.12.1.1 举例：配置本地端口镜像（1:1）

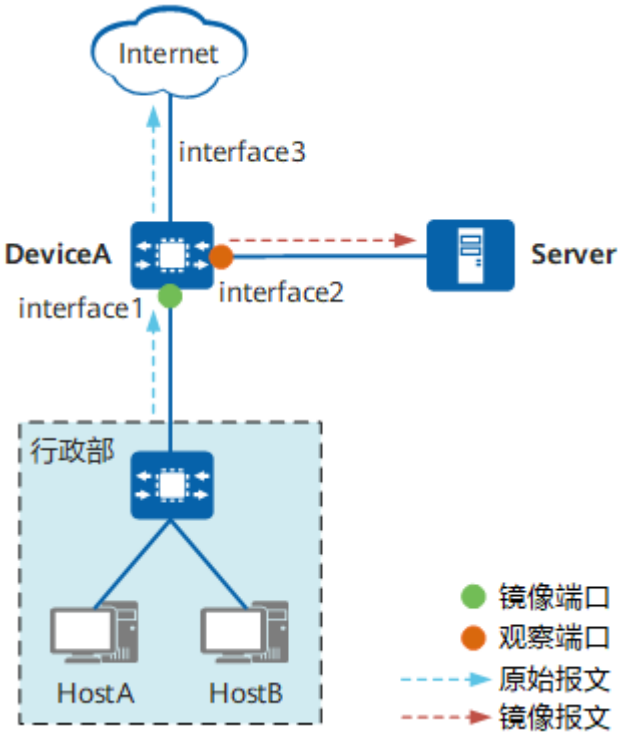
组网需求

如图2-33所示，某公司行政部通过DeviceA与外部Internet通信，监控设备Server与DeviceA直连。用户希望通过监控设备Server对行政部访问Internet的流量进行监控。

图 2-33 配置本地端口镜像组网图

说明

本例中interface1、interface2和interface3分别代表100GE1/0/1、100GE1/0/2、100GE1/0/3。



操作步骤

步骤1 在DeviceA上配置100GE1/0/2为本地观察端口。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] observe-port 1 interface 100ge 1/0/2
[*DeviceA] commit
```

步骤2 在DeviceA上配置100GE1/0/1为镜像端口，以监控行政部主机发送的报文。

```
[~DeviceA] interface 100ge 1/0/1
[~DeviceA-100GE1/0/1] port-mirroring observe-port 1 inbound
[*DeviceA-100GE1/0/1] quit
[*DeviceA] commit
```

----结束

检查配置结果

查看镜像的配置信息。

```
[~DeviceA] display port-mirroring
 Observe port mirroring:
 ------------------------------------------------------------------------------
 MirroringPort       Direction       ObservePort : Interface
 ------------------------------------------------------------------------------
 100GE1/0/1          Inbound                 1 : 100GE1/0/2
 ------------------------------------------------------------------------------
```

配置脚本

DeviceA

```
#
sysname DeviceA
```

```
#
observe-port 1 interface 100GE1/0/2
#
interface 100GE1/0/1
 port-mirroring observe-port 1 inbound
#
return
```

2.12.1.2 举例：配置本地基于 MQC 的流镜像

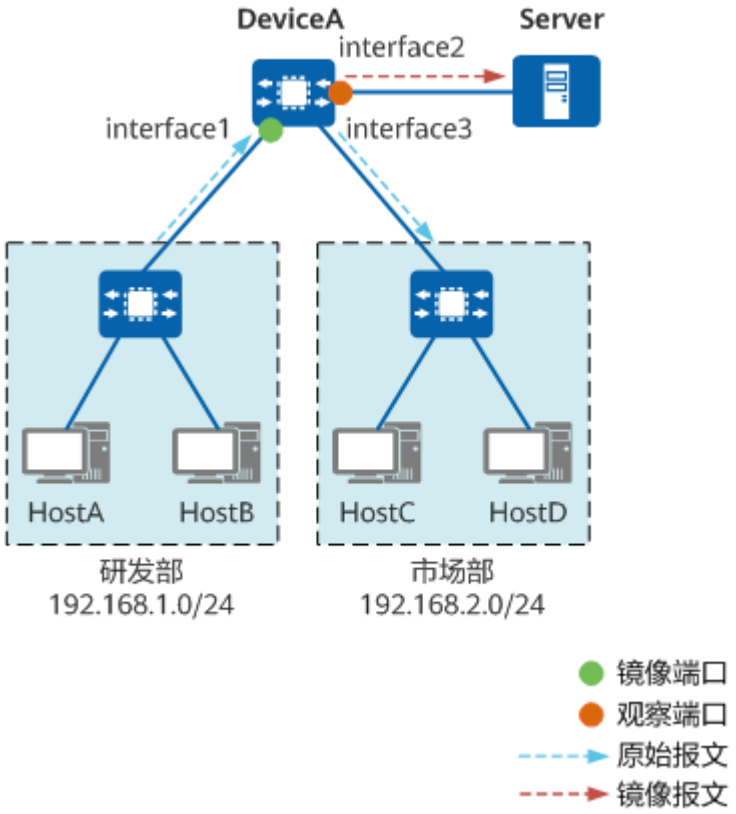
组网需求

如图2-34所示，某公司研发部和市场部分别使用192.168.1.0/24和192.168.2.0/24两个网段地址，通过DeviceA进行通信，监控设备Server与DeviceA直连。用户希望通过监控设备Server对研发部发往市场部的报文进行监控。

图 2-34 配置本地基于 MQC 的流镜像组网图

说明

本例中interface1、interface2、interface3分别代表100GE1/0/1、100GE1/0/2、100GE1/0/3。



操作步骤

步骤1 在DeviceA上配置100GE1/0/2为观察端口。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceA
[*HUAWEI] commit
[~DeviceA] observe-port 1 interface 100ge 1/0/2
[*DeviceA] commit
```

步骤2 在DeviceA上创建流分类c1，并配置流分类规则为匹配源地址为192.168.1.0/24，目的地址为192.168.2.0/24的报文。

```
[~DeviceA] acl number 3000
[*DeviceA-acl4-advance-3000] rule permit ip source 192.168.1.0 24 destination 192.168.2.0 24
[*DeviceA-acl4-advance-3000] quit
[*DeviceA] traffic classifier c1
[*DeviceA-classifier-c1] if-match acl 3000
[*DeviceA-classifier-c1] quit
[*DeviceA] commit
```

步骤3 在DeviceA上创建流行为b1，并配置流镜像动作。

```
[~DeviceA] traffic behavior b1
[*DeviceA-behavior-b1] mirroring observe-port 1
[*DeviceA-behavior-b1] quit
[*DeviceA] commit
```

步骤4 在DeviceA上创建流策略p1，将流分类和对应的流行为进行绑定；并将流策略应用到接口100GE1/0/1的入方向上，对研发部访问市场部的报文进行监控。

```
[~DeviceA] traffic policy p1
[*DeviceA-trafficpolicy-p1] classifier c1 behavior b1
[*DeviceA-trafficpolicy-p1] quit
[*DeviceA] interface 100ge 1/0/1
[*DeviceA-100GE1/0/1] traffic-policy p1 inbound
[*DeviceA-100GE1/0/1] quit
[*DeviceA] commit
```

----结束

检查配置结果

查看流分类的配置信息。

```
[~DeviceA] display traffic classifier c1
  Traffic Classifier Information:
    Classifier: c1
      Type: OR
      Rule(s):
        if-match acl 3000
```

查看流策略的配置信息。

```
[~DeviceA] display traffic policy p1
  Traffic Policy Information:
    Policy: p1
      Classifier: c1
        Type: OR
      Behavior: b1
        Mirroring observe-port 1
```

查看镜像的配置信息。

```
[~DeviceA] display port-mirroring
  Traffic mirroring:
  ------------------------------------------------------------------------------
  TrafficBehavior                  ObservePort : Interface
  ------------------------------------------------------------------------------
  b1                                     1 : 100GE1/0/2
  ------------------------------------------------------------------------------
```

配置脚本

DeviceA

```
#
sysname DeviceA
```

```
#
observe-port 1 interface 100GE1/0/2
#
acl number 3000
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.2.0 0.0.0.255
#
traffic classifier c1 type or
 if-match acl 3000
#
traffic behavior b1
 mirroring observe-port 1
#
traffic policy p1
 classifier c1 behavior b1 precedence 5
#
interface 100GE1/0/1
 traffic-policy p1 inbound
#
return
```

2.12.2 NetStream

2.12.2.1 举例：配置原始流统计信息的输出功能

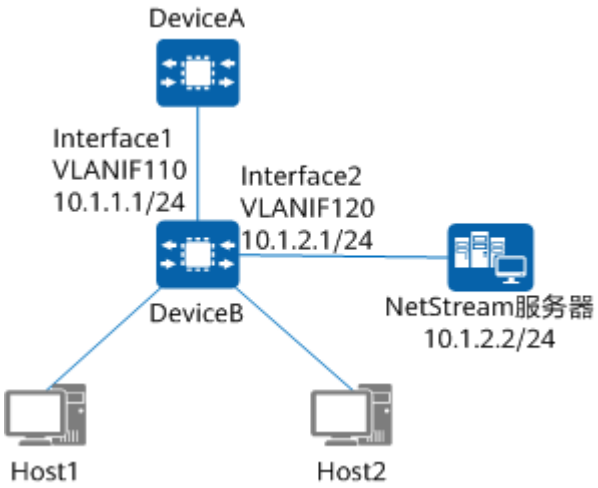
组网需求

如图 Netstream组网图所示，Host1与Host2通过DeviceB与DeviceA通信，网管人员希望掌握2个部门与DeviceA通信的流量统计信息，以便进行网络规划。

图 2-35 Netstream 组网图

说明

本例中interface1和interface2分别代表100GE1/0/1和100GE1/0/2。



操作步骤

步骤1 如图 Netstream组网图标注所示，配置DeviceB的接口IP地址。

```
<HUAWEI> system-view
[~HUAWEI] sysname DeviceB
[*DeviceB] commit
```

```
[~DeviceB] vlan 110
[*DeviceB-vlan110] quit
[*DeviceB] interface vlanif 110
[*DeviceB-Vlanif110] ip address 10.1.1.1 24
[*DeviceB-Vlanif110] quit
[*DeviceB] interface 100ge1/0/1
[*DeviceB-100GE1/0/1] port link-type trunk
[*DeviceB-100GE1/0/1] port trunk pvid vlan 110
[*DeviceB-100GE1/0/1] port trunk allow-pass vlan 110
[*DeviceB-100GE1/0/1] quit
[*DeviceB] vlan 120
[*DeviceB-vlan120] quit
[*DeviceB] interface vlanif 120
[*DeviceB-Vlanif120] ip address 10.1.2.1 24
[*DeviceB-Vlanif120] quit
[*DeviceB] interface 100ge1/0/2
[*DeviceB-100GE1/0/2] port link-type trunk
[*DeviceB-100GE1/0/2] port trunk pvid vlan 120
[*DeviceB-100GE1/0/2] port trunk allow-pass vlan 120
[*DeviceB-100GE1/0/2] quit
```

步骤2 配置NetStream采样功能。

配置100GE1/0/1接口出、入流量的NetStream采样功能，采样间隔为8192。

```
[*DeviceB] interface 100ge1/0/1
[*DeviceB-100GE1/0/1] netstream sampler random-packets 8192 inbound
[*DeviceB-100GE1/0/1] netstream sampler random-packets 8192 outbound
[*DeviceB-100GE1/0/1] quit
```

步骤3 配置NetStream的流老化。

配置非活跃流的老化时间为100秒，以及开启由TCP连接的FIN和RST报文触发老化。

```
[*DeviceB] netstream timeout ip inactive 100
[*DeviceB] netstream timeout ip tcp-session
```

步骤4 配置NetStream原始流统计信息输出。

配置NetStream原始流统计信息输出报文源地址为10.1.2.1，目的地址为10.1.2.2，目的端口号为6000，DSCP为0。

```
[*DeviceB] netstream export ip source 10.1.2.1
[*DeviceB] netstream export ip host 10.1.2.2 6000 dscp 0
```

步骤5 配置输出报文版本格式。

缺省情况下，输出报文版本格式为V9。

步骤6 使能接口的原始流NetStream统计功能。

使能100GE1/0/1接口出、入流量的NetStream统计功能。

```
[*DeviceB] interface 100ge1/0/1
[*DeviceB-100GE1/0/1] netstream outbound ip
[*DeviceB-100GE1/0/1] netstream inbound ip
[*DeviceB-100GE1/0/1] quit
[*DeviceB] commit
```

步骤7 验证配置结果。

配置成功后，查看流量统计信息显示如下。

```
[~DeviceB] display netstream statistics ip slot 1
 Last time when statistics were cleared: -
------------------------------------------------------------------------------
Packet Length    : Number
------------------------------------------------------------------------------
1     ~    64   : 0
```



```
65     ~    128  : 14
129    ~    256  : 1
257    ~    512  : 0
513    ~    1024 : 0
1025   ~    1500 : 0
longer than 1500 : 0
------------------------------------------------------------------------------
StreamType
     Current          Aged        Created       Exported       Exported
     (streams)         (streams)   (streams)     (streams)      (Packets)
------------------------------------------------------------------------------
origin
          0             0             0             0             0
------------------------------------------------------------------------------
```

----结束

配置脚本

DeviceB

```
#
sysname DeviceB
#
vlan batch 110 120
#
netstream timeout ip inactive 100
netstream timeout ip tcp-session
netstream export ip source 10.1.2.1
netstream export ip host 10.1.2.2 6000 dscp 0
#
interface Vlanif110
 ip address 10.1.1.1 255.255.255.0
#
interface Vlanif120
 ip address 10.1.2.1 255.255.255.0
#
interface 100GE1/0/1
 port link-type trunk
 port trunk pvid vlan 110
 port trunk allow-pass vlan 110
 netstream inbound ip
 netstream outbound ip
 netstream sampler random-packets 8192 inbound
 netstream sampler random-packets 8192 outbound
#
interface 100GE1/0/2
 port link-type trunk
 port trunk pvid vlan 120
 port trunk allow-pass vlan 120
#
return
```

