CN9000/CN6000 系列
/S6820/S6550E/S6220/S5560V2 系列
配置指导手册

前 言

概述

本文档系统介绍了三层以太网交换机设备支持的特性及其相关配置。主要内容包括基础配置、以太网、环网保护、IP 业务、IP 路由、可靠性、安全性、QoS、IPV6 等基本原理和配置过程，并提供相关的配置案例。在本文档的附录中，提供了该文档所涉及的术语和缩略语。

阅读本文档有助于读者系统掌握设备的原理和各种配置信息，以及如何应用该设备进行组网。

该手册适用于以下交换机系列型号，包括：

数据中心 CN9000 系列（含 CN9408H/CN9300-48Y8C/CN9008-48YC-S/CN9100-48X8C）、CN6000 系列（含 CN61108PC-V-H）；

园区网 S6820 系列（S6820-24XQ-E）、S6550E 系列（含 S6550E-48T4X-C/S6550E-48TS4X-C/S6550E-48S4X-C）、S6220 系列（含 S6220-24TQ-S-PWR/S6220-48TQ-S-PWR/S6220-24TQ-S/S6220-48TQ-S/S6220-24S4X-S）、S5560V2 系列（含 S5560V2-48T4X-S/ S5560V2-48T4S-S/S5560V2-24T4X-S/S5560V2-24T4S-S/S5560V2-24TS-L-PWR/S5560V2-24T4X-HS）；

注：每款产品所支持特性有差异，具体请以产品实际支持功能为准。

约定

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 警告	以本标志开始的文本表示有潜在危险，如果不能避免，可能导致人员伤害。
 注意	以本标志开始的文本表示有潜在风险，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
 说明	以本标志开始的文本是正文的附加信息，是对正文的强调和补充。

符号	说明
	以本标志开始的文本能帮助您解决某个问题或节省您的时间。

通用格式约定

格式	说明
宋体	正文采用宋体表示。
黑体	一级标题、二级标题、三级标题、Block 采用**黑体**表示。
楷体	警告、提示等内容用楷体表示。
“Lucida Console”格式	“Lucida Console”格式表示屏幕输出信息。此外，屏幕输出信息中夹杂的用户从终端输入的信息采用加粗字体表示。

命令行格式约定

格式	说明
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**粗体**表示。
斜体	命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
{ x | y | ... }	表示从两个或多个选项中选取一个。
[x | y | ...]	表示从两个或多个选项中选取一个或者不选。
{ x | y | ... } *	表示从两个或多个选项中选取多个，最少选取一个，最多选取所有选项。
[x | y | ...] *	表示从两个或多个选项中选取多个或者不选。

目 录

1 基础配置指导

1.1 系统管理配置

1.1.1 简介

MOTD 信息(message-of-the-day)和登录提示信息都是可配置的，可以显示给所有登录到系统的用户。如果某用户出现了不当操作可能影响到网上所有的用户，给该用户发送提示信息是非常有必要的(比如注销系统)。登录提示信息会在终端用户登录到系统时显示。

1.1.2 配置消息 Banner

用户可以创建一个或多个提示信息，这些信息将会显示在已登录用户的终端上。可以通过以下步骤配置此功能。

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# banner motd c message c	指定相应的字符串，最多 255 个字符串
Switch(config)# exit	退出配置模式

1.1.3 配置登录 Banner

用户可以配置一条登录提示信息，以显示给所有登录到系统的用户，可以通过以下步骤配置此功能。

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# banner login c message c	指定相应的字符串，最多 255 个字符串
Switch(config)# exit	退出配置模式

1.1.4 配置退出 Banner

用户可以配置一条 EXEC 模式的提示信息，以显示给所有登录到 EXEC 模式的用户，可以通过以下步骤配置此功能。

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# banner exec c message c	指定相应的字符串，最多 255 个字符串
Switch(config)# exit	退出配置模式

1.1.5 显示 Banner 信息

显示当前所有的配置

Switch1

Switch# show running	显示系统当前的配置

1.2 用户管理配置

1.2.1 概述

用户管理功能可用来增加系统的安全性，用户可以通过密码来登录。系统会限制登录用户的数量。交换机上有三种模式登录：“no login”模式，任何人都可以直接登录交换机并且不需要密码；“login”模式，只有默认的用户登录；“login local”模式，假如用户想登录交换机，必须在系统中创建一个用户帐号。在本地创建用户帐号和密码可以帮助用户登录交换机。每个交换机只有 32 个账户。在用户启用本地账户验证之前，必须提前创建一个账户。

用户可以为每个用户名设置不同的密码。每个用户名不能超过 32 个字符。

用户可以设置每个账户的等级，有效的等级 1-4。只有一个账户可以进入配置模式。

1.2.2 配置用户等级

I. 配置

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# line vty 0 7	进入用户模式
Switch(config-line)# login local	设置验证模式
Switch(config-line)# exit	退出用户模式
Switch(config)# username testname privilege 4 password 123abc<>	创建用户名和密码
Switch(config)# exit	退出配置模式

I. 命令验证

经过以上配置，登录交换机时，系统会提示类似如下验证信息：

```
Username: testname
Password:
```

1.2.3 配置用户管理

使用不带用户名的密码登录。

I. 配置

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# line vty 0 7	进入用户模式
Switch(config-line)# login	设置验证模式
Switch(config-line)# line-password abc	设置登录密码为 abc
Switch(config-line)# end	退出用户模式

I. 命令验证

经过以上配置后，登录交换机时系统会提示类似如下的验证信息，用户可以使用之前创建的密码来登录交换机。

```
Password:
```

1.2.4 密码恢复步骤

忘记密码，可以用以下方式恢复。

步骤 1 通过 Console 线连接交换机并加电。

```
CPU:   MPC8247 (HiP7 Rev 14, Mask 1.0 1K50M) at 350 MHz
Board: 8247 (PCI Agent Mode)
I2C:   ready
DRAM:  256 MB
In:    serial
Out:   serial
Err:   serial
Net:   FCC1 ETHERNET, FCC2 ETHERNET [PRIME]
Press ctrl+b to stop autoboot: 3
```

步骤 2 按 **ctrl + b** 进入 Uboot 模式。

步骤 3 根据以下配置，系统将正常进入交换机。

Bootrom# boot_flash_nopass	使用没有密码的空配置文件启动系统
Bootrom# Do you want to revert to the default config file ? [Y|N|E]:	输入“Y”

1.3 FTP 配置

1.3.1 简介

用户可从 FTP 服务器下载一个交换机配置文件，或从交换机上传文件到 FTP 服务器上。

从 FTP 服务器下载一个交换机的配置文件以升级交换机的配置，只需用新的文件覆盖当前的启动配置文件即可。交换机配置文件上传到服务器可以起到备份作用，如后续需要，可下载到本交换机或者相同类型的交换机，以更新交换机的配置。

1.3.2 IPv4 配置

I. 准备用 FTP 下载或上传配置文件

用户可以复制或上传文件到 FTP 服务器。

FTP 协议要求 FTP 客户端每次发送 FTP 请求到服务器时都要包含远程用户名和密码。在用户开始使用 FTP 上传或下载一个配置文件前，必须完成以下操作：

1. 确保交换机到 FTP 服务器之间有一个可达路由。如果用户的网络中不存在子网间路由通信，交换机和 FTP 服务器就必须要在同一网络中，通过 ping 命令检查 FTP 服务器的连通性。
2. 如果用户正通过控制台或 Telnet 访问交换机，需确保当前的 FTP 用户名有效，是一个可以使用 FTP 下载功能的用户名。
3. 当用户上传配置文件到 FTP 服务器，用户必须正确配置 FTP 服务器以接受来自交换机用户的写请求。

II. 通过 FTP 下载一个配置文件

用户可以下载一个新的配置文件来覆盖当前的配置。

Switch1

Switch# configure terminal	进入全局配置模式
Switch(config)# ftp username test	（可选）创建一个用户“test”
Switch(config)# ftp password test	（可选）创建一个密码 “test”
Switch(config)# end	退出 EXEC 模式
Switch# copy mgmt-if ftp://test:test@10.10.10.163/ startup-config.conf flash:/startup-config.conf	从远程 FTP 服务器下载启动配置文件，用户名“test”，密码“test”
Switch# show startup-config	显示配置

I. 通过 FTP 上传配置文件

用户可以从一个 FTP 服务器上传一个配置文件，稍后从这个交换机或其它交换机下载这个配置。

Switch1

Switch# configure terminal	进入全局配置模式
Switch(config)# ftp username test	（可选）创建一个用户“test”
Switch(config)# ftp password test	（可选）创建一个密码“test”
Switch(config)# end	退出 EXEC 模式
Switch# copy flash:/startup-config.conf mgmt-if ftp://test:test@10.10.10.163/startup-config.conf	向远程 FTP 服务器上传配置文件，用户名“test”，密码“test”

1.3.3 IPv6 配置

I. 通过 FTP 下载一个配置文件

Switch1

Switch# copy ftp://root:root@2001:1000::2/startup-config.conf flash:/startup-config.conf	从远程 FTP 服务器下载启动配置文件，用户名“root”，密码“root”
Switch# show startup-config	显示配置

I. 通过 FTP 上传配置文件

Switch1

Switch# copy flash:/startup-config.conf mgmt-if ftp://root:root@2001:1000::2 startup-config.conf	向远程 FTP 服务器上传配置文件，用户名“root”，密码“root”

1.4 TFTP 配置

1.4.1 概述

TFTP（Trivial File Transfer Protocol，简单文件传输协议）是 TCP/IP 协议族中的一个用来在客户机与服务器之间进行简单文件传输的协议，提供不复杂、开销不大的文件传输服务。端口号为 69。

此协议设计的时候是进行小文件传输的。因此它不具备通常的 FTP 的许多功能，它只能从文件服务器上获得或写入文件，不能列出目录，不进行认证，它传输 8 位数据。

1.4.2 配置

I. 通过 TFTP 服务器上传下载软件

在进行上传下载之前，需要执行如下操作：

- 确保作为 TFTP 服务器的工作站配置正确。
- 确保 Switch 到 TFTP 服务器的路由可达。如果子网间不存在进行路由通信的路由器，交换机和 TFTP 服务器必须在同一网络中。ping 命令可以检查是否能连接到 TFTP 服务器。
- 确保要下载的配置文件在 TFTP 服务器上的正确目录下。
- 下载操作，确保该文件的权限设置正确。

- 上传操作，如果要覆盖服务器上现有的文件（包括空文件），确保该文件的权限设置正确。

II. 下载

Switch1

Switch# copy mgmt-if tftp://10.10.10.163/startup-config.conf flash:/startup-config.conf	指定 TFTP 服务器的 IPv4 地址以及相应的文件
Switch# copy mgmt-if tftp://2001:1000::2/startup-config.conf flash:/startup-config.conf	指定 TFTP 服务器的 IPv6 地址以及相应的文件
Switch# show startup-config	检查下载的文件

I. 上传

Switch1

Switch# copy flash:/startup-config.conf mgmt-if tftp://10.10.10.163/startup-config.conf	指定上传的文件以及服务器的 IPv4 地址
Switch# copy flash:/startup-config.conf mgmt-if tftp://2001:1000::2/startup-config.conf	指定上传的文件以及服务器的 IPv6 地址

1.5 Telnet 配置

1.5.1 概述

Telnet 协议是 TCP/IP 协议族中的一员，是 Internet 远程登录服务的标准协议和主要方式。它为用户提供了在本地计算机上完成远程登录主机工作的能力。在终端用户的电脑上使用 Telnet 程序，用它连接到服务器。终端用户可以在 Telnet 程序中输入命令，这些命令会在服务器上运行，就像直接在服务器的控制台上输入一样。通过 Telnet 程序，用户在本地就能控制服务器。要开始一个 Telnet 会话，必须输入用户名和密码来登录服务器。Telnet 是常用的远程控制 Web 服务器的方法。

1.5.2 配置

步骤 1 通过带内口 Telnet 到其他交换机。

Switch# telnet 10.10.29.247	通过带内口 Telnet 到其他交换机
Switch# telnet 2001:1000::71	通过带内口 Telnet 到其他交换机

步骤 2 通过管理口 Telnet 到其他交换机。

Switch# telnet mgmt-if 10.10.29.247	通过管理口 Telnet 到其他交换机
Switch# telnet mgmt-if 2001:1000::2	通过管理口 Telnet 到其他交换机

步骤 3 交换机同样也是一个 Telnet 服务器。

Switch# configure terminal	进入配置模式
Switch(config)# service telnet enable	启用 Telnet 服务

1.5.3 命令验证

Switch# telnet mgmt-if 10.10.38.1

```
Entering character mode
Escape character is '^]'.
Switch #
```

Switch# telnet 2001:1000::71

```
Entering character mode
Escape character is '^]'.
Switch #
```

1.6 SSH 配置

1.6.1 概述

安全 shell（SSH）是一种协议，可为用户提供一个安全环境，远程连接到设备。当设备进行远程访问时，SSH 提供了比 Telnet 更强大的加密功能，SSH 支持数据加密标准（DES）加密算法，三重 DES（3DES）加密算法，并且提供基于密码的用户认证。

1.6.2 拓扑

图1-1 SSH system application

1.6.3 配置

I. 创建 SSH 的 KEY

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# rsa key a generate	创建一个 key
Switch(config)# rsa key a export url flash:/a.pri private ssh2	从 Flash 里面取一个 a.pri 私有 key
Switch(config)# rsa key a export url flash:/a.pub public ssh2	从 Flash 里面取一个 a.pub 公有 key

I. 导入 KEY

Switch1

Switch(config)# rsa key importKey import url flash:/a.pub public ssh2	导入一个 key 名字为 a.pub
Switch(config)# username aaa privilege 4 password abc	创建用户名为 aaa
Switch(config)# username aaa assign rsa key importKey	指定 SSH 用户名为 aaa

1.6.4 命令验证

在 SSH 客户端

- 下载 a.pri key
- 加载交换机

[root@test1 tftpboot]# ssh -i a.pri aaa@10.10.39.101

```
aaa@10.10.39.101's password:
Switch#
```

1.7 NETCONF-SSH 配置

1.7.1 概述

Netconf 功能依赖 SSH 提供的特定端口监听服务，默认 830 端口。通过控制 SSH 830 端口监听服务的开启/关闭实现 netconf 功能使能与否。

1.7.2 拓扑

图1-2 Netconf-SSH system application

1.7.3 配置

Switch# configure terminal	进入配置模式
Switch(config)# netconf ssh enable	使能 netconf 功能 ssh 监听服务
Switch(config)# netconf ssh disable	去使能 netconf 功能 ssh 监听服务

1.7.4 命令验证

router# show run | include netconf

router# config terminal

Enter configuration commands, one per line. End with CNTL/Z.

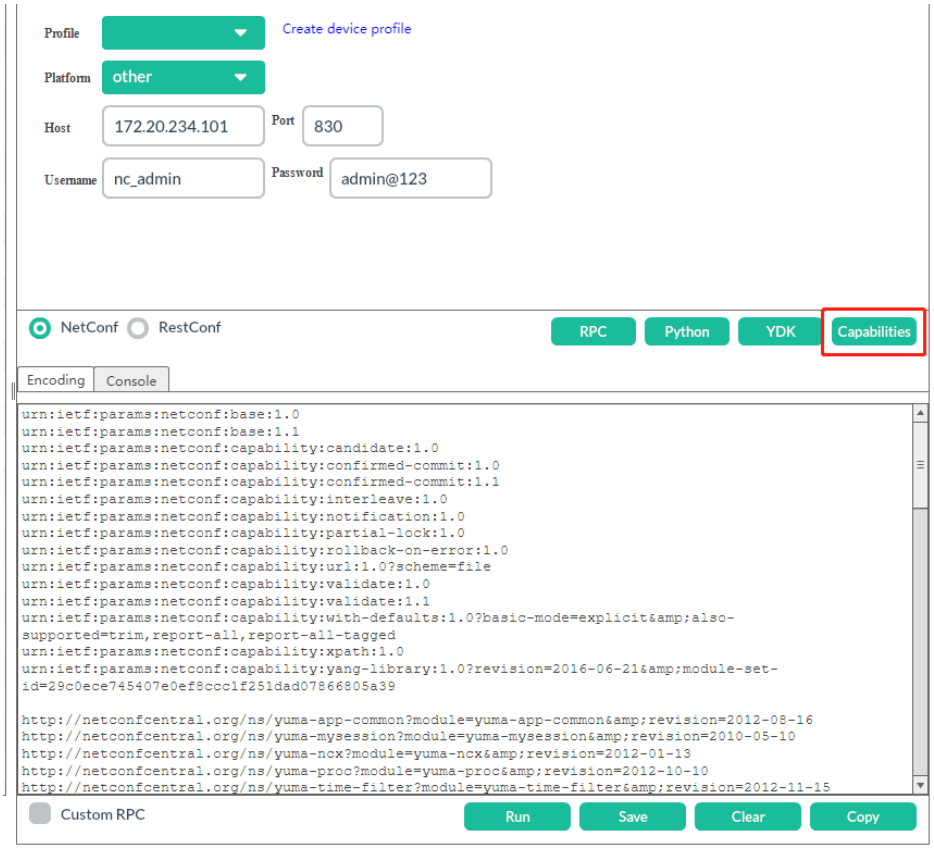
router(config)# netconf ssh enable

router(config)# exit

router# show run | include netconf

netconf ssh enable

利用第三方工具 yang-explorer 可与交换机实现 netconf 协议通信：



1.8 时间配置

1.8.1 概述

为了保证与其他设备协调工作，用户需要将系统时间设置准确。在没有其他外部时间源的情况下，您可以在系统启动后手动的设置时间和日期。如果您还有其他的同步时间的方式，比如 NTP，不建议您进行手动设置。

1.8.2 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# clock set datetime 11:30:00 10 26 2013	设置系统当前时间
Switch(config)# clock set summer-time dst date 6 1 2013 02:00:00 10 31 2013 02:00:00 120	设置夏令时时间和时区
Switch(config)#exit	退出全局配置模式

Switch# show clock detail	显示当前的时间日期信息

1.8.3 命令验证

Switch# show clock detail

```
13:31:10 dst Sat Oct 26 2013
Time zone: (GMT + 08:00:00)  beijing
Summer time starts at beijing 02:00:00  06/01/2013
Summer time ends at dst 02:00:00  10/31/2013
Summer time offset:  120 minutes
```

1.9 证书配置

1.9.1 概述

交换机的高级功能特性需要使用证书认证才可以使用，每台交换机有自己专属的证书来防止未授权的用户使用高级特性造成未知错误。一共有 3 类证书：Enterprise Base, Metro Service, and Metro Advanced。不同种类的证书包含不同的功能特性，用户可以根据需要来申请不同种类的证书。如果交换机没有证书，该交换机只能使用 L2 相关的功能特性。

不同的交换机不能共享同一份证书，为了能够获得指定交换机的证书，首先需要生成指定交换机的设备唯一标识符(UDI)， 将 UDI 发送给设备商来申请该交换机的证书。获得证书后将其应用到对应交换机上即可。

1.9.2 配置

创建 UDI

Switch# generate device identifier mgmt-if ftp://test:test@10.10.25.33/device.udi	为当前交换机创建 UDI 并且发送到 FTP 服务器上

申请证书

将 UDI 文件发送给设备商，设备商将根据客户需求来生成相应的证书并发送给客户。

使用证书

NOTE

- 必须重启交换机才能让证书生效。

- 如果交换机没有有效证书，只能使用 L2 相关功能特性
- 如果交换机有多张有效证书，则可以使用所有证书中包含的功能特性。

Switch# copy mgmt-if ftp://test:test@10.10.25.33/device.lic flash:/device.lic	将证书从 FTP 服务器拷贝到本地
Switch# reload	重启系统

1.9.3 命令验证

Switch# show license

```
License files:
========================================================================
flash:/ma.lic:
  Created Time: Fri Dec  6 17:22:23 CST 2013
  Vendor:       centec
  Customer:     centec
  Device MAC:   00:1E:08:09:03:00
  Feature Set:  QINQ MVR ERPS MEF ETHOAM
                VPWS VPLS HVPLS SMLK TPOAM
                OSPF PIM_SM IGMP VRF MPLS
                LDP BGP RSVP OSPF_TE EXTEND_ACL
                PTP BFD SSM IPV6 OSPF6
                PIM_SM6 MVR6 RIPNG TUNNEL_V6
```

2 以太网配置指导

2.1 接口配置

2.1.1 简介

以太网接口工作在 10/100/1000 Mbps 速度，可以是全双工或半双工模式。Combo 是光电复用的，用户可根据实际组网情况选择其中的一个使用，但两者不能同时工作，当激活其中的一个端口时，另一个端口就自动处于禁用状态。当 combo 端口工作在光口模式下时配置速度或双工是无效的。

2.1.2 接口状态配置

I. 配置

Switch# configure terminal	进入全局配置模式
Switch#(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# interface eth-0-2	进入接口配置模式
Switch(config-if)# shutdown	关闭接口 eth-0-2
Switch(config)# end	退出
Switch# show interface status	显示接口状态

II. 命令验证

Switch# show interface status

```
Port       Status     Duplex   Speed    Mode    Type
------------------------------------------------------------
eth-0-1    up         a-full   a-1000   access  1000BASE_T
eth-0-2    admin down auto     auto     access  1000BASE_T
```

2.1.3 接口速率配置

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# speed 100	设置接口 eth-0-1 速率为 100M.
Switch(config-if)# no shutdown	端口 UP
Switch(config-if)# interface eth-0-2	进入接口配置模式
Switch(config-if)# no shutdown	端口 UP
Switch(config-if)# speed 1000	设置接口 eth-0-2 速率为 1000M.
Switch(config-if)# interface eth-0-3	进入接口配置模式
Switch(config-if)# no shutdown	端口 UP
Switch(config-if)# speed auto	设置 eth-0-3 速率为自适应模式
Switch(config)# end	退出 exec 模式
Switch# show interface status	显示接口状态

II. 命令验证

Switch# show interface status

```
Port       Status     Duplex   Speed    Mode    Type
------------------------------------------------------------
eth-0-1    up         a-full   100      access  1000BASE_T
eth-0-2    up         a-full   1000     access  1000BASE_T
eth-0-3    up         a-full   a-1000   access  1000BASE_T
```

2.1.4 接口 Duplex 配置

设置端口的双工模式时存在三种情况：

- 当需要端口在发送数据包的同时可以接收数据包时，可以将端口设置为全双工（full）属性。
- 当需要端口同一时刻只能发送数据包或接收数据包时，可以将端口设置为半双工（half）属性。
- 当需要端口的双工属性由本端端口和对端端口自动协商决定时，可以将端口设置为自协商（auto）属性。

用户可以根据实际组网情况选择端口的双工模式。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# no shutdown	接口 up
Switch(config-if)# duplex full	设置接口 eth-0-1 为全双工.
Switch(config-if)# interface eth-0-2	进入接口配置模式
Switch(config-if)# no shutdown	接口 up
Switch(config-if)# duplex half	设置接口 eth-0-2 为半双工.
Switch(config)# interface eth-0-3	进入接口配置模式
Switch(config-if)# no shutdown	接口 up
Switch(config-if)# duplex auto	设置接口 eth-0-3 为自适应
Switch(config-if)# end	退出 exec 模式.
Switch# show interface status	显示接口状态

II. 命令验证

Switch# show interface status

```
Port       Status     Duplex   Speed    Mode    Type
------------------------------------------------------------
eth-0-1    up         full     a-1000   access  1000BASE_T
eth-0-2    up         half     a-100    access  1000BASE_T
eth-0-3    up         a-full   a-1000   access  1000BASE_T
```

2.2 Layer 3 Interface 配置

2.2.1 简介

系统支持 3 种类型的三层接口：

- VLAN 接口：你可以为想要转发路由的流量创建任意 VLAN 接口。
- 路由端口：使用 no switchport 将物理端口切换为路由端口。
- 三层 link aggregation 端口：链路聚合接口，由路由端口组成。

每一个三层接口都至少会拥有一个 IP 地址，所有三层接口都需要一个 IP 地址进行路由，此文档描述如何配置三层接口，以及如何分配一个 IP 地址到接口。

2.2.2 配置路由端口

下面步骤描述了如何配置路由端口。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# no switchport	端口设置为三层接口
Switch(config-if)# no shutdown	使能端口
Switch(config-if)# ip address 1.1.1.1/24	配置 IP 地址为 1.1.1.1/24
Switch(config-if)# end	退出 EXEC 模式
Switch# show ip interface brief	显示配置

II. 命令验证

Switch# show ip interface brief

```
Interface              IP-Address      Status                Protocol
eth-0-1                 1.1.1.1         up                      up
Switch# show ip interface
Interface eth-0-1
  Interface current state: UP
  Internet address(es):
    1.1.1.1/24 broadcast 1.1.1.255
  Joined group address(es):
    224.0.0.1
  The maximum transmit unit is 1500 bytes
  ICMP error messages limited to one every 1000 milliseconds
  ICMP redirects are always sent
ICMP unreachables are always sent
  ICMP mask replies are always sent
  ARP timeout 01:00:00,  ARP retry interval 1s
  VRRP master of:  VRRP is not configured on this interface
```

2.2.3 配置路由端口子接口

下面步骤描述了如何配置路由端口子接口。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式

Switch(config-if)# no switchport	端口设置为三层接口
Switch(config-if)# no shutdown	使能端口
Switch(config-if)# subif 5 encapsulation-dot1q 5	进入子接口模式
Switch(config-if)# ip address 11.11.11.11/24	配置子接口 IP 地址为 10.10.10.10/24
Switch(config-subif)# ip address 100.100.10.10/24 secondary	配置子接口 secondaryIP 地址为 100.100.10.10/24
Switch(config-subif)# ip vrf forwarding vpn1	配置子接口 vrf
Switch(config-subif)# exit-subif	退出子接口模式
Switch(config-if)# end	退出 EXEC 模式
Switch# show interface eth-0-1 subif 5	显示子接口

II. 命令验证

Switch# show interface eth-0-1 subif 5

```
Interface eth-0-1 subif 5
  Interface current state: UP
  Hardware is Subif, address is d886.0b00.09d5 (bia d886.0b00.09d5)
  Encapsulation-dot1q 5
  Bandwidth 1000000 kbits
  Index 16901 , Metric 1 , Encapsulation ARPA
  The maximum transmit unit (MTU) is 1500 bytes
  VRF binding: associated with vpn1
  VRRP master of :  VRRP is not configured on this interface
  ARP timeout 01:00:00,  ARP retry interval 1s
  ARP Proxy is disabled, Local ARP Proxy is disabled
```

Switch# show ip interface brief

```
Interface             IP-Address      Status                Protocol
eth-0-1 subif 5       11.11.11.11     up                    up
agg15 subif 15        15.15.15.2      up                    up
eth-0-1               unassigned      down                  down
eth-0-2               unassigned      down                  down
eth-0-3               3.3.3.1         down                  down
eth-0-4               unassigned      down                  down
eth-0-9               9.9.9.1         down                  down
eth-0-14              unassigned      up                    up
eth-0-15              unassigned      down                  down
eth-0-40              1.1.1.2         down                  down
eth-0-48              48.48.48.1      up                    up
agg15                 unassigned      up                    up
vlan1                 unassigned      down                  down
```

2.2.4 配置 VLAN Interfaces

在一个以太网接口上可以配置多个虚拟 VLAN 接口，一旦创建，任何 VLAN 接口和物理接口功能相同，它们可以像任何物理接口一样进行配置和显示。动态路由协议，如：RIP、OSPF 和 BGP，都可以在整个网络使用 VLAN 接口。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	进入 VLAN 接口配置模式
Switch(config-vlan)# vlan 10	创建 VLAN 10
Switch(config-vlan)# exit	退出 VLAN 接口配置模式
Switch(config)# interface eth-0-2	进入接口配置模式
Switch(config-if)# switchport mode trunk	设置此接口的交换机特性为 trunk 模式
Switch(config-if)# switchport trunk allowed vlan all	将此端口加入所有 VLAN
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface vlan10	进入 vlan 接口配置模式
Switch(config-if)# ip address 2.2.2.2/24	配置 IP 地址为 2.2.2.2/24
Switch(config-if)# end	退出 EXEC 模式
Switch# show ip interface brief	显示配置

II. 命令验证

Switch# show ip interface brief

```
Interface            IP-Address      Status              Protocol
vlan10               2.2.2.2         up                  up
```

Switch# show ip interface

```
Interface vlan10
  Interface current state: UP
  Internet address(es):
    2.2.2.2/24 broadcast 2.2.2.255
  Joined group address(es):
    224.0.0.1
```

```
The maximum transmit unit is 1500 bytes
ICMP error messages limited to one every 1000 milliseconds
ICMP redirects are always sent
ICMP redirects are always sent
ICMP unreachables are always sent
ICMP mask replies are always sent
ARP timeout 01:00:00,  ARP retry interval 1s
VRRP master of :  VRRP is not configured on this interface
```

2.3 接口 Errdisable 配置

2.3.1 简介

Errdisable 是一种通过关闭异常接口来保护系统的机制。如果一个接口进入 Errdisable 状态，有两种方法从 errdisabled 状态恢复。第一个方法是在 Errdisable 检测之前使能 Errdisable 恢复，接口在被设定的时间之后自动恢复。但如果 Errdisable 先发生，然后 Errdisable 恢复功能才使能，Errdisable 将不会自动恢复。第二个方法是在 Errdisable 接口上配置“no shutdown”命令。

接口链路状态的摆动抑制是一个潜在的硬件或线路问题造成的错误。管理员还可以配置接口的链路摆动抑制的检测条件。

2.3.2 配置 Errdisable 检测

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# errdisable detect reason link-flap	使能检测链路摆动抑制 errdisable
Switch(config)# end	退出 exec 模式
Switch# show errdisable detect	显示 errdisable 检测

II. 命令验证

Switch# show errdisable detect

```
ErrDisable Reason        Detection status
-----------------        ----------------
bpduguard                Enabled
bpduloop                 Enabled
link-monitor-failure     Enabled
oam-remote-failure       Enabled
port-security            Enabled
link-flap                Enabled
monitor-link             Enabled
udld                     Enabled
```

```
fdb-loop                Enabled
loopback-detection      Enabled
reload-delay            Enabled
```

2.3.3 配置 Errdisable 恢复

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# errdisable recovery reason link-flap	使能 errdisable 恢复
Switch(config)# errdisable recovery interval 30	设置恢复时间段
Switch(config)# end	退出 exec 模式
Switch# show errdisable recovery	显示 errdisable 恢复

II. 命令验证

Switch# show errdisable recovery

```
ErrDisable Reason        Timer Status
-----------------        --------------
bpduguard                Disabled
bpduloop                 Disabled
link-monitor-failure     Disabled
oam-remote-failure       Disabled
port-security            Disabled
link-flap                Enabled
udld                     Disabled
fdb-loop                 Disabled
loopback-detection       Disabled
Timer interval: 30 seconds
```

2.3.4 配置 Errdisable 摆动抑制

Switch# configure terminal	进入全局配置模式
Switch(config)# errdisable flap reason link-flap 20 60	设置 link flap 条件是每分钟 20 次
Switch(config)# end	退出全局配置模式
Switch# show errdisable flap	显示 errdisable flap

```
Switch# show errdisable flap
ErrDisable Reason        Flaps     Time (sec)
-----------------        ------    ----------
```

```
link-flap               20        60
```

2.3.5 配置关闭端口进入 Errdisable 状态功能

管理员可以通过该命令来控制在端口发生 mac flap 的时候是否进入 errdisable 状态

Switch(config-if)#no errdisable	端口不进入 errdisable 状态
Switch(config-if)#errdisable	端口进入 errdisable 状态

2.3.6 检查 Errdisable 状态

管理员可以通过两种命令来检查接口 errdisable 状态，具体参照下面表格中的命令和配置说明。

Switch# show errdisable recovery	显示 errdisable 回复
Switch# show interface status	显示 interface 状态

如果 errdisable 使能恢复，命令行将显示恢复所剩时间，否则将显示没有恢复。

例子 1：使能链路摆动抑制 errdisable

```
Switch# show errdisable recovery
ErrDisable Reason        Timer Status
-----------------        --------------
bpduguard                Disabled
bpduloop                 Disabled
link-monitor-failure     Disabled
oam-remote-failure       Disabled
port-security            Disabled
link-flap                Enabled
udld                     Disabled
fdb-loop                 Disabled
loopback-detection       Disabled
Timer interval: 300 seconds
Interfaces that will be enabled at the next timeout:
Interface Errdisable Reason Time Left(sec)
--------- ----------------- --------------
eth-0-3   link-flap         25
```

例子 2：链路摆动抑制 errdisable 去使能

Switch# show errdisable recovery

```
ErrDisable Reason        Timer Status
-----------------        --------------
```

```
bpduguard                Disabled
bpduloop                 Disabled
link-monitor-failure     Disabled
oam-remote-failure       Disabled
port-security            Disabled
link-flap                Disabled
udld                     Disabled
fdb-loop                 Disabled
loopback-detection       Disabled
Timer interval: 300 seconds
```

接口状态命令也显示简短的信息来表示接口的 errdisable 状态。

Switch# show interface status

```
Port       Status     Duplex   Speed    Mode    Type         Description
--------------------------------------------------------------------------------
eth-0-1    up         a-full   a-1000   TRUNK   1000BASE_SX
eth-0-2    down       auto     auto     TRUNK   Unknown
eth-0-3    errdisable a-full   a-1000   TRUNK   1000BASE_SX
eth-0-4    down       auto     auto     ACCESS  Unknown
```

2.4 MAC 表配置

2.4.1 简介

MAC 地址表中包含交换机端口之间转发流量的地址信息。地址表包括地址类型如下：

- 动态地址：根据交换机的源地址学习，如果该地址在老化时间后未学习到，进入老化状态。我们只支持 IVL 的学习模式。
- 静态地址：由管理员手动添加源地址。

2.4.2 参考

IEEE 802.1D

IEEE 802.1Q

2.4.3 术语

以下是用来形容 MAC 地址表中的术语和概念的简要描述：

IVL：独立 VLAN 学习：对于一个给定的 VLAN，如果某个特定的 MAC 地址是在一个 VLAN 学习的，它不能被作为任何其他 VLAN 地址转发决策。

SVL：共享 VLAN 学习：对于一个给定的 VLAN，如果某个特定的 MAC 地址是在一个 VLAN 中学习的，它可以作为任何其他 VLAN 地址转发决策。

2.4.4 地址老化时间配置

老化时间不是精确的时间。如果老化时间设置为 N，动态地址将在 N - 2N 间隔后老化。默认的老化时间为 300 秒。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# mac-address-table ageing-time 10	设置动态地址老化时间为 10 秒
Switch(config)# end	退出至 EXEC 模式
Switch# show mac address-table ageing-time	显示地址老化时间

II. 命令验证

Switch# show mac address-table ageing-time

```
MAC address table ageing time is 10 seconds
```

2.4.5 静态单播地址配置

单播地址表只能绑定在一个端口上。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# mac-address-table 0000.1234.5678 forward eth-0-1 vlan 1	添加静态单播地址
Switch(config)# end	退出至 EXEC 模式
Switch# show mac address-table	显示 MAC 地址表

II. 命令验证

Switch# show mac address-table

```
Mac Address Table
-------------------------------------------
(*) - Security Entry
Vlan    Mac Address      Type      Ports
----   --------------   -------   --------
1       0000.1234.5678   static    eth-0-1
```

2.4.6 静态组播地址配置

组播地址可以绑定在多个端口上。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# mac-address-table 0100.0000.0000 forward eth-0-1 vlan 1	在接口 eth-0-1 添加静态组播地址
Switch(config)# mac-address-table 0100.0000.0000 forward eth-0-2 vlan 1	在接口 eth-0-2 添加静态组播地址
Switch(config)# end	退出至 EXEC 模式
Switch# show mac address-table	显示 MAC 地址表

II. 命令验证

Switch# show mac address-table

```
   Mac Address Table
------------------------------------------
(*) - Security Entry
Vlan    Mac Address       Type        Ports
----   --------------    -------     -------
1      0100.0000.0000    static      eth-0-1
                                     eth-0-2
```

2.4.7 MAC 地址过滤配置

MAC 过滤会丢弃那些源或目的地址设置为丢弃数据帧。 AC 过滤的优先级高于 MAC 地址。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# mac-address-table 0000.1234.5678 discard	添加单播地址被丢弃
Switch(config)# end	退出至 EXEC 模式
Switch# show mac-filter address-table	显示 MAC 地址表

II. 命令验证

Switch# show mac-filter address-table

```
    MAC Filter Address Table
----------------------------------
 Current count       : 0
 Max count           : 128
 Left count          : 128
 Filter address list :
----------------------------------
```

2.5 VLAN 配置

2.5.1 简介

VLAN（虚拟局域网）是一个逻辑上分割成不同广播域的网络，使数据包只能在被指定为同一个 VLAN 的端口之间进行交换。每个 VLAN 都被视为一个逻辑网络，目的地不属于同一个 VLAN 的数据包必须通过路由转发。

2.5.2 配置 Access 端口

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config-vlan)# vlan 2	创建 VLAN 2
Switch(config-vlan)# exit	退出 VLAN 模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# switchport mode access	设置接口类型为 Access
Switch(config-if)# switchport access vlan 2	指定端口到相应的 VLAN
Switch(config-if)# end	退出配置模式
Switch# show vlan brief	显示 VLAN 的简要配置信息
Switch# show interface switchport interface eth-0-1	显示交换机的接口信息

II. 命令验证

```
Switch# show interface switchport interface eth-0-1
 Interface name          : eth-0-1
 Switchport mode         : access
 Ingress filter          : enable
 Acceptable frame types  : vlan-untagged only
 Default Vlan            : 2
```

```
 Configured Vlans        :    2
Switch# show vlan brief
VLAN ID  Name            State   STP ID      Member ports
                                                 (u)-Untagged, (t)-Tagged
======= ================ ======= =======  ========================
1       default          ACTIVE  0       eth-0-2(u) eth-0-3(u)
                                                 eth-0-4(u) eth-0-5(u)
                                                 eth-0-6(u) eth-0-7(u)
                                                 eth-0-8(u) eth-0-9(u)
                                                 eth-0-10(u) eth-0-11(u)
                                                 eth-0-12(u) eth-0-13(u)
                                                 eth-0-14(u) eth-0-15(u)
                                                 eth-0-16(u) eth-0-17(u)
                                                 eth-0-18(u) eth-0-19(u)
                                                 eth-0-20(u) eth-0-21(u)
                                                 eth-0-22(u) eth-0-23(u)
2       VLAN0002         ACTIVE  0       eth-0-1(u)
```

2.5.3 Trunk 端口配置

Trunk 端口能接收标记、无标记的、优先级标记的帧，并发送未标记和标记的帧。如果端口收到一个未标记的帧，此帧将分配端口的 PVID 为 VLAN ID；如果一个帧的 VID 与端口的 PVID 相等，此帧发送时会剥掉 VLAN 标签。

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config-vlan)# vlan 10,20	创建 VLAN10，20
Switch(config-vlan)# exit	退出 VLAN 模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# switchport mode trunk	设置端口为 Trunk 模式
Switch(config-if)# switchport trunk allowed vlan all	设置端口允许所有的 VLAN 通过
Switch(config-if)# switchport trunk native vlan 10	设置端口的本地 VLAN 为 10
Switch(config-if)# exit	退出接口模式

II. 命令验证

Switch# show interface switchport

```
 Interface name          : eth-0-1
 Switchport mode         : trunk
 Ingress filter          : enable
 Acceptable frame types  : all
 Default Vlan            : 10
 Configured Vlans        :    1   10   20
 Interface name          : eth-0-2
 Switchport mode         : access
 Ingress filter          : enable
 Acceptable frame types  : vlan-untagged only
 Default Vlan            : 10
 Configured Vlans        :   10
Switch# show vlan brief
VLAN ID  Name            State   STP ID      Member ports
                                                 (u)-Untagged, (t)-Tagged
======= ================ ======= =======  ========================
1       default          ACTIVE  0       eth-0-1(t) eth-0-3(u)
                                                 eth-0-4(u) eth-0-5(u)
                                                 eth-0-6(u) eth-0-7(u)
                                                 eth-0-8(u) eth-0-9(u)
                                                 eth-0-10(u) eth-0-11(u)
                                                 eth-0-12(u) eth-0-13(u)
                                                 eth-0-14(u) eth-0-15(u)
                                                 eth-0-16(u) eth-0-17(u)
                                                 eth-0-18(u) eth-0-19(u)
                                                 eth-0-20(u) eth-0-21(u)
                                                 eth-0-22(u) eth-0-23(u)
10      VLAN0010         ACTIVE  0       eth-0-1(t) eth-0-2(u)
20      VLAN0020         ACTIVE  0       eth-0-1(t)
```

2.6 VOICE VLAN 配置

2.6.1 简介

随着语音技术的日益发展，IP 电话、IAD（Integrated Access Device，综合接入设备）应用越来越广泛，尤其在宽带小区，网络中经常同时存在语音数据和业务数据两种流量。语音数据在传输时需要具有比业务数据更高的优先级，以减少传输过程中可能产生的时延和丢包现象。

提高语音数据传输优先级的传统处理方法是使用 ACL 对语音数据进行区分，并使用 QoS 保证传输质量。为简化用户配置、更方便的管理语音流的传输策略，设备机提供了 Voice VLAN 功能。Voice VLAN 的主要特点就是可以通过报文的源 MAC 地址自动识别出语音流量，保证语音流量传输。

2.6.2 配置 VOICE VLAN

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config-vlan)# vlan 2	创建 VLAN2
Switch(config-vlan)# exit	退出 VLAN 模式，
Switch(config)# voice vlan 2	将 VLAN 指定为 VOICE VLAN
Switch(config)# voice vlan mac-address 0055.0000.0000 ffff.ff00.0000 description test	添加一条 entry 指定某类 MAC 为 VOICE VLAN
Switch(config)# interface eth-0-1	进入端口 eth-0-1 的配置模式
Switch(config-if)# switchport mode trunk	将端口设置为 truck 口
Switch(config-if)# switchport trunk allowed vlan all	允许端口通过所有带 tag 的 VLAN
Switch(config-if)# voice vlan enable	在端口上启用 VLOICE VLAN
Switch(config-if)# interface eth-0-2	进入端口 eth-0-2 的配置模式
Switch(config-if)# switchport mode trunk	将端口设置为 truck 口
Switch(config-if)# switchport trunk allowed vlan all	允许端口通过所有带 tag 的 VLAN

2.6.3 命令验证

向 eth-0-1 发送报文，报文格式为：

```
0x0000:  0000 0a02 0001 0055 0000 0011 8100 0002  .......k.......
0x0010:  0800 aadd aadd aadd aadd aadd aadd aadd  ................
0x0020:  aadd aadd aadd aadd aadd aadd aadd aadd  ................
0x0030:  aadd aadd aadd aadd aadd aadd            ............
```

Vlan tag 字段当中的 priority 为 0

在 eth-0-2 接收报文，收到报文的格式为：

```
0x0000:  0000 0a02 0001 0055 0000 0011 8100 a002  .......k.......
0x0010:  0800 aadd aadd aadd aadd aadd aadd aadd  ................
0x0020:  aadd aadd aadd aadd aadd aadd aadd aadd  ................
0x0030:  aadd aadd aadd aadd aadd aadd            ............
```

计算之后可以看出其中 vlan tag 字段当中的 priority 被修改为了 5。

2.7 VLAN Classification 配置

2.7.1 概述

VLAN 分类是基于协议或子网标准的具体规则将数据包发送到选定的 VLAN。每一个接口可以应用一种规则集。

VLAN 分类规则有 3 种类型：基于 MAC、基于 IP 和基于协议的分类。基于 MAC 的 VLAN 分类规则是根据传入数据包的源 MAC 地址将数据包进行分类；基于 IP 的 VLAN 分类规则将根据传入数据包的源 IP 地址进行分类；基于协议的 VLAN 分类规则将根据数据包的三层协议类型进行分类，以下三层类型可以支持 ARP、IP（V4）、MPLS、MCAST MPLS、PPPoE 协议和 RARP。

不同类型的 VLAN 分类规则，可以添加到同一 VLAN 的分类组。只有一个 VLAN 分类规则可以在一个交换机端口生效。

2.7.2 拓扑

在下面配置的例子中，创建三个 VLAN 分类规则：

- 第 1 条是基于 MAC 的规则，它将源 MAC 2222.2222.2222 分类到 VLAN 5；
- 第 2 条是基于 IP 的规则，它将源 IP 1.1.1.1 分类到 VLAN 5；
- 第 3 条是基于协议的规则，它将 ARP 的协议数据包分类到 VLAN 5。

把规则 1，2，3 加入到组 31，并且在三个接口上应用组 31。在 eth-0-1、eth-0-2 和 eth-0-3 三个接口上应用不同的分类策略。eth-0-1 基于 IP 分类，意味着匹配 IP 的数据包在这个接口上将转发到规则对应的 VLAN。eth-0-2 基于 MAC 分类，意味着匹配 MAC 地址的数据包将转发到规则对应的 VLAN。eth-0-3 基于协议的分类，意味着匹配协议的数据包将转发到规则对应的 VLAN。

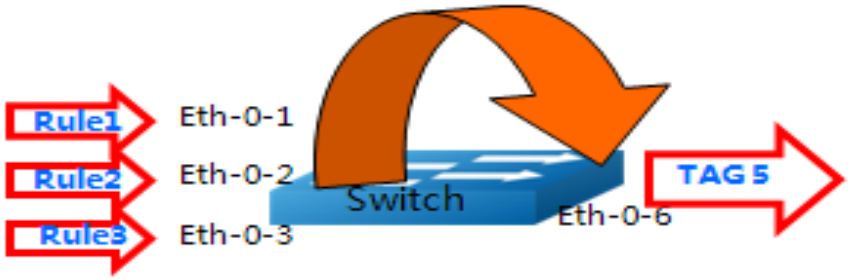


图2-1 VLAN Classification 拓扑图

2.7.3 配置

VLAN Classification 配置细节

“show vlan classifier group”显示所有的 VLAN 分类组；“show vlan classifier rule”显示所有的 VLAN 分类规则。

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config-vlan)# vlan 5	创建 VLAN5
Switch(config-vlan)# vlan 6	创建 VLAN6
Switch(config-vlan)# exit	退出 VLAN 模式
Switch(config)# vlan classifier rule 1 mac 2222.2222.2222 vlan 5	创建基于 MAC 的 VLAN 分类规则
Switch(config)# vlan classifier rule 2 ip 1.1.1.1 vlan 5	创建基于 IP 的 VLAN 分类规则
Switch(config)# vlan classifier rule 3 protocol arp vlan 5	创建基于协议的 VLAN 分类规则
Switch(config)# vlan classifier group 31 add rule 1	把规则 1 加入到组 31
Switch(config)# vlan classifier group 31 add rule 2	把规则 2 加入到组 31
Switch(config)# vlan classifier group 31 add rule 3	把规则 3 加入到组 31

Interface 配置细节

“show vlan classifier interface group”显示接口上配置的 VLAN 分类信息。

Switch# configure terminal	进入配置模式

Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# switchport access vlan 6	将 PVID 6 指定给 eth-0-1
Switch(config-if)# switchport access allowed vlan add 5	接口上允许 VLAN5
Switch(config-if)# vlan classifier activate 31 based ip	接口上应用组 31 并且设置接口 VLAN 分类类型为基于 IP 分类
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# switchport access vlan 6	将 PVID 6 指定给 eth-0-2
Switch(config-if)# switchport access allowed vlan add 5	接口上允许 VLAN5
Switch(config-if)# vlan classifier activate 31 based mac	接口上应用组 31 并且设置接口 VLAN 分类类型为基于 mac 分类
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-3	进入接口模式
Switch(config-if)# switchport access vlan 6	将 PVID 6 指定给 eth-0-3
Switch(config-if)# switchport access allowed vlan add 5	接口上允许 VLAN5
Switch(config-if)# vlan classifier activate 31 based protocol	接口上应用组 31 并且设置接口 VLAN 分类类型为基于协议分类
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-6	进入接口模式
Switch(config)#switchport mode trunk	配置端口为 Trunk 模式
Switch(config-if)# switchport trunk allowed vlan add 5	将 eth-0-6 加入到 vlan5 中
Switch(config-if)# exit	退出接口模式

2.7.4 命令验证

步骤 1 验证 VLAN 分类规则。

Switch# show vlan classifier rule

```
vlan classifier rule 1 mac 2222.2222.2222 vlan 5
vlan classifier rule 2 ip 1.1.1.1 vlan 5
vlan classifier rule 3 protocol arp vlan 5
```

步骤 2 验证 VLAN 分类规则组。

Switch# show vlan classifier group

```
vlan classifier group 31 add rule 1
vlan classifier group 31 add rule 2
vlan classifier group 31 add rule 3
```

步骤 3 验证 VLAN 分类规则接口应用。

Switch# show vlan classifier interface grou

```
vlan classifier group 31 on interface eth-0-2, based mac
vlan classifier group 31 on interface eth-0-1, based ip
vlan classifier group 31 on interface eth-0-3, based protocol
```

2.8 VLAN Mapping 配置

2.8.1 配置 VLAN 转换

I. 概述

服务供应商的业务客户往往有特定的 VLAN ID 的要求，同一个网络服务提供商的不同客户的要求的 VLAN 可能会重叠，并且通过服务商设备的用户流量也可能会混合。通过给每一个客户分配不同的 VLAN ID 来映射自己的 VLAN ID，能够将不同应用的客户的通讯相分开。

使用 VLAN 转换功能，服务提供商可以使用一系列的 VLAN 来服务拥有自己 VLAN ID 的客户。客户 VLAN ID 被转换，来自不同应用的客户的流量在服务提供商的设备上被分开，甚至当他们出现在同一 VLAN。

II. 拓扑

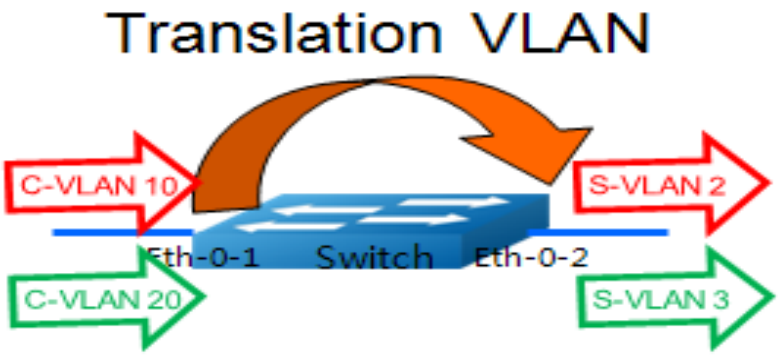


图2-2 VLAN 转换拓扑图

III. 配置

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 配置模式
Switch(config-vlan)# vlan 2,3	创建 S-TAG VLAN 2,3
Switch(config)# ethernet evc evc_c1	创建 EVC evc_c1
Switch(config-evc)# dot1q mapped-vlan 2	设置 VLAN2 关联到 EVE evc_c1
Switch(config)# ethernet evc evc_c2	创建 EVC evc_c2
Switch(config-evc)# dot1q mapped-vlan 3	设置 VLAN3 关联到 EVE evc_c2
Switch(config)# vlan mapping table vm	创建 VLAN MAPPING 表 VM
Switch(config-vlan-mapping)# raw-vlan 10 evc evc_c1	设置 C-Tag 为 10 映射到 S-Tag 为 2
Switch(config-vlan-mapping)# raw-vlan 20 evc evc_c2	设置 C-Tag 为 20 映射到 S-Tag 为 3
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# switchport mode trunk	配置端口为 Trunk 模式
Switch(config-if)# switchport trunk vlan-translation	设置 Trunk 模式为 VLAN 转换模式
Switch(config-if)# switchport trunk vlan-	在接口上应用 VLAN MAPPING 表

translation mapping table vm	
Switch(config-if)# switchport trunk allowed vlan add 2,3	加入 VLAN2,3
Switch(config-if)# interface eth-0-2	进入接口模式
Switch(config-if)# switchport mode trunk	设置端口为 Trunk 模式
Switch(config-if)# switchport trunk allowed vlan add 2,3	加入 VLAN2，3
Switch(config-if)# end	退出接口模式
Switch# show interface switchport interface eth-0-1	检查端口配置
Switch# show vlan mapping table	检查 VLAN mapping 表

IV. 命令验证

Switch# show interface switchport interface eth-0-1

```
Interface name          : eth-0-1
 Switchport mode         : trunk
 VLAN traslation         : enable
 VLAN mapping table      : vm
 Ingress filter          : enable
 Acceptable frame types  : all
 Default Vlan            : 1
 Configured Vlans        : 1    2    3
Switch# show vlan mapping table
Table Name        EVC Name          Mapped VLAN Raw VLAN
 ================ ================  =========== ======================
vm                evc_c1            2           10
                  evc_c2            3           20
```

2.8.2 配置 802.1Q Tunneling

I. 简介

QinQ 技术通过在以太帧中堆叠两个 802.1Q 包头，有效地扩展了 VLAN 数目，使 VLAN 的数目最多可达 4096 × 4096 个。同时，多个 VLAN 能够被复用到一个核心 VLAN 中。ISP 通常为每个客户建立一个 VLAN 模型，用通用属性注册协议/通用 VLAN 注册协议（GARP/GVRP）自动监控整个主干网络的 VLAN，并通过扩展生成树协议（STP）来加快网络收敛速度，从而为网络提供弹性。

QinQ 技术作为初始的解决方案是不错的，但随着用户数量的增加，SVLAN 模型也会带来可扩展性的问题。因为有些用户可能希望在分支机构间进行数据传输时可以携带

自己的 VLAN ID，这就使采用 QinQ 技术的 MSP 面临以下两个问题：第一，第一名客户的 VLAN 标识可能与其他客户冲突；第二，服务提供商将受到客户可使用标识数量的严重限制。如果允许用户按他们自己的方式使用各自的 VLAN ID 空间，那么核心网络仍存在 4096 个 VLAN 的限制。

QinQ 是指将用户私网 VLAN tag 封装在公网 VLAN tag 中，使报文带着两层 VLAN tag 穿越运营商的骨干网络（公网）。在公网中报文只根据外层 VLAN tag（即公网 VLAN tag）传播，用户的私网 VLAN tag 被屏蔽。这样，不仅对数据流进行了区分，而且由于私网 VLAN 标签被透明传送，不同的用户 VLAN 标签可以重复使用，只需要外层 VLAN 标签的在公网上唯一即可，实际上也扩大了可利用的 VLAN 标签数量。

封装外层 VLAN 标签有两种方法，一种是标准 QINQ 封装，即基于端口打外层标签的，该端口下所有的用户数据统一封装一个共同的 VLAN 标签，在实际应用中局限性太大，另外一种是灵活 QINQ 封装方法，既可以根据一些特性对用户数据进行流分类，然后不同的类别封装不同的外层 VLAN 标签。

II. 配置基本 QINQ

拓扑

图2-3 基本 802.1Q 隧道拓扑图

配置

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# switchport mode dot1q-tunnel	配置接口为 DOT1Q-Tunnel 模式
Switch(config-if)# end	退出接口模式
Switch# show interface switchport interface eth-0-1	检查接口的配置

命令验证

这个例子显示了如何配置一个基本的 802.1q 隧道口的交换机端口。可用显示命令显示接口上的配置。

Switch# show interface switchport interface eth-0-1

```
Interface name           : eth-0-1
 Switchport mode         : dot1q-tunnel(basic)
 Ingress filter          : enable
 Acceptable frame types  : all
 Default Vlan            : 1
 Configured Vlans        : 1
```

I. 配置灵活 QINQ

U-tag 报文加一层 TAG 的配置步骤如下所示。

拓扑

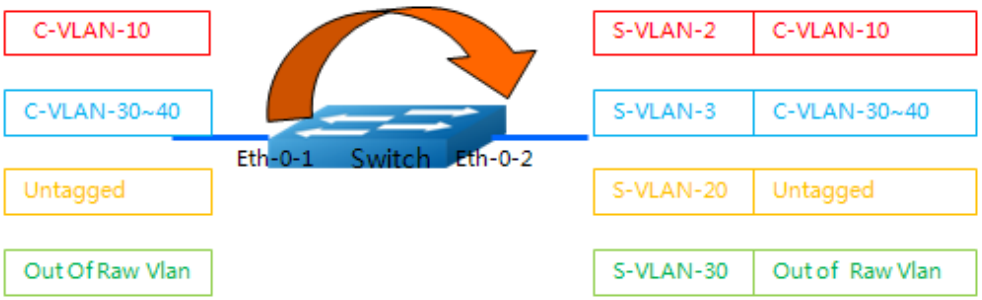


图2-4 加一层 TAG

配置

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config-vlan)# vlan 2,3,20,30	创建 S-TAG 2，3，20，30
Switch(config)# ethernet evc evc_c1	创建 EVC evc_c1
Switch(config-evc)# dot1q mapped-vlan 2	设置 S-TAG 2 关联 EVC_C1
Switch(config)# ethernet evc evc_c2	创建 EVC evc_c2
Switch(config-evc)# dot1q mapped-vlan 3	设置 S-TAG 3 关联 EVC_C2
Switch(config)# ethernet evc evc_c3	创建 EVC evc_c3
Switch(config-evc)# dot1q mapped-vlan 20	设置 S-TAG 20 关联 EVC_C3

Switch(config)# ethernet evc evc_c4	创建 EVC evc_c4
Switch(config-evc)# dot1q mapped-vlan 30	设置 S-TAG 30 关联 EVC_C4
Switch(config)# vlan mapping table vm	创建 VLAN mapping 表
Switch(config-vlan-mapping)# raw-vlan 10 evc evc_c1	设置 C-TAG10 加入 evc_c1
Switch(config-vlan-mapping)# raw-vlan 30-40 evc evc_c2	设置 C-TAG30-40 加入 evc_c2
Switch(config-vlan-mapping)# raw-vlan untagged evc evc_c3	设置 U-TAG 加入 evc_c3
Switch(config-vlan-mapping)# raw-vlan out-of-range evc evc_c4	设置范围之外的 C-TAG 加入 evc_c4
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# switchport mode dot1q-tunnel	设置接口为 Dot1q-tunnel 模式
Switch(config-if)# switchport dot1q-tunnel type selective	设置接口为灵活 QINQ
Switch(config-if)# switchport dot1q-tunnel vlan mapping table vm	在接口上应用 VLAN mapping 表 VM
Switch(config-if)# switchport dot1q-tunnel allowed vlan add 2,3,20,30	接口上允许 VLAN 2，3，20，30
Switch(config-if)# interface eth-0-2	进入接口模式
Switch(config-if)# switchport mode trunk	设置接口为 Trunk 模式
Switch(config-if)# switchport trunk allowed vlan add 2,3,20,30	接口上允许 VLAN 2，3，20，30
Switch(config-if)# end	退出接口模式
Switch# show interface switchport interface eth-0-1	检查端口配置
Switch# show vlan mapping table	检查 VLAN mapping 表

命令验证

这个例子显示了如何配置一个灵活 QINQ。可用显示命令显示接口上的配置。

注意：如果 eth-0-1 的 tpid 和 eth-0-2 的 tpid 不同，用户需要全局启用 qos 并且将 eth-0-2 设置为 replace cos 来替换从 eth-0-2 出去的 stag 的 tpid。

Switch# show interface switchport interface eth-0-1

```
 Interface name          : eth-0-1
 Switchport mode         : dot1q-tunnel(selective)
 VLAN mapping table      : vm
 Ingress filter          : enable
 Acceptable frame types  : all
 Default Vlan            : 1
 Configured Vlans        : 1    2    3    20   30
Switch# show vlan mapping table
Table Name       EVC Name          Mapped VLAN Raw VLAN
 ================ ================ =========== ==========================
 vm               evc_c1           2           10
                  evc_c2           3           30-40
                  evc_c3           20          untagged
                  evc_c4           30          out-of-range
```

U-tag 报文加两层 TAG 的配置步骤如下所示。

拓扑

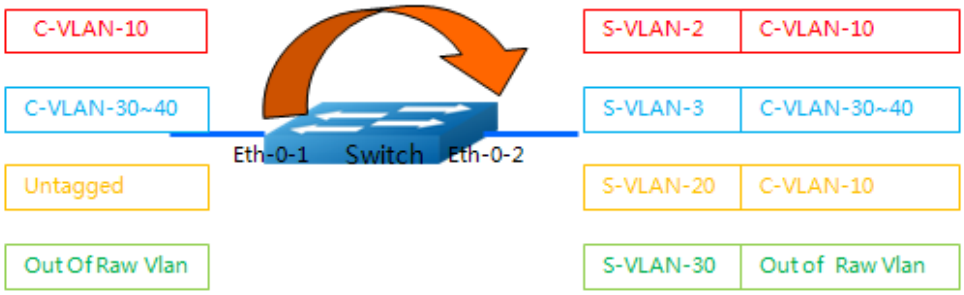


图2-5 加两层 TAG

配置

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config-vlan)# vlan 2,3,10,20,30	创建 S-TAG 2，3，10，20，30
Switch(config)# ethernet evc evc_c1	创建 EVC evc_c1
Switch(config-evc)# dot1q mapped-vlan 2	设置 S-TAG 2 关联 evc_c1
Switch(config-evc)# exit	退出 EVC 模式
Switch(config)# ethernet evc evc_c2	创建 EVC evc_c2

Switch(config-evc)# dot1q mapped-vlan 3	设置 S-TAG 3 关联 evc_c2
Switch(config-evc)# exit	退出 EVC 模式
Switch(config)# ethernet evc evc_c3	创建 EVC evc_c3
Switch(config-evc)# dot1q mapped-double-vlan 10 20	设置 C-TAG 10, S-TAG 20 关联 evc_c3
Switch(config-evc)# exit	退出 EVC 模式
Switch(config)# ethernet evc evc_c4	创建 EVC
Switch(config-evc)# dot1q mapped-vlan 30	设置 S-TAG 30 关联 evc_c4
Switch(config-evc)# exit	退出 EVC 模式
Switch(config)# vlan mapping table vm	创建 VLAN mapping 表
Switch(config-vlan-mapping)# raw-vlan 10 evc evc_c1	设置 C-TAG10 加入 evc_c1
Switch(config-vlan-mapping)# raw-vlan 30-40 evc evc_c2	设置 C-TAG30-40 加入 evc_c2
Switch(config-vlan-mapping)# raw-vlan untagged evc evc_c3	设置 U-TAG 加入 evc_c3
Switch(config-vlan-mapping)# raw-vlan out-of-range evc evc_c4	设置范围之外的 C-TAG 加入 evc_c4
Switch(config-vlan-mapping)# exit	退出 mapping 配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# switchport mode dot1q-tunnel	配置接口为 do1q-tunnel
Switch(config-if)# switchport dot1q-tunnel type selective	设置端口为灵活 QINQ
Switch(config-if)# switchport dot1q-tunnel vlan mapping table vm	在接口上应用 vlan mapping 表 vm
Switch(config-if)# switchport dot1q-tunnel native inner-vlan 10	配置 native inner-vlan 10
Switch(config-if)# switchport dot1q-tunnel allowed vlan add 2,3,20,30	接口上允许 VLAN 2，3，20，30

Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# switchport mode trunk	设置接口为 Trunk 模式
Switch(config-if)# switchport trunk allowed vlan add 2,3,20,30	接口上允许 VLAN 2，3，20，30
Switch(config-if)# end	退出接口模式
Switch# show interface switchport interface eth-0-1	检查配置
Switch# show vlan mapping table	检查 vlan mapping 表项

命令验证

这个例子显示了如何配置一个灵活 QINQ。可用显示命令显示接口上的配置。

Switch# show interface switchport interface eth-0-1

```
 Interface name          : eth-0-1
 Switchport mode         : dot1q-tunnel(selective)
 VLAN mapping table      : vm
 Ingress filter          : enable
 Acceptable frame types  : all
 Default Vlan            : 10
 Configured Vlans        : 1    2    3    20   30
Switch# show vlan mapping table
 Table Name       EVC Name          Mapped VLAN Raw VLAN
 ================ ================ =========== ================================
 vm               evc_c1           2           10
                  evc_c2           3           30-40
                  evc_c3           20(10)      untagged
                  evc_c4           30          out-of-range
```

2.9 Link Aggregation 配置

2.9.1 简介

本章包含了一个链路聚合控制协议（LACP）配置示例。LACP 协议是基于 802.3ad 的 IEEE 规范。它允许多个物理接口的捆绑，形成一个单一的逻辑通道，提供增强的性能和冗余。聚合接口被视为单一链路接到每个交换机上。生成树将它视为一个接口。当有一个物理接口出现故障，其他接口正常连接，链路不会中断。此实现在单一的逻辑

通道上支持最多 16 个物理以太网链路。 LACP 协议使我们的设备可以管理与符合 802.3ad 的协议的其他设备之间的链接聚合组。使用 LACP 协议，交换机学习支持 LACP 成员识别的每个端口的能力。然后，具有相同的属性动态组端口捆绑到一个单一逻辑链路。

2.9.2 参考

LACP 基于 IEEE 802.3ad 标准协议

2.9.3 配置动态 AGG

I. 拓扑

本例中，两个交换机 S1 和 S2 之间配置三条链路。这三个链路都分配在同一个管理中心（1），使他们能聚合形成一个单一的通道 1。

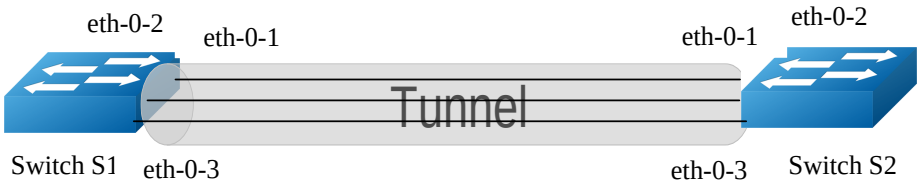


图2-6 LACP 拓扑

II. 配置

Switch 1

Switch1# configure terminal	进入全局配置模式
Switch1(config)# lacp system-priority 2000	在此交换机设置系统优先级。这个优先级用于决定系统解决在选择聚合组中的冲突。数值越低，优先级越高
Switch1(config)# port-channel load-balance hash-field-select macsa	通过源 MAC 地址实现负载均衡
Switch1(config)# interface eth-0-1	进入接口配置模式
Switch1(config-if)# no shutdown	端口 up
Switch1(config-if)# channel-group 1 mode active	添加接口到 channel group 1 启用链路聚合，因此可以通过本地系统选择聚合

Switch1(config-if)# exit	退出接口配置模式
Switch1(config)# interface eth-0-2	进入接口配置模式
Switch1(config-if)# channel-group 1 mode active	添加接口到 channel group 1 启用链路聚合，模式为主动模式，因此可以通过本地系统选择聚合
Switch1(config-if)# no shutdown	端口 up
Switch1(config-if)# exit	退出接口配置模式
Switch1(config)# interface eth-0-3	进入接口配置模式
Switch1(config-if)# channel-group 1 mode active	添加接口到 channel group 1 启用链路聚合，因此可以通过本地系统选择聚合
Switch1(config-if)# no shutdown	端口 up
Switch1(config-if)# end	退出接口配置模式

Switch 2

Switch2# configure terminal	进入全局配置模式
Switch2(config)# lacp system-priority 1000	设置交换机的系统优先级，这个优先级用于决定系统解决在选择聚合组中的冲突。数值越低，优先级越高
Switch2(config)# interface eth-0-1	进入接口配置模式
Switch2(config-if)# no shutdown	端口 up
Switch2(config-if)# channel-group 1 mode active	添加接口到 channel group 1 启用链路聚合，模式为主动模式，因此可以通过本地系统选择聚合
Switch2(config-if)# exit	退出接口配置模式
Switch2(config)# interface eth-0-2	进入接口配置模式
Switch2(config-if)# channel-group 1 mode active	添加接口到 channel group 1 启用链路聚合，因此可以通过本地系统选择聚合

Switch2(config-if)# no shutdown	端口 up
Switch2(config-if)# exit	退出接口配置模式
Switch2(config)# interface eth-0-3	进入接口配置模式
Switch2(config-if)# channel-group 1 mode active	添加接口到 channel group 1 启用链路聚合，模式为主动模式，因此可以通过本地系统选择聚合
Switch2(config-if)# no shutdown	端口 up
Switch2(config-if)# end	退出接口配置模式

I. 命令验证

Switch1# show channel-group summary

```
port-channel load-balance hash-arithmetic: xor
port-channel load-balance hash-field-select:
     macsa
Flags:  s - suspend           T - standby
        D - down/admin down   B - in Bundle
        R - Layer3            S - Layer2
        w - wait              U - in use
Mode:   SLB  - static load balance
        DLB  - dynamic load balance
        SHLB - self-healing load balance
        RR   - round robin load balance
Aggregator Name  Mode      Protocol     Ports
---------------+---------+------------+-----------------------------------------
----
agg1(SU)         SLB       LACP         eth-0-1(B) eth-0-2(B) eth-0-3(B)
Switch1# show interface agg1
Interface agg1
  Interface current state: UP
  Hardware is AGGREGATE, address is cce3.33fc.330b (bia cce3.33fc.330b)
  Bandwidth 3000000 kbits
  Index 1025 , Metric 1 , Encapsulation ARPA
  Speed - 1000Mb/s , Duplex - Full  , Media type is Aggregation
  Link speed type is autonegotiation, Link duplex type is autonegotiation
  Input flow-control is off, output flow-control is off
  The Maximum Frame Size is 1534 bytes
  VRF binding: not bound
  Label switching is disabled
  No virtual circuit configured
  ARP timeout 01:00:00,  ARP retry interval 1s
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 2 bits/sec, 0 packets/sec
    13 packets input, 1184 bytes
    Received 0 unicast, 0 broadcast, 0 multicast
    0 runts, 0 giants, 0 input errors, 0 CRC
```

```
0 frame, 0 overrun, 0 pause input
0 input packets with dribble condition detected
20 packets output, 2526 bytes
Transmitted 0 unicast, 0 broadcast, 0 multicast
0 underruns, 0 output errors, 0 pause output
```

2.9.4 配置静态 AGG

I. 拓扑

在这个例子中，在两个交换机 S1 和 S2 之间的配置三条链路。这三个链路都分配在同一个管理中心（1），使他们能聚合形成一个单一的通道 1。

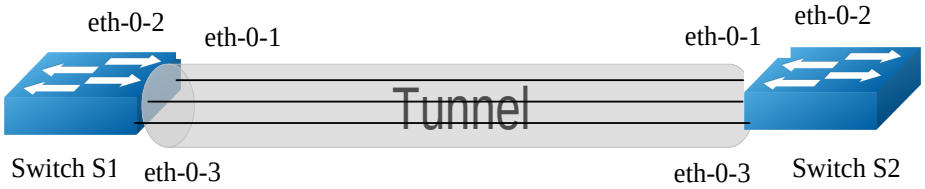


图2-7 LACP 拓扑

II. 配置

Switch 1

Switch1# configure terminal	进入全局配置模式
Switch1(config)# interface eth-0-1	进入接口配置模式
Switch1(config-if)# no shutdown	端口 up
Switch1(config-if)# static-channel-group 1	添加接口到 channel group 1
Switch1(config-if)# exit	退出接口配置模式
Switch1(config)# interface eth-0-2	进入接口配置模式
Switch1(config-if)# static-channel-group 1	添加接口到 channel group 1
Switch1(config-if)# no shutdown	端口 up
Switch1(config-if)# exit	退出接口配置模式
Switch1(config)# interface eth-0-3	进入接口配置模式
Switch1(config-if)# static-channel-group 1	添加接口到 channel group 1

Switch1(config-if)# no shutdown	端口 up
Switch1(config-if)# end	退出接口配置模式

Switch 2

Switch2# configure terminal	进入全局配置模式
Switch2(config)# interface eth-0-1	进入接口配置模式
Switch2(config-if)# no shutdown	端口 up
Switch2(config-if)# static-channel-group 1	添加接口到 channel group 1
Switch2(config-if)# exit	退出接口配置模式
Switch2(config)# interface eth-0-2	进入接口配置模式
Switch2(config-if)# static-channel-group 1	添加接口到 channel group 1
Switch2(config-if)# no shutdown	端口 up
Switch2(config-if)# exit	退出接口配置模式
Switch2(config)# interface eth-0-3	进入接口配置模式
Switch2(config-if)# static-channel-group 1	添加接口到 channel group 1
Switch2(config-if)# no shutdown	端口 up
Switch2(config-if)# end	退出接口配置模式

I. 命令验证

Switch1# show channel-group summary

```
port-channel load-balance hash-arithmetic: xor
port-channel load-balance hash-field-select:
     macsa
Flags:  s - suspend           T - standby
        D - down/admin down   B - in Bundle
        R - Layer3            S - Layer2
        w - wait              U - in use
Mode:   SLB  - static load balance
```

```
        DLB  - dynamic load balance
        SHLB - self-healing load balance
        RR   - round robin load balance
Aggregator Name  Mode      Protocol     Ports
---------------+---------+------------+-----------------------------------------
----
agg1(SU)         SLB       Static         eth-0-1(B) eth-0-2(B) eth-0-3(B)
Switch1# show interface agg 1
Interface agg1
  Interface current state: UP
  Hardware is AGGREGATE, address is cce3.33fc.330b (bia a876.6b2c.9c01)
  Bandwidth 3000000 kbits
  Index 1025 , Metric 1 , Encapsulation ARPA
  Speed - 1000Mb/s , Duplex - Full  , Media type is Aggregation
  Link speed type is autonegotiation, Link duplex type is autonegotiation
  Input flow-control is off, output flow-control is off
  The Maximum Frame Size is 1534 bytes
  VRF binding: not bound
  Label switching is disabled
  No virtual circuit configured
  ARP timeout 01:00:00,  ARP retry interval 1s
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 140 bits/sec, 0 packets/sec
    0 packets input, 0 bytes
    Received 0 unicast, 0 broadcast, 0 multicast
    0 runts, 0 giants, 0 input errors, 0 CRC
    0 frame, 0 overrun, 0 pause input
    0 input packets with dribble condition detected
    1080 packets output, 60614 bytes
    Transmitted 0 unicast, 0 broadcast, 0 multicast
0 underruns, 0 output errors, 0 pause output
```

2.10 流量控制配置

2.10.1 简介

流量控制在直连的以太端口上启用，在拥塞期间允许另一端拥塞的节点暂停链路运作来控制 流量速率。当本地设备在本地检测到了任何拥塞，他能够发送一个暂停帧通知链路伙伴或者远程设备已发生拥塞。紧随收到暂停帧之后，远程设备停止发送任何数据包，这样防止在拥塞期间丢弃任何一个数据包。在自协商链路上，本地的流控制能力能通过链路断开/连接来通知对方。

2.10.2 拓扑

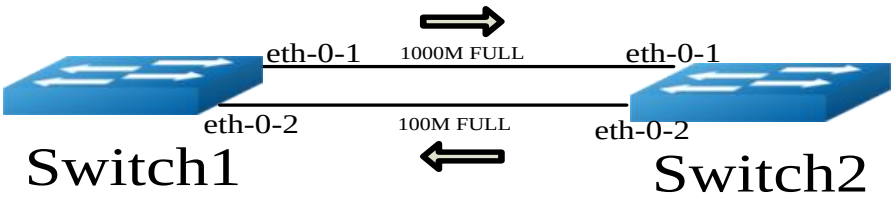


图2-8 流量控制

2.10.3 配置发送流量控制报文

Switch2# configure terminal	进入配置模式
Switch2(config)# interface eth-0-1	进入接口模式
Switch2(config-if)# flowcontrol send on	在端口使能发送流量控制报文
Switch2 (config-if)# exit	返回配置模式

NOTE

流控制只在全双工链路上有效

2.10.4 配置接收流量控制报文

Switch1 # configure terminal	进入配置模式
Switch1(config)# interface eth-0-1	进入接口模式
Switch1(config-if)# flowcontrol receive on	在端口使能接收流量控制报文
Switch1(config-if)# exit	返回配置模式

2.10.5 验证配置

Switch2# show flowcontrol

```
Port       Receive FlowControl  Send FlowControl   RxPause      TxPause
           admin    oper        admin    oper
---------  -------- --------    -------- --------  -----------  -----------
eth-0-1    off      off         on       on        0            0
eth-0-2    off      off         off      off       0            0
eth-0-3    off      off         off      off       0            0
```

Switch2# show flowcontrol eth-0-1

```
Port       Receive FlowControl  Send FlowControl   RxPause      TxPause
```

```
           admin    oper        admin    oper
---------  -------- --------    -------- --------  -----------  -----------
eth-0-1    off      off         on       on        0            0
```

Switch1# show flowcontrol

```
Port       Receive FlowControl  Send FlowControl   RxPause      TxPause
           admin    oper        admin    oper
---------  -------- --------    -------- --------  -----------  -----------
eth-0-1    on       on          off      off       0            0
eth-0-2    off      off         off      off       0            0
eth-0-3    off      off         off      off       0            0
```

Switch1# show flowcontrol eth-0-1

```
Port       Receive FlowControl  Send FlowControl   RxPause      TxPause
           admin    oper        admin    oper
---------  -------- --------    -------- --------  -----------  -----------
eth-0-1    on       on          off      off       0            0
```

2.11 Loopback Detection 配置

2.11.1 简介

网络中的环路会导致设备对广播、组播以及未知单播等报文进行重复发送，造成网络资源的浪费甚至导致网络瘫痪。为了能够及时发现二层网络中的环路，以避免对整个网络造成严重影响，需要提供一种检测功能，使网络中出现环路时能及时通知用户检查网络连接和配置情况，并能够将出问题的接口受控。

Loopback Detection（环回检测）功能可以检测设备的接口是否发生环回，它通过从接口定时发送检测报文，并检测该报文是否会从发出去的接口收到，如果收到从该接口发出的检测报文，则认为当设备的此接口存在环路，可以及时通过发送告警信息到网管系统，使管理人员及时发现并解决网络环路问题，避免长时间的网络异常。此外，设备还可以进行接口受控，根据需求选择配置Trap、关闭接口等处理方式，能迅速将环回对网络的影响降低至最小。

2.11.2 配置使能 Loopback Detect

默认情况下，loopback detection 功能未使能，使能接口 Loopback Detection 功能后，接口才会发送环回检测报文来进行接口环回检测。默认检测报文发送间隔为 5 秒。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# loopback-detect enable	使能 Loopback Detect.
Switch(config)#end	退出配置模式

Switch#show loopback-detect	显示 Loopback Detect 状态

II. 命令验证

```
Switch# show loopback-detect
Loopback detection packet interval(second): 5
Loopback detection recovery time(second):   15
Interface         Action             Status
eth-0-2           shutdown           NORMAL
```

2.11.3 配置 Loopback Detect 报文发送周期

由于网络时刻处于变化中，因此环回检测是一个持续的过程，接口以一定的时间间隔发送环回检测报文，这个时间间隔即 Loopback Detection 报文发送周期。

I. 配置

系统支持配置 Loopback Detection 报文的发送间隔（1-300 秒）。Loopback 状态恢复的时间为发送间隔的 3 倍，且最小为 10 秒。Loop 消失后，接口状态会自动恢复到之前的状态。

Switch#configure terminal	进入配置模式
Switch(config)# loopback-detect packet-interval 10	设置 Loopback Detect 报文发送间隔为 10 秒
Switch(config-if)# end	退出配置模式
Switch# show loopback-detect packet-interval	验证 Loopback Detect 报文发送间隔命令

II. 命令验证

```
Switch# show loopback-detect packet-interval
Loopback detection packet interval(second): 10
Loopback detection recovery time(second):   30
```

2.11.4 配置 Loopback Detect 处理动作

如果发现接口有环回，设备会将该接口设置为处于环回检测工作状态，可配置发送告警、关闭接口等处理动作。

I. 配置

开启 Loopback Detection 功能后，接口会周期性地检测是否收到环回报文。用户可以配置检测到有环回报文时的处理方式，尽量减轻环路对本设备和整个网络的影响。

Switch#configure terminal	进入配置模式

Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# loopback-detect action shutdown	配置 loopback detect 处理动作为 shutdown
Switch(config-if)# end	退出配置模式
Switch# show loopback-detect interface eth-0-1	验证命令

II. 命令验证

```
Switch# show loopback-detect interface eth-0-1
Interface         Action              Status
eth-0-1           shutdown            NORMAL
```

2.11.5 配置对指定 VLAN 的 Loopback Detection 功能

接口开启 Loopback Detection 功能后，系统默认发送的为 Untag 检测报文，即不对任何指定 VLAN 进行环回检测。当接口是以 Tagged 方式加入 VLAN，接口发出去 Untag 检测报文在链路上会被丢弃，接口将收不到环回回来的报文，因此需要配置对指定的 VLAN 进行环回检测。配置对指定 VLAN 的 Loopback Detection 功能，接口会定时发送 1 份 Untagged 检测报文和多份带指定 VLAN Tag 的检测报文，一个接口最多可发送 8 份带指定 VLAN Tag 的检测报文。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# loopback-detect packet vlan 20	配置该接口下 VLAN 20 的 loopback detect 功能
Switch(config-if)# end	退出配置模式
Switch# show running-config interface eth-0-1	验证命令.

II. 命令验证

```
Switch# show running-config interface eth-0-1
Building configuration...
!
interface eth-0-1
 loopback-detect enable
loopback-detect packet vlan 2
```

2.12 基于优先级的流量控制配置

2.12.1 简介

基于优先级的流量控制（PFC）是对 flow control 机制的一种增强（如下图 2-1）。当前以太网暂停选择（IEEE 802.3 Annex 31B）也能达到无丢包的要求，但它会阻止一条链路上的所有流量，本质上来讲，它会暂停整条链路。PFC 允许在一条以太网链路上创建 8 个虚拟通道，并为每条虚拟通道指定一个 IEEE 802.1P 优先等级，允许单独暂停和重启其中任意一条虚拟通道，同时允许其它虚拟通道的流量无中断通过。这一方法使网络能够为单个虚拟链路创建无丢包类别的服务，使其能够与同一接口上的其它流量类型共存。

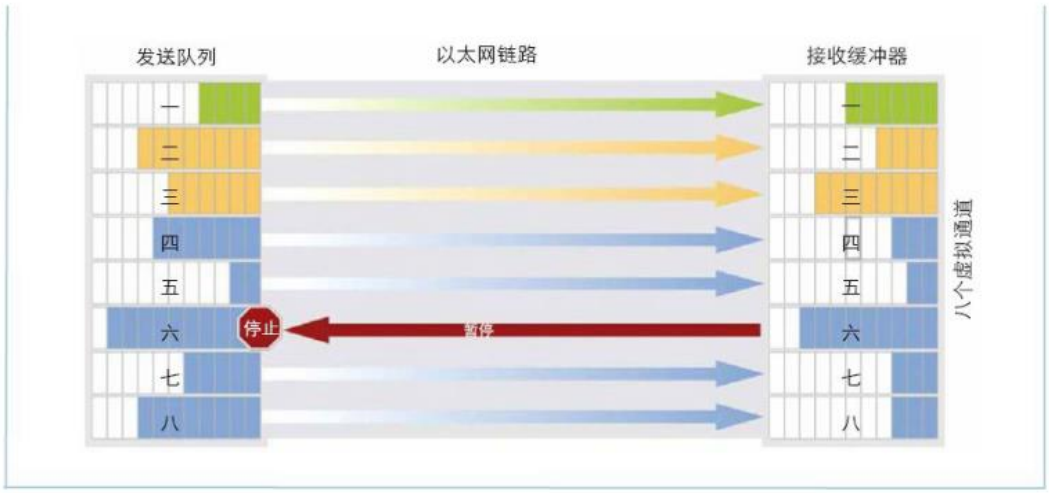


图2-9 基于优先级的流量控制

2.12.2 拓扑

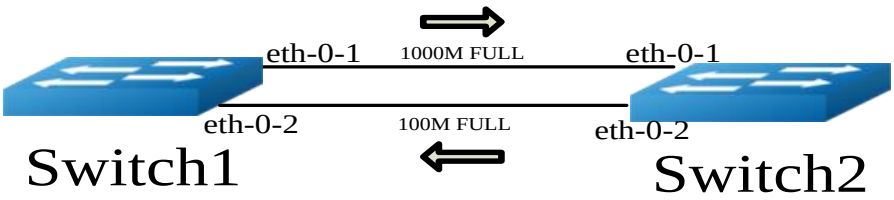


图2-10 基于优先级的流量控制

2.12.3 配置使能 PFC 功能

Switch1# configure terminal	进入配置模式
Switch1(config)# lldp enable	全局使能 lldp

Switch1(config)# interface eth-0-1	进入接口模式
Switch1(config-if)#lldp enable	在接口下使能 lldp
Switch1(config-if)# lldp tlv 8021-org-specific dcbx	在接口下使能 dcbx tlv
Switch1(config-if)# priority-flow-control mode on	在端口 1 上使能 PFC，不与对端协商
Switch1(config-if)# priority-flow-control enable priority 2 3 4	配置在 priority 2 3 4 上使能 PFC
Switch1(config)# interface eth-0-2	进入接口模式
Switch1(config-if)#lldp enable	在接口下使能 lldp
Switch1(config-if)# lldp tlv 8021-org-specific dcbx	在接口下使能 dcbx tlv
Switch1(config-if)# priority-flow-control mode auto	在端口 2 上使能 PFC，需要与对端协商成功才能启用
Switch1(config-if)# priority-flow-control enable priority 2 3 4	配置在 priority 2 3 4 上使能 PFC
Switch1 (config-if)# exit	返回配置模式

Switch2# configure terminal	进入配置模式
Switch2(config)# lldp enable	全局使能 lldp
Switch2(config)# interface eth-0-1	进入接口模式
Switch2(config-if)#lldp enable	在接口下使能 lldp
Switch2(config-if)# lldp tlv 8021-org-specific dcbx	在接口下使能 dcbx tlv
Switch2(config-if)# priority-flow-control mode on	在端口 1 上使能 PFC，不与对端协商
Switch2(config-if)# priority-flow-control enable priority 2 3 4	配置在 priority 2 3 4 上使能 PFC
Switch2(config)# interface eth-0-2	进入接口模式
Switch2(config-if)#lldp enable	在接口下使能 lldp
Switch2(config-if)# lldp tlv 8021-org-	在接口下使能 dcbx tlv

specific dcbx	
Switch2(config-if)# priority-flow-control mode auto	在端口 2 上使能 PFC，需要与对端协商成功才能启用
Switch2(config-if)# priority-flow-control enable priority 2 3 4	配置在 priority 2 3 4 上使能 PFC
Switch2 (config-if)# exit	返回配置模式

NOTE

流控制只在全双工链路上有效

2.12.4 验证配置

Switch2# show priority-flow-control

Port	PFC-enable		PFC-enable on priority		RxPause	TxPause
	admin	oper	admin	oper		
eth-0-1	on	on	234	234	0	0
eth-0-2	auto	off	234	off	0	0
eth-0-3	off	off	off	off	0	0
eth-0-4	off	off	off	off	0	0
eth-0-5	off	off	off	off	0	0
eth-0-6	off	off	off	off	0	0
eth-0-7	off	off	off	off	0	0
eth-0-10	off	off	off	off	0	0
eth-0-11	off	off	off	off	0	0
eth-0-12	off	off	off	off	0	0
eth-0-13	off	off	off	off	0	0
eth-0-14	off	off	off	off	0	0
eth-0-15	off	off	off	off	0	0
eth-0-16	off	off	off	off	0	0
eth-0-17	off	off	off	off	0	0
eth-0-18	off	off	off	off	0	0
eth-0-19	off	off	off	off	0	0
eth-0-20	off	off	off	off	0	0

2.13 风暴控制配置

2.13.1 概述

风暴控制是指在指定接口上，通过对接收的最大广播、最大未知组播以及最大未知单播流量进行限制，防止泛洪消耗过多的交换机资源，确保业务正常运行。

可以使用以下两种方式之一进行风暴控制：

- 百分比模式 (Level)
- 包速率模式 (PPS)

2.13.2 术语

PPS：Packets per second 的首字母缩写，每秒传输的报文数，即包速率。

2.13.3 使用百分比(LEVEL)模式配置风暴控制

I. 基于二层端口的配置

Switch 1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# storm-control unicast level 0.1	设置限制未知单播报文的百分比
Switch(config-if)# storm-control multicast level 1	设置限制组播报文的百分比
Switch(config-if)# storm-control broadcast level 10	设置限制广播报文的百分比
Switch(config-if)# end	退出到 EXEC 模式
Switch# show storm-control interface eth-0-1	显示风暴控制在接口上的配置信息

I. 命令验证

Switch# show storm-control interface eth-0-1

```
Port     ucastMode ucastLevel bcastMode bcastLevel mcastMode mcastLevel
eth-0-1 Level             0.10 Level            10.00 Level             1.00
```

2.13.4 使用包速率(PPS)模式配置风暴控制

I. 基于二层端口的配置

Switch 1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# storm-control unicast pps 1000	设置未知单播报文每秒通过 1000 个
Switch(config-if)# storm-control multicast pps 10000	设置组播报文每秒通过 10000 个
Switch(config-if)# storm-control broadcast pps 100000	设置广播报文每秒通过 100000 个

Switch(config-if)# end	退出到 EXEC 模式
Switch# show storm-control interface eth-0-1	显示风暴控制在接口上的配置信息

I. 命令验证

Switch# show storm-control interface eth-0-1

```
Port     ucastMode ucastLevel bcastMode bcastLevel mcastMode mcastLevel
eth-0-1 PPS        1000          PPS        100000      PPS         10000
```

2.14 L2 Protocol Tunnel 配置

2.14.1 简介

在不同站点上通过运营商网络连接的客户需要能正常运行二层协议。这一需求希望运营商网络能透明传输 STP/RSTP/MSTP 报文，因此客户可以跨越运营商网络构建自己的 STP 树，切断冗余链路。

当二层协议报文透传功能被启用后，在运营商网络边缘的交换机会使用一个新的二层头封装二层协议报文，然后向运营商网络传输。在运营商的网络里，该封装后的报文作为普通报文传输。当报文到达运营商网络边缘时，该报文新加的二层头被剥去，然后二层协议报文被转发给用户交换机处理。

二层协议报文透传功能可以独立使用也可以和 QinQ 功能一起使用。

2.14.2 配置透传指定的二层协议报文

I. 简介

指定的二层协议报文包括 STP BPDU 报文，Slow proto 报文，802.1X EAPOL 报文，CFM 报文。

在下面的例子中，Switch1 eth-0-1 和 Switch2 eth-0-1 配置成 tunnel 端口。Switch1 eth-0-2 和 Switch2 的 eth-0-2 配置成上联口。如果在 Switch1 的 eth-0-1 口上收到上面三种协议报文，协议报文会加上新的二层头然后从上联口发出。在新的二层头中：目的 MAC 是 tunnel dmac；源 MAC 是交换机的 route-mac；VLAN id 是 tunnel evc 所对应的 VLAN id；VLAN priority 是配置的 Layer 2 protocol cos；Ethertype 是 0xFFEE。如果在 Switch2 的 eth-0-2 上收到带上新二层头的协议报文，协议报文上所带的新二层头会被剥去，然后从 Switch2 的 eth-0-1 发出。

II. 拓扑

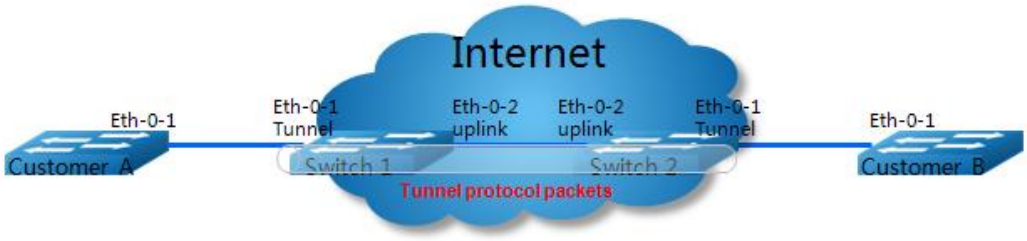


图2-11 L2 Protocol Tunnel 拓扑图

III. 配置

使用下表所示的命令配置 Switch 1 和 Switch 2。

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	进入 VLAN 配置模式
Switch(config-vlan)# vlan 2-5	创建 vlan 2-5
Switch(config)# ethernet evc evc_c1	创建 EVC evc_c1
Switch(config-evc)# dot1q mapped-vlan 2	配置 evc_c1 对应的 vlan id 为 2
Switch(config)# ethernet evc evc_c2	创建 EVC evc_c2
Switch(config-evc)# dot1q mapped-vlan 3	配置 evc_c2 对应的 vlan id 为 3
Switch(config)# ethernet evc evc_c3	创建 EVC evc_c3
Switch(config-evc)# dot1q mapped-vlan 4	配置 evc_c3 对应的 vlan id 为 4
Switch(config)# ethernet evc evc_c4	创建 EVC evc_c4
Switch(config-evc)# dot1q mapped-vlan 5	配置 evc_c3 对应的 vlan id 为 5
Switch(config)# l2protocol enable	全局使能二层协议报文透传
Switch(config)# l2protocol tunnel-dmac 0100.0CCD.CDD2	全局配置 tunnel dmac
Switch(config)# interface eth-0-1	进入端口模式
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# switchport mode trunk	配置端口为 trunk 口

Switch(config-if)# switchport trunk allowed vlan add 2-5	配置端口允许 vlan 2-5 通过
Switch(config-if)# spanning-tree port disable	在端口上关闭 STP 协议
Switch(config-if)# l2protocol stp tunnel evc evc_c1	配置 stp bpdu 报文 tunnel 到 evc_c1
Switch(config-if)# l2protocol slow-proto tunnel evc evc_c2	配置 slow protocol 报文 tunnel 到 evc_c2
Switch(config-if)# l2protocol dot1x tunnel evc evc_c3	配置 dot1x eapol 报文 tunnel 到 evc_c3
Switch(config-if)# l2protocol cfm tunnel evc evc_c4	配置 cfm 报文 tunnel 到 evc_c4
Switch(config)# interface eth-0-2	进入端口模式
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch(config-if)# switchport trunk allowed vlan add 2-5	配置端口允许 vlan 2-5 通过
Switch(config-if)# l2protocol uplink enable	配置端口为二层协议报文透传时的上联口

IV. 验证配置

Switch1# show l2protocol interface eth-0-1

```
Interface  PDU Address     MASK            Status     EVC
=========  ==============  ==============  ========  ================
eth-0-1    stp             FFFF.FFFF.FFFF  Tunnel     evc_c1
eth-0-1    slow-proto      FFFF.FFFF.FFFF  Tunnel     evc_c2
eth-0-1    dot1x           FFFF.FFFF.FFFF  Tunnel     evc_c3
eth-0-1    cfm             FFFF.FFFF.FFFF  Tunnel     evc_c4
```

Switch1# show l2protocol interface eth-0-2

```
Interface  PDU Address     MASK            Status     EVC
=========  ==============  ==============  ========   ================
eth-0-2    stp             FFFF.FFFF.FFFF  Peer       N/A
eth-0-2    slow-proto      FFFF.FFFF.FFFF  Peer       N/A
eth-0-2    dot1x           FFFF.FFFF.FFFF  Peer       N/A
eth-0-2    cfm             FFFF.FFFF.FFFF  Peer       N/A
eth-0-2    N/A             N/A             Uplink     N/A
```

Switch1# show l2protocol tunnel-dmac

```
Layer2 protocols tunnel destination MAC address is 0100.0ccd.cdd2
```

2.14.3 配置透传可配的二层协议报文

I. 简介

可配的二层协议报文是指地址是 0180.c200.0000 – 0x0180.c2ff.ffff 间的报文，

全 mac 地址协议报文是指地址是 0000.0000.0000 – ffff.ffff.fff 间的报文。

在下面的例子中，Switch1 eth-0-1 和 Switch2 eth-0-1 配置成 tunnel 端口。Switch1 eth-0-2 和 Switch2 eth-0-2 配置成上联口。如果在 Switch 1 的 eth-0-1 口上收到协议报文符合配置的 mac 地址，协议报文会加上新的二层头然后从上联口发出。在新的二层头中：目的 MAC 是 tunnel dmac；源 MAC 是交换机的 route-mac；VLAN id 是 tunnel evc 所对应的 VLAN id；VLAN priority 是配置的 Layer 2 protocol cos；Ethertype 是 0xFFEE。如果在 Switch2 的 eth-0-2 上收到带上新二层头的协议报文，协议报文上所带的新二层头会被剥去，然后从 Switch2 的 eth-0-1 发出。

II. 拓扑

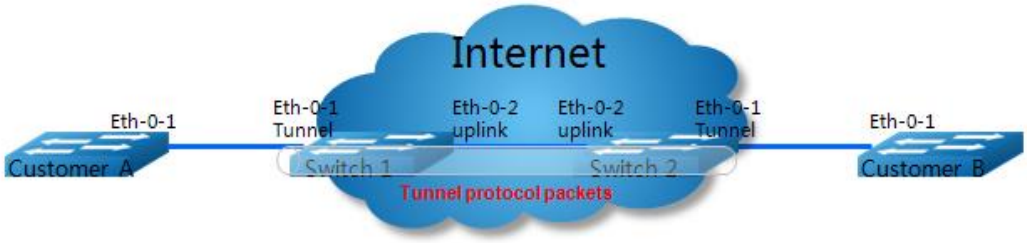


图2-12 L2 Protocol Tunnel 拓扑图

III. 配置

使用下表所示的命令，配置 Switch 1 和 Switch 2。

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	进入 VLAN 配置模式
Switch(config-vlan)# vlan 2-4	创建 vlan 2-4
Switch(config)# ethernet evc evc_c1	创建 EVC evc_c1
Switch(config-evc)# dot1q mapped-vlan 2	配置 evc_c1 对应的 vlan id 为 2
Switch(config)# ethernet evc evc_c2	创建 EVC evc_c2
Switch(config-evc)# dot1q mapped-vlan 3	配置 evc_c2 对应的 vlan id 为 3
Switch(config)# ethernet evc evc_c3	创建 EVC evc_c3

Switch(config-evc)# dot1q mapped-vlan 4	配置 evc_c2 对应的 vlan id 为 4
Switch(config)# l2protocol enable	全局使能二层协议报文透传
Switch(config)# l2protocol tunnel-dmac 0100.0CCD.CDD2	全局配置 tunnel dmac
Switch1(config)# l2protocol mac 3 0180.C200.0008	配置可透传的二层协议报文 3 的 mac 地址为 0180.C200.0008
Switch1(config)# l2protocol mac 4 0180.C200.0009	配置可透传的二层协议报文 4 的 mac 地址为 0180.C200.0009
Switch1(config)# l2protocol full-mac 0100.0CCC.CCCC	配置可透传全 mac 地址为 0100.0CCC.CCCC
Switch(config)# interface eth-0-1	进入端口模式
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch(config-if)# switchport trunk allowed vlan add 2-4	配置端口允许 vlan 2-4 通过
Switch(config-if)# spanning-tree port disable	在端口上关闭 STP 协议
Switch(config-if)# l2protocol mac 3 tunnel evc evc_c1	配置将二层协议报文 3 透传到 evc_c1
Switch(config-if)# l2protocol mac 4 tunnel evc evc_c2	配置将二层协议报文 4 透传到 evc_c2
Switch(config-if)# l2protocol full-mac tunnel evc evc_c3	配置将全 mac 地址透传到 evc_c3
Switch(config)# interface eth-0-2	进入端口模式
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch(config-if)# switchport trunk allowed vlan add 2-4	配置端口允许 vlan 2-4 通过
Switch(config-if)# l2protocol uplink enable	配置端口为二层协议报文透传时的上联口

IV. 验证配置

Switch1# show l2protocol interface eth-0-1

```
Interface   PDU Address      MASK              Status     EVC
```

```
=========  ==============  ==============  ========  ================
eth-0-1    0180.c200.0008  FFFF.FFFF.FFFF  Tunnel    evc_c1
eth-0-1    0180.c200.0009  FFFF.FFFF.FFFF  Tunnel    evc_c2
eth-0-1    0100.0ccc.cccc  FFFF.FFFF.FFFF  Tunnel    evc_c3
eth-0-1    stp             FFFF.FFFF.FFFF  Peer      N/A
eth-0-1    slow-proto      FFFF.FFFF.FFFF  Peer      N/A
eth-0-1    dot1x           FFFF.FFFF.FFFF  Peer      N/A
eth-0-1    cfm             FFFF.FFFF.FFFF  Peer      N/A
```

Switch1# show l2protocol interface eth-0-2

```
Interface  PDU Address     MASK            Status    EVC
=========  ==============  ==============  ========  ================
eth-0-2    0180.c200.0008  FFFF.FFFF.FFFF  Peer      N/A
eth-0-2    0180.c200.0009  FFFF.FFFF.FFFF  Peer      N/A
eth-0-2    0100.0ccc.cccc  FFFF.FFFF.FFFF  Peer      N/A
eth-0-2    stp             FFFF.FFFF.FFFF  Peer      N/A
eth-0-2    slow-proto      FFFF.FFFF.FFFF  Peer      N/A
eth-0-2    dot1x           FFFF.FFFF.FFFF  Peer      N/A
eth-0-2    cfm             FFFF.FFFF.FFFF  Peer      N/A
eth-0-2    N/A             N/A             Uplink    N/A
```

Switch1# show l2protocol tunnel-dmac

```
Layer2 protocols tunnel destination MAC address is 0100.0ccd.cdd2
```

2.15 MSTP 配置

2.15.1 简介

MST（Multiple Spanning Tree）多生成树（MST）是把 IEEE802.1w 的快速生成树（RST）算法扩展而得到的。MST 能够通过 trunk 链路建立多个生成树，关联 VLANs 到相关的生成树实例，并且每个生成树实例可以具备区别于其他实例的拓扑结构。MST 提供了多个数据转发路径和负载均衡，提高了网络容错能力。因为一个实例（转发路径）的故障不会影响其他实例（转发路径）。一个生成树实例只能存在于一致的 VLAN 实例分配的桥中，必须用同样的 MST 配置信息来配置一组桥，这使得这些桥能属于同一组生成树实例，具备同样的 MST 配置信息的互连的桥构成多生成树区（MST Region）。

MSTP 将环路网络修剪成为一个无环的树型网络，避免报文在环路网络中的增生和无限循环，同时还提供了数据转发的多个冗余路径，在数据转发过程中实现 VLAN 数据的负载均衡。MSTP 兼容 STP 和 RSTP，并且可以弥补 STP 和 RSTP 的缺陷。它既可以快速收敛，也能使不同 VLAN 的流量沿各自的路径分发，从而为冗余链路提供了更好的负载分担机制。

2.15.2 拓扑

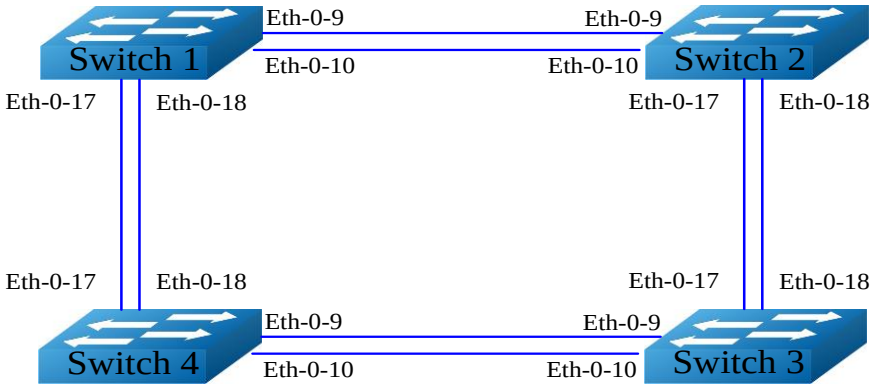


图2-13 MSTP 拓扑示例

2.15.3 配置

此配置示例假定您正在运行的二层协议。如果您使用的是非二层协议，必须在每个端口上运行的交换机端口命令来设置二层协议。

Switch 1 – Switch 4

Switch# configure terminal	进入配置模式
Switch(config)# spanning-tree mode mstp	配置 STP 的模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config-vlan)# vlan 10	创建 VLAN10
Switch(config-vlan)# vlan 20	创建 VLAN20
Switch(config-vlan)# exit	退出 VLAN 模式
Switch(config)# spanning-tree mst configuration	进入 MSTP 配置模式
Switch(config-mst)# region RegionName	配置 MSTP 的区域名字
Switch(config-mst)# instance 1 vlan 10	配置 MSTP 的实例 1 关联 VLAN10
Switch(config-mst)# instance 2 vlan 20	配置 MSTP 的实例 2 关联 VLAN20
Switch(config-mst)# exit	退出 MSTP 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# switchport mode trunk	设置接口为 Trunk

Switch(config-if)# switchport trunk allowed vlan all	配置 Trunk 允许所有的 VLAN 通过
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-10	进入接口模式
Switch(config-if)# switchport mode trunk	设置接口为 Trunk
Switch(config-if)# switchport trunk allowed vlan all	配置 Trunk 允许所有的 VLAN 通过
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-17	进入接口模式
Switch(config-if)# switchport mode trunk	设置端口为 Trunk 模式
Switch(config-if)# switchport trunk allowed vlan all	配置 Trunk 允许所有的 VLAN 通过
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-18	进入接口模式
Switch(config-if)# switchport mode trunk	设置端口为 Trunk 模式
Switch(config-if)# switchport trunk allowed vlan all	配置 Trunk 允许所有的 VLAN 通过
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# exit	退出接口模式
Switch(config)# exit	退出配置模式

Switch 1

Switch# configure terminal	进入配置模式
Switch(config)# spanning-tree priority 0	设置 STP 的优先级为 0
Switch(config)# spanning-tree enable	启用 STP，系统默认未启用

	STP

Switch 2

Switch# configure terminal	进入配置模式
Switch(config)# spanning-tree instance 1 priority 0	配置 STP 实例 1 的优先级为 0
Switch(config)# spanning-tree enable	启用 STP，系统默认未启用 STP

Switch 3

Switch# configure terminal	进入配置模式
Switch(config)# spanning-tree instance 2 priority 0	设置 STP 实例 2 的优先级为 0
Switch(config)# spanning-tree enable	启用 STP，系统默认未启用 STP

Switch 4

Switch# configure terminal	进入配置模式
Switch(config)# spanning-tree enable	启用 STP，系统默认未启用 STP

2.15.4 命令验证

步骤 1 验证 Switch 1 的 MSTP 的端口的状态。

Switch# show spanning-tree mst brief

```
##### MST0:  Vlans: 1
Multiple spanning tree protocol Enabled
Root ID     Priority       0 (0x0000)
            Address     2225.fa28.c900
            Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
Bridge ID   Priority       0 (0x0000)
            Address     2225.fa28.c900
            Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
            Aging Time  300 sec
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Designated     Forwarding      20000           128.9        P2p
```

```
eth-0-10    Designated     Forwarding      20000           128.10       P2p
eth-0-17    Designated     Forwarding      20000           128.17       P2p
eth-0-18    Designated     Forwarding      20000           128.18       P2p
##### MST1:  Vlans: 10
Root ID     Priority       1 (0x0001)
            Address     9c9a.7d91.9f00
Bridge ID   Priority    32769 (0x8001)
            Address     2225.fa28.c900
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Rootport       Forwarding      20000           128.9        P2p
eth-0-10    Alternate      Discarding      20000           128.10       P2p
eth-0-17    Designated     Forwarding      20000           128.17       P2p
eth-0-18    Designated     Forwarding      20000           128.18       P2p
##### MST2:  Vlans: 20
Root ID     Priority       2 (0x0002)
            Address     304c.275b.b200
Bridge ID   Priority    32770 (0x8002)
            Address     2225.fa28.c900
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Alternate      Discarding      20000           128.9        P2p
eth-0-10    Alternate      Discarding      20000           128.10       P2p
eth-0-17    Rootport       Forwarding      20000           128.17       P2p
eth-0-18    Alternate      Discarding      20000           128.18       P2p
```

步骤 2 验证 Switch 2 的 MSTP 的端口的状态。

Switch# show spanning-tree mst brief

```
##### MST0:  Vlans: 1
Multiple spanning tree protocol Enabled
Root ID     Priority       0 (0x0000)
            Address     2225.fa28.c900
            Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
Bridge ID   Priority    32768 (0x8000)
            Address     9c9a.7d91.9f00
            Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
            Aging Time  300 sec
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Rootport       Forwarding      20000           128.9        P2p
eth-0-10    Alternate      Discarding      20000           128.10       P2p
eth-0-17    Designated     Forwarding      20000           128.17       P2p
eth-0-18    Designated     Forwarding      20000           128.18       P2p
##### MST1:  Vlans: 10
Root ID     Priority       1 (0x0001)
            Address     9c9a.7d91.9f00
Bridge ID   Priority       1 (0x0001)
            Address     9c9a.7d91.9f00
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Designated     Forwarding      20000           128.9        P2p
eth-0-10    Designated     Forwarding      20000           128.10       P2p
```

```
eth-0-17    Designated     Forwarding      20000           128.17        P2p
eth-0-18    Designated     Forwarding      20000           128.18        P2p
##### MST2:  Vlans: 20
Root ID     Priority       2 (0x0002)
            Address     304c.275b.b200
Bridge ID   Priority    32770 (0x8002)
            Address     9c9a.7d91.9f00
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Designated     Forwarding      20000           128.9         P2p
eth-0-10    Designated     Forwarding      20000           128.10        P2p
eth-0-17    Rootport       Forwarding      20000           128.17        P2p
eth-0-18    Alternate      Discarding      20000           128.18        P2p
```

步骤 3 验证 Switch 3 的 MSTP 的端口的状态。

Switch# show spanning-tree mst brief

```
##### MST0:  Vlans: 1
Multiple spanning tree protocol Enabled
Root ID     Priority       0 (0x0000)
            Address     2225.fa28.c900
            Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
Bridge ID   Priority    32768 (0x8000)
            Address     304c.275b.b200
            Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
            Aging Time  300 sec
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Rootport       Forwarding      20000           128.9        P2p
eth-0-10    Alternate      Discarding      20000           128.10       P2p
eth-0-17    Alternate      Discarding      20000           128.17       P2p
eth-0-18    Alternate      Discarding      20000           128.18       P2p
##### MST1:  Vlans: 10
Root ID     Priority       1 (0x0001)
            Address     9c9a.7d91.9f00
Bridge ID   Priority    32769 (0x8001)
            Address     304c.275b.b200
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Designated     Forwarding      20000           128.9         P2p
eth-0-10    Designated     Forwarding      20000           128.10        P2p
eth-0-17    Rootport       Forwarding      20000           128.17        P2p
eth-0-18    Alternate      Discarding      20000           128.18        P2p
##### MST2:  Vlans: 20
Root ID     Priority       2 (0x0002)
            Address     304c.275b.b200
Bridge ID   Priority       2 (0x0002)
            Address     304c.275b.b200
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Designated     Forwarding      20000           128.9         P2p
eth-0-10    Designated     Forwarding      20000           128.10        P2p
eth-0-17    Designated     Forwarding      20000           128.17        P2p
```

```
eth-0-18    Designated     Forwarding      20000           128.18        P2p
```

步骤 4 验证 Switch 4 的 MSTP 的端口的状态。

Switch# show spanning-tree mst brief

```
s##### MST0:  Vlans: 1
Multiple spanning tree protocol Enabled
Root ID     Priority       0 (0x0000)
            Address     2225.fa28.c900
            Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
Bridge ID   Priority    32768 (0x8000)
            Address     80a4.be55.6400
            Hello Time  2 sec  Max Age  20 sec Forward Delay 15 sec
            Aging Time  300 sec
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Designated     Forwarding      20000           128.9         P2p
eth-0-10    Designated     Forwarding      20000           128.10        P2p
eth-0-17    Rootport       Forwarding      20000           128.17        P2p
eth-0-18    Alternate      Discarding      20000           128.18        P2p
##### MST1:  Vlans: 10
Root ID     Priority       1 (0x0001)
            Address     9c9a.7d91.9f00
Bridge ID   Priority    32769 (0x8001)
            Address     80a4.be55.6400
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Alternate      Discarding      20000           128.9         P2p
eth-0-10    Alternate      Discarding      20000           128.10        P2p
eth-0-17    Rootport       Forwarding      20000           128.17        P2p
eth-0-18    Alternate      Discarding      20000           128.18        P2p
##### MST2:  Vlans: 20
Root ID     Priority       2 (0x0002)
            Address     304c.275b.b200
Bridge ID   Priority    32770 (0x8002)
            Address     80a4.be55.6400
Interface    Role          State            Cost      Priority.Number     Type
--------------------------------------------------------------------------------
eth-0-9     Rootport       Forwarding      20000           128.9         P2p
eth-0-10    Alternate      Discarding      20000           128.10        P2p
eth-0-17    Designated     Forwarding      20000           128.17        P2p
eth-0-18    Designated     Forwarding      20000           128.18        P2p
```

2.16 MLAG 配置

2.16.1 简介

在高可靠性的数据中心拓扑中，典型的会通过两台聚合交换机来连接 TOR 交换机和服务器以提供冗余保护。在这样的拓扑结构中，生成树协议通过 block 聚合交换机的一半的端口来防止网络环路，但这样做会降低 50%的带宽。

通过部署 MLAG 可以解决这个问题。在两台聚合交换机的中间通过一条 MLAG 链路进行连接，使其在逻辑上如同一台设备。两台设备上的端口共同形成聚合口，使得所有端口可以共同参与数据流量的转发。

MLAG 提供了如下好处：

- 在网络流量增加的时候提供了更高的带宽；
- 通过减少被 STP block 的端口的方式更加高效的利用了网络带宽；
- 使用静态 LAG 或者 LACP 来连接其他交换机或者服务器，而不需要借助其他协议；
- 支持通过生成树协议来防止环路

2.16.2 拓扑

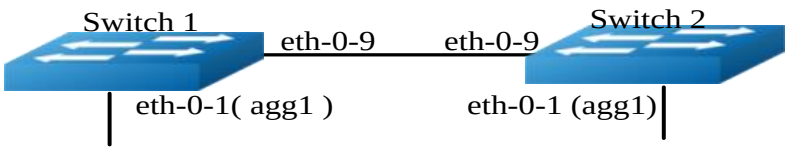


图2-14 MLAG 配置拓扑

2.16.3 配置

I. 配置 switch1

Switch1 (config)# vlan database	进入 vlan 模式
Switch1 (config-vlan)# vlan 10,4094	创建 vlan10 和 vlan4094
Switch1(config-vlan)# exit	退出 vlan 模式并返回全局配置模式
Switch1 (config)# interface eth-0-1	进入接口配置模式
Switch1(config-if)# static-channel-group 1	将此接口加入静态 agg1 组
Switch1(config-if)# no shutdown	配置端口 up
Switch1(config-if)# exit	退出接口配置模式并返回全局配置模式

Switch1 (config)# interface eth-0-9	进入接口配置模式
Switch1(config-if)# switchport mode trunk	配置接口为 trunk 口
Switch1(config-if)# switchport trunk allowed vlan all	在此 trunk 口允许所有 vlan
Switch1(config-if)# spanning-tree port disable	去使能接口的生成树协议
Switch1(config-if)# no shutdown	配置端口 up
Switch1(config-if)# exit	退出接口配置模式并返回全局配置模式
Switch1 (config)# interface agg1	进入聚合 agg1 接口模式
Switch1(config-if)# switchport mode trunk	配置为 turnk 口
Switch1(config-if)# switchport trunk allowed vlan add 10	在此端口上允许 vlan 10
Switch1(config-if)# mlag 1	绑定此接口到 mlag1 上
Switch1(config-if)# exit	退出接口配置模式并返回全局配置模式
Switch1 (config)# interface vlan4094	创建三层接口 vlan 4094
Switch1(config-if)# ip address 12.1.1.1/24	配置 ip 地址为 12.1.1.1/24
Switch1(config-if)# exit	退出接口配置模式并返回全局配置模式
Switch1 (config)# mlag configuration	进入 mlag 配置模式
Switch1 (config-mlag)# peer-link eth-0-9	配置 peer link 链路
Switch1 (config-mlag)# peer-address 12.1.1.2	配置 peer 邻居地址
Switch1 (config-mlag)# exit	退出 mlag 模式并返回全局配置模式

II. 配置 switch2

Switch2 (config)# vlan database	进入 vlan 模式
Switch2 (config-vlan)# vlan 10,4094	创建 vlan10 和 vlan4094
Switch2(config-vlan)# exit	返回全局配置模式
Switch2 (config)# interface eth-0-1	进入接口模式
Switch2(config-if)# static-channel-group 1	将此接口加入到静态 agg1 中
Switch2(config-if)# no shutdown	使能接口 up

Switch2(config-if)# exit	返回全局配置模式
Switch2 (config)# interface eth-0-9	进入接口模式
Switch2(config-if)# switchport mode trunk	配置接口为 trunk 模式
Switch2(config-if)# switchport trunk allowed vlan all	在此 trunk 口允许所有 vlan
Switch2(config-if)# spanning-tree port disable	去使能接口的生成树协议
Switch2(config-if)# no shutdown	使能接口 up
Switch2(config-if)# exit	返回到全局配置模式
Switch2 (config)# interface agg1	进入 agg 接口模式
Switch2(config-if)# switchport mode trunk	配置接口为 trunk 口
Switch2(config-if)# switchport trunk allowed vlan add 10	在此接口上允许 vlan10
Switch2(config-if)# mlag 1	绑定此接口到 mlag1 上
Switch2(config-if)# exit	返回到全局配置模式
Switch2 (config)# interface vlan4094	创建 interface vlan4094
Switch2(config-if)# ip address 12.1.1.2/24	配置 ip 地址 12.1.1.2/24
Switch2(config-if)# exit	返回到全局配置模式
Switch2 (config)# mlag configuration	进入 mlag 配置模式
Switch2 (config-mlag)# peer-link eth-0-9	配置 peer link 链路
Switch2 (config-mlag)# peer-address 12.1.1.1	配置 peer 邻居地址
Switch2 (config-mlag)# end	返回到特权模式

2.16.4 命令验证

验证 switch1

```
Switch1# show mlag
MLAG configuration:
-----------------
role        : Master
local_sysid : ea90.aecc.cc00
mlag_sysid  : ea90.aecc.cc00
peer-link   : eth-0-9
peer conf   : Yes
```

```
Switch1# show mlag interface
mlagid  local-if  local-state    remote-state
1       agg1      up             up
Switch1# show mlag peer
MLAG neighbor is 12.1.1.2, MLAG version 1
MLAG state = Established, up for 00:13:07
Last read 00:00:48, hold time is 240, keepalive interval is 60 seconds
Received 19 messages,Sent 23 messages
Open     : received 1, sent 2
KAlive   : received 15, sent 16
Fdb sync : received 0, sent 0
Failover : received 0, sent 0
Conf     : received 1, sent 1
STP Total: received 2, sent 4
 Global  : received 2, sent 3
 Packet  : received 0, sent 0
 Instance: received 0, sent 0
 State   : received 0, sent 1
Connections established 1; dropped 0
Local host: 12.1.1.1, Local port: 61000
Foreign host: 12.1.1.2, Foreign port: 46157
remote_sysid: baa7.8606.8b00
Switch1# show mac address-table
          Mac Address Table
------------------------------------------
(*) - Security Entry
Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
验证 switch2
Switch2# show mlag
MLAG configuration:
-----------------
role        : Slave
local_sysid : baa7.8606.8b00
mlag_sysid  : ea90.aecc.cc00
peer-link   : eth-0-9
peer conf   : Yes
Switch2# show mlag interface
mlagid  local-if  local-state    remote-state
1       agg1      up             up
Switch2# show mlag peer
MLAG neighbor is 12.1.1.1, MLAG version 1
MLAG state = Established, up for 00:14:29
Last read 00:00:48, hold time is 240, keepalive interval is 60 seconds
Received 23 messages,Sent 21 messages
Open     : received 1, sent 1
KAlive   : received 17, sent 17
Fdb sync : received 0, sent 0
Failover : received 0, sent 0
Conf     : received 1, sent 1
STP Total: received 4, sent 2
 Global  : received 3, sent 2
 Packet  : received 0, sent 0
```

```
 Instance: received 0, sent 0
 State   : received 1, sent 0
Connections established 1; dropped 0
Local host: 12.1.1.2, Local port: 46157
Foreign host: 12.1.1.1, Foreign port: 61000
remote_sysid: ea90.aecc.cc00
Switch2# show mac address-table
          Mac Address Table
------------------------------------------
(*) - Security Entry
Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
```

3 设备管理配置指导

3.1 STM 配置

3.1.1 简介

交换机表项管理（STM）是通过配置交换机的系统资源来支持优化特定功能。您可以选择一个配置文件提供来发挥系统的最大功能，例如，使用默认的配置文件以平衡资源；使用 VLAN 配置文件，以获得最大的 MAC 条目。为了在不同的场合下最大限度的利用 TCAM 资源，STM 提供了不同特性的系统优化功能。目前的版本中支持的 STM 模版包括：

- layer3：路由模板，支持最大数目的路由，通常应用在在网络中心的路由器或聚合层。
- layer2：VLAN 模板，支持单播 MAC 地址的最大数量。它通常会被选定为第 2 层交换机。
- default：默认模板，提供所有特性的平衡。
- ipv6：ipv6 模板，支持 ipv6 协议及 v4 和 v6 双栈使用。通常用于 ipv6 网络。

当您配置了（或当前使用的）STM 模式不存在于下一个要启动的 image 里时，那么当这个 image 启动的时候就会使用默认的硬编码配置，这个配置可能和正常的 default 模式是不一样的。

3.1.2 配置

通过配置指南来选择正确的 STM profiles：

- 修改配置后必须重启交换机。
- STM layer2 模板，一般适用于 2 层交换机且没有路由交换的场合。
- 当交换机上没有使能路由功能的时候就不需要切换到 layer3 模版

Switch# configure terminal	进入配置模式
Switch(config)# stm prefer layer3	设置 STM profile 为 layer3

Switch(config)# end	退出配置模式
Switch# reload	重启系统

3.1.3 命令验证

下面的例子显示了使用路由模版后的输出结果：

Switch# show stm prefer

```
Current profile is :default
   number of vlan instance                        : 1/4094
   number of unicast & multicast mac address      : 0/65536
   number of backhole mac address                 : 0/128
   number of max applied vlan mapping             : 0/1024
   number of mac based vlan class                 : 0/512
   number of ipv4 based vlan class                : 0/512
   number of dot1x mac based                      : 0/2048
   number of unicast ipv4 host routes             : 0/4096
   number of unicast ipv4 indirect routes         : 0/8192
   number of unicast ipv4 ecmp groups             : 0/256
   number of unicast ipv4 policy based routes     : 0/16
   number of unicast ip tunnel peers              : 0/8
   number of multicast ipv4 routes                : 0/1023
   number of multicast ipv4 routes member         : 0/1024
   number of ipv4 source guard entries            : 0/1024
   number of ipv4 acl/qos flow entries            : 0/511
   number of link aggregation (static & lacp)     : 0/55
The profile stored for use after the next reload is the layer3 profile.
   number of vlan instance                        : 1/4094
   number of unicast & multicast mac address      : 0/32768
   number of backhole mac address                 : 0/128
   number of max applied vlan mapping             : 0/1024
   number of mac based vlan class                 : 0/512
   number of ipv4 based vlan class                : 0/1024
   number of dot1x mac based                      : 0/512
   number of unicast ipv4 host routes             : 0/20480
   number of unicast ipv4 indirect routes         : 0/8192
   number of unicast ipv4 ecmp groups             : 0/256
   number of unicast ipv4 policy based routes     : 0/64
   number of unicast ip tunnel peers              : 0/8
   number of multicast ipv4 routes                : 0/1024
   number of multicast ipv4 routes member         : 0/1024
   number of ipv4 source guard entries            : 0/512
   number of ipv4 acl/qos flow entries            : 0/1536
   number of link aggregation (static & lacp)     : 0/55
   number of ipfix cache                          : 0/16384
```

3.2 系统日志配置

3.2.1 简介

系统消息可以保存在日志文件中，也可以发送到其他服务器设备。系统消息管理模块如下功能：

- 记录日志信息以便监测和故障排除
- 可以选择记录日志信息的类型
- 可以选择日志的目的地

默认情况下，交换机会记录重要的系统信息记录到其内部缓冲区，同时也会发送到系统控制台。用户可以指定保存的消息级别。消息都会添加发生时间，以提高实时调试和管理性。

您可以使用交换机的命令行界面（CLI）来读取系统消息，也可以通过将它保存到一个日志服务器的形式来获取消息。交换机的日志缓冲区最多可存储 1000 条信息。用户可以通过 Telnet 或控制台端口登录设备后打开终端监控来实时监控系统日志。

3.2.2 术语

Logging：当前日志配置

Show：显示日志配置

Levels：安全等级信息

Enable：开启日志保存到本地文件

Disable：关闭日志保存到本地文件

表3-1 系统消息类型

名称	定义
kern	kernel 消息
user	随机用户等级消息
mail	邮件系统
daemon	系统进程
auth	安全/验证消息
syslog	通过 syslogd 生成系统内部消息
lpr	行式打印机子系统
news	网络新闻子系统
uucp	UUCP 子系统
cron	时钟进程

名称	定义
authpriv	私有的安全/验证消息
ftp	FTP 进程

表3-2 安全等级的定义

严重等级	定义
emergency	系统无法使用
alert	必须立即采取行动
critical	严重事件
error	错误事件
warning	警告事件
notice	正常的，但重要的事件
information	信息
debug	调试级别的消息

3.2.3 配置日志服务器

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# logging server enable	启用 LOG Server
Switch(config)# logging server address 1.1.1.1	指定 LOG Server Ipv4 地址
Switch(config)# logging server address 2001:1000::2	指定 LOG Server IPv6 地址
Switch(config)# logging server severity debug	设置记录日志的等级
Switch(config)# logging server facility mail	设置日志消息

II. 命令验证

Switch# show logging

```
Current logging configuration:
============================================================
```

```
logging buffer 500
logging timestamp bsd
logging file enable
logging level file warning
logging level module debug
logging server enable
logging server severity debug
logging server facility mail
logging server address 1.1.1.1
logging server address 2001:1000::2
logging alarm-trap enable
logging alarm-trap level middle
logging merge enable
logging merge fifo-size 1024
logging merge timeout 10
logging operate disable
```

3.2.4 设置日志缓冲大小

默认情况下，日志缓冲区只保存 500 条最新的消息日志。用户也可以通过命令将范围改为 10 和 1000 之间的任何值。

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# logging buffer 700	设置日志缓冲区的大小为 700

II. 命令验证

Switch# show logging

```
Current logging configuration:
============================================================
logging buffer 700
logging timestamp bsd
logging file enable
logging level file warning
logging level module debug
logging server enable
logging server severity debug
logging server facility mail
logging server address 1.1.1.1
logging alarm-trap enable
logging alarm-trap level middle
logging merge enable
logging merge fifo-size 1024
logging merge timeout 10
logging operate disable
```

NOTE

您可以通过 show 命令来检查显示日志配置。如果配置 syslog 服务器，请确保连线正确，这个可以通过两台电脑互相 ping 来保证。同时用户也需要在日志服务器上配置 syslog 软件来接收日志。

3.3 镜像配置

3.3.1 简介

用户可以将某一端口或某一 VLAN 收、发的报文复制一份，从设备的另一个端口上发送出去，在这个端口连上测试仪或其他报文收集设备，可达到对原始报文进行捕获和分析的目的。

只有指定端口或指定 VLAN 收发的报文可以被镜像复制，这个端口或 VLAN 称作镜像源。 镜像功能监测的是镜像源，而不是流量的“源”。例如，当对一个 VLAN 的入方向流量做镜像的时候，从其他 VLAN 转发到这个 VLAN 来的报文是不会被复制的，而从这个 VLAN 收到、需要转发去别的 VLAN 的流量会被复制。

镜像功能不影响交换机源端口或源 VLAN 上原始的的网络流量；通过源端口发送或接收的报文可以被复制，复制出来那份流量将发送到指定的目的接口。

3.3.2 术语

和镜像配置相关的概念和术语描述如下：

镜像会话

镜像会话是一组镜像源和一个镜像目的的集合，其中镜像源可以是任意个端口或者 VLAN， 镜像目的可以是二层或者三层物理接口。

系统最多支持三组镜像会话。

镜像功能不应干扰正常业务。

在一组镜像会话中，如果镜像源的总流量超过了镜像目的接口的转发能力，例如用一个最大速率为 10Mbps 的目的端口去监控 100Mbps 的流量，将会产生丢包。

一个可以正常工作的镜像会话，需要配置镜像目的端口，以及至少一个镜像源。

流量类型

镜像会话包括三个流量类型：

接收方向镜像（RX）：对一个端口或 VLAN 做接收方向的镜像，原则是将这个端口或 VLAN 上的收到的流量，在系统对这些报文做任何修改和处理之前，尽可能完整、真实的复制出来。对于镜像源端口来说，有如下限制：CRC 错误的报文将不能被镜像复制。对于镜像源 VLAN 来说，有如下限制：BPDU, LACPDU, BMGPDU 报文，IP-

MAC 绑定检查不通过的报文，CRC 错误的报文，不能被镜像复制。除此以外的其他功能，例如 QOS 的修改 DSCP 值、VLAN translation、VLAN classification，ACL，VLAN’s ingress filter, MAC filter, STP, VLAN tag control, port security, unknown routing packets 等功能， 对报文进行修改或丢弃，都不应影响到接收方向的镜像功能。复制到目的端口的报文，应该和镜像源收到的报文完全一致。

发送方向镜像（TX）：对一个端口或 VLAN 做发送方向的镜像，原则是将这个端口或 VLAN 上的发送出去的流量，尽可能真实的复制出来。从端口或 VLAN 送出之前就被丢弃的报文，不会被镜像复制。当镜像源是 VLAN 的时候，有如下限制：来自 CPU 的报文不能被镜像复制。

双向镜像（BOTH）：在一个镜像会话中，用户可以监控同一个镜像源上接收和发送两个方向的报文流量。

镜像源

源端口（也称为被监测端口）是一个需要被监控或分析的二层或三层端口。源 VLAN（也称为被监测 VLAN）是一个需要被监控或分析的 VLAN。在一个镜像会话中，用户可以监控一个或多个镜像源上接收（RX）、发送（TX）、或双向的流量。系统支持任意多个镜像源端口（等同于系统最多可用端口数）和任意多个镜像源 VLAN（等同于系统最多可用 VLAN 数）。

源端口特性如下：

- 它可以是任何端口类型（例如，以太网端口）
- 只能在一个镜像会话中被监视
- 它不能是任意一个镜像会话的目的端口

每个镜像源端口或镜像源 VLAN 可以配置方向（入口、出口或双向）来监视。对于端口聚合组，监控方向将适用于该组中的所有物理端口

镜像源端口可以在相同或不同的 VLAN 中。

要配置镜像源 VLAN，必须先创建 VLAN 接口。

聚合组的成员端口不能单独配置成镜像源。

目的端口

每个镜像会话必须有一个目的端口（也称为监测端口），接收镜像功能复制出来的报文。

目的端口特性如下：

- 它必须和镜像源处在同一台设备上 。
- 可以是任何物理端口。
- 聚合组的成员端口不能配置成镜像目的端口。
- 只能在一个镜像会话中作为目的端口。
- 不能配置为任何镜像会话的镜像源端口。

- 端口不传输任何镜像功能以外的流量。
- 当镜像会话在执行时，该端口不能使用 STP。
- 该端口的所有其他系统功能的相关配置继续保留，但是不能工作。直到在此端口不再作为镜像会话的目的端口。
- 镜像目的端口不学习 MAC。
- 实时速率/双工状态可能会和显示的数值不一致。

3.3.3 配置

镜像配置如下

Switch #configure terminal	进入全局配置模式
Switch(config)# vlan database	进入 VLAN 配置模式
Switch(config-vlan)# vlan 10	创建 VLAN
Switch(config-vlan)# exit	退出 VLAN 配置模式
Switch(config)# interface vlan10	创建 VLAN 接口
Switch(config-if)# exit	退出 VLAN 接口配置模式
Switch(config)# interface eth-0-2	进入接口配置模式
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# exit	退出接口配置模式
Switch(config)# monitor session 1 destination interface eth-0-2	指定镜像会话的目的端口
Switch(config)# monitor session 1 source interface eth-0-1both	指定镜像会话和源端口和监控方向
Switch(config)# monitor session 1 source vlan 10 rx	指定镜像会话和源 VLAN
Switch(config)# end	退出全局配置模式
Switch# show monitor session 1	显示配置

3.3.4 命令验证

这个例子中创建了会话 1 用以监控源端口和源 VLAN 的流量。

可以使用显示会话命令查看配置

Switch # show monitor session 1

```
Session   1
```

```
----------
Status              :  Valid
Type                :  Local Session
Source Ports       :
  Receive Only      :
  Transmit Only     :
  Both              : eth-0-1
Source VLANs        :
  Receive Only      : 10
  Transmit Only     :
  Both              :
Destination Port    : eth-0-2
```

3.4 多目的端口镜像配置

3.4.1 简介

用户可以将某一端口的收、发的报文复制一份，从设备的另外几个端口上发送出去，在这些端口连上测试仪或其他报文收集设备，可达到对原始报文进行捕获和分析的目的。

只有指定端口的收发的报文可以被镜像复制，这个端口称作镜像源。 镜像功能监测的是镜像源，而不是流量的“源”。

镜像功能不影响交换机源端口上原始的的网络流量；通过源端口发送或接收的报文可以被复制，复制出来那份流量将发送到指定的目的接口。

3.4.2 配置

镜像配置如下

Switch #configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface eth-0-2	进入接口配置模式
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface eth-0-3	进入接口配置模式
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# exit	退出接口配置模式

Switch(config)# monitor session 1 destination group 1	创建镜像会话的目的端口组
Switch(config-monitor-d-group)# member eth-0-2	将接口 eth-0-2 加入目的端口组
Switch(config-monitor-d-group)# member eth-0-3	将接口 eth-0-3 加入目的端口组
Switch(config)# monitor session 1 source interface eth-0-1	指定镜像会话和源端口和监控方向
Switch(config)# end	退出全局配置模式
Switch# show monitor session 1	显示配置

3.4.3 命令验证

这个例子中创建了会话 1 用以监控源端口流量。

可以使用显示会话命令查看配置

Switch # show monitor session 1

```
Session   1
----------
Status             :  Valid
Type               :  Local Session
Source Ports      :
  Receive Only     :
  Transmit Only    :
  Both             : eth-0-1
Source VLANs       :
  Receive Only     :
  Transmit Only    :
  Both             :
Destination Port   : eth-0-2 eth-0-3
```

3.5 远程镜像配置

3.5.1 配置远程镜像

I. 简介

远程镜像功能，支持镜像的源端口（或源 VLAN）与镜像目的端口在不同的设备上，以实现整个网络中的跨设备远程监控。

II. 术语

和远程镜像配置相关的概念和术语描述如下：

远程镜像会话

是一组镜像源和一个远程镜像目的的集合，其中远程镜像目的包括一个物理出接口以及一个 VLAN。

在远程镜像源会话中，源端口和 VLAN 的概念和本地镜像一样。

一个远程镜像目的特性如下：

- 是一个指定的端口和一个 VLAN 的组合
- 远程 VLAN 范围在 2-4094，如果在系统中没有创建 VLAN，用户不能将这个 VLAN 作为远程镜像 VLAN
- 出端口应该是一个普通的物理端口，需要用户的配置来保证这个端口可以传输镜像报文，并且不被其他功能的流量所干扰。
- 镜像源的报文将被加上指定的远程 VLAN ID Tag，然后从指定的出接口发出去，到达远端设备上。
- 建议使用二层接口为远程镜像的目的端口，并且用户需要将这个端口加入到指定的远程 VLAN 中，否则镜像报文将不能成功发送出去。

I. 拓扑

如图 3-1 中，镜像源端口在 Switch A 上，不同的镜像会话通过将镜像复制报文封装在指定的 VLAN 中，通过网络传递到远端设备 Switch B 上。

远程镜像会话的镜像源和本地会话的镜像源一样，可以是一个端口，也可以是一个 VLAN。

远程镜像会话的目的必须指定一个物理端口作为出接口，同时指定一个 VLAN 用以封装镜像报文。

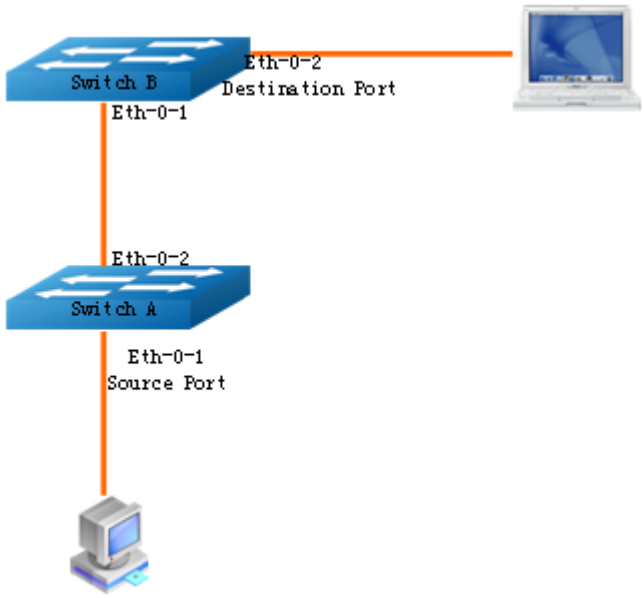


图3-1 远程镜像

配置

在 switch A 上的远程镜像配置如下表所示：

SwitchA# configure terminal	进入全局配置模式
SwitchA(config)# vlan database	进入 VLAN 配置模式
SwitchA(config-vlan)# vlan 10	创建 VLAN 10
SwitchA(config-vlan)# vlan 15	创建 VLAN 15
SwitchA(config-vlan)# exit	退出 VLAN 配置模式
SwitchA(config-if)# exit	退出接口配置模式
SwitchA(config)# interface eth-0-2	进入接口配置模式
SwitchA(config-if)# no shutdown	端口 up
SwitchA(config-if)# switchport mode trunk	端口模式为 trunk
SwitchA(config-if)# switchport trunk allowed vlan add 15	添加 eth-0-2 到 vlan 15
SwitchA(config-if)# exit	退出接口配置模式
SwitchA(config)# interface eth-0-1	进入接口配置模式
SwitchA(config-if)# switchport mode access	端口模式为 access

SwitchA(config-if)# switchport access vlan 10	添加 eth-0-1 到 vlan 10
SwitchA(config)# monitor session 1 destination remote vlan 15 interface eth-0-2	指定镜像会话、远程目的 vlan 和出端口
SwitchA(config)# monitor session 1 source interface eth-0-1 both	指定镜像会话和源端口（镜像端口）
SwitchA(config)# end	退出 EXEC 模式
SwitchA# show monitor session 1	显示配置

Switch B 配置

1. 使用“monitor session ID source vlan”命令得到一份远程镜像报文的拷贝，报文是加标签的。

SwitchB# configure terminal	进入全局配置模式
SwitchB(config)# vlan database	进入 VLAN 配置模式
SwitchB(config-vlan)# vlan 15	创建 VLAN 15
SwitchB(config-vlan)# exit	退出 VLAN 配置模式
SwitchB(config)# interface vlan15	进入 vlan 配置模式
SwitchB(config-if)# exit	退出 vlan 配置模式
SwitchB(config)# interface eth-0-2	进入接口配置模式
SwitchB(config-if)# no shutdown	端口 up
SwitchB(config-if)# switchport mode access	端口模式为 access
SwitchB(config-if)# switchport access vlan 15	添加 eth-0-2 到 vlan 15
SwitchB(config)# interface eth-0-1	进入接口配置模式
SwitchB(config-if)# no shutdown	端口 up
SwitchB(config-if)# switchport mode trunk	端口模式为 trunk
SwitchB(config-if)# switchport trunk allowed vlan add 15	添加 eth-0-1 到 vlan 15
SwitchB(config-if)# exit	退出接口配置模式
SwitchB(config)# monitor session 1 destination interface eth-0-2	指定镜像会话和目的地址
SwitchB(config)# monitor session 1 source vlan 15 rx	指定镜像会话和源 VLAN

SwitchB(config)# end	退出 EXEC 模式
SwitchB# show monitor session 1	显示配置

2. 使用 access 端口获取报文（在 Switch B 不需要配置任何镜像会话）

SwitchB# configure terminal	进入全局配置模式
SwitchB(config)# no spanning-tree enable	禁止 stp
SwitchB(config)# vlan database	进入 VLAN 配置模式
SwitchB(config-vlan)# vlan 15	创建 VLAN 15
SwitchB(config-vlan)# exit	退出 VLAN 配置模式
SwitchB(config)# interface eth-0-2	进入接口配置模式
SwitchB(config-if)# no shutdown	端口 up
SwitchB(config-if)# switchport mode access	端口模式为 access
SwitchB(config-if)# switchport access vlan 15	添加 eth-0-2 到 vlan 15
SwitchB(config)# interface eth-0-1	进入接口配置模式
SwitchB(config-if)# no shutdown	端口 up
SwitchB(config-if)# switchport mode trunk	端口模式为 trunk
SwitchB(config-if)# switchport trunk allowed vlan add 15	添加 eth-0-1 到 vlan 15
SwitchB(config-if)# exit	退出接口配置模式

3. 使用 trunk 端口获取目的报文（在 Switch B 不需要配置任何镜像会话）

SwitchB# configure terminal	进入全局配置模式
SwitchB(config)# no spanning-tree enable	禁止 stp
SwitchB(config)# vlan database	进入 VLAN 配置模式
SwitchB(config-vlan)# vlan 15	创建 VLAN 15
SwitchB(config-vlan)# exit	退出 VLAN 配置模式
SwitchB(config)# interface eth-0-2	进入接口配置模式
SwitchB(config-if)# no shutdown	端口 up
SwitchB(config-if)# switchport mode trunk	端口模式为 trunk

SwitchB(config-if)# switchport trunk allowed vlan add 15	添加 eth-0-2 到 vlan 15
SwitchB(config)# interface eth-0-1	进入接口配置模式
SwitchB(config-if)# no shutdown	端口 up
SwitchB(config-if)# switchport mode trunk	端口模式为 trunk
SwitchB(config-if)# switchport trunk allowed vlan add 15	添加 eth-0-1 到 vlan 15
SwitchB(config-if)# exit	退出接口配置模式

使用方法 2 和方法 3 会导致系统学习镜像报文的 MAC，有可能导致 FDB 表资源耗尽。

II. 命令验证

这个例子中创建了会话 1 用以监控源端口和源 VLAN 的流量。

可以使用显示会话命令查看配置。

SwitchA# show monitor session 1

```
Session   1
----------
Status             : Valid
Type                :  Remote Session
Source Ports       :
  Receive Only     :
  Transmit Only    :
  Both               : eth-0-1
Source VLANs       :
  Receive Only     :
  Transmit Only    :
  Both               :
Destination Port   : eth-0-2
Destination remote VLAN : 15
SwitchB# show monitor session 1
Session   1
----------
Status             : Valid
Type                :  Local Session
Source Ports       :
  Receive Only     :
  Transmit Only    :
  Both               :
Source VLANs       :
  Receive Only     : 15
  Transmit Only    :
```

```
  Both               :
Destination Port   : eth-0-2
```

3.5.2 配置 Mac Escape 远程镜像

I. 简介

MAC escape 是远程镜像的子功能，它只会影响远程镜像的结果。一个 MAC escape 条目包括一个 MAC 地址和一个 MAC 掩码。当 MAC escape 条目建立，MAC-DA 报文相匹配的条目不会镜像到远程目的 VLAN。用户可以通过 MAC escape 条目防止协议报文镜像到远端。

全局最多配置两个 MAC escape 条目。

II. 配置

SwitchA# configure terminal	进入全局配置模式
SwitchA(config)# monitor mac escape 00cc.12A9.33D8 ffff.ffff.ffff	创建 mac escape 条目
SwitchA(config)# monitor mac escape 00cc.159E.24F0 ffff.ffff.ffff	创建另一个 mac escape 条目
SwitchA(config)# end	退出全局配置模式
SwitchA# show monitor mac escape	显示 mac escape 配置

III. 命令验证

例子中建立了 mac escape 条目。

你可以通过显示命令来产看配置

SwitchA# show monitor mac escape

```
------------------------------------------------------------
           monitor rspan mac escape database
------------------------------------------------------------
count   : 2
------------------------------------------------------------
 Mac     : 00:cc:12:a9:33:d8
 Mask    : ff:ff:ff:ff:ff:ff
 Mac     : 00:cc:15:9e:24:f0
 Mask    : ff:ff:ff:ff:ff:ff
```

3.5.3 配置 ERSPAN 远程镜像

I. 简介

在一些数据处理场合，需要将交换机上某些端口上收发的数据通过三层网络发送给远端的分析仪进行分析。ERSPAN 可以通过 GRE 通道将数据添加 GRE 头部信息后发送

给分析仪，监控的过程中不影响数据的正常传输。如果发送的数据很多，ERSPAN 为了降低负载，可以设置将报文负载分担到多个目的分析设备，如 Figure 5-3 所示。

II. 拓扑

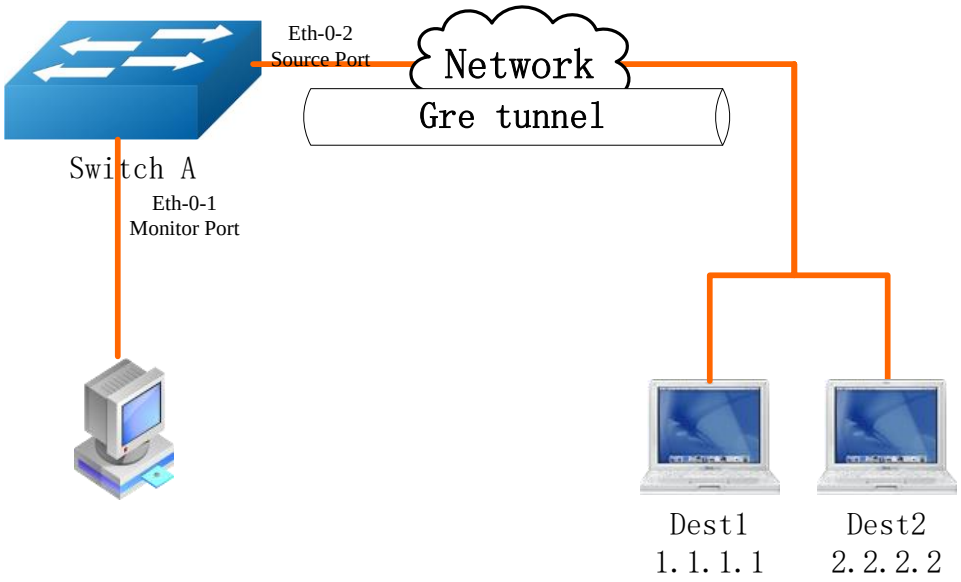


图3-2 Erspan

III. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)#no shutdown	端口 up
Switch(config-if)#exit	退出接口配置模式
Switch(config)#interface eth-0-2	进入接口配置模式
Switch(config-if)#no switchport	端口模式为 trunk
Switch(config-if)# ip address 10.10.10.1/24	配置端口 IP 地址
Switch(config-if)#no shutdown	端口 up
Switch(config-if)#exit	退出接口配置模式
Switch(config)# interface tunnel1	创建 tunnel1 并进入其配置模式
Switch(config-if)# tunnel source eth-0-2	指定 tunnel 源端口
Switch(config-if)# tunnel multi-destination 1.1.1.1	指定 tunnel 目的 IP 地址 1
Switch(config-if)# tunnel multi-destination 2.2.2.2	指定 tunnel 目的 IP 地址 2

Switch(config-if)# tunnel gre key 3333	设定 gre key
Switch(config-if)# tunnel extend-header (dst-lod-balance|)	设定 extend header
Switch(config-if)# tunnel mode (multi-dst-gre|gre)	设定 tunnel 模式
Switch(config-if)#exit	退出 tunnel1 配置模式
Switch(config)# arp 10.10.10.2 0000.0000.0001	arp 信息设定
Switch(config)# arp 11.11.11.2 0000.0000.0002	arp 信息设定
Switch(config)# ip route 1.1.1.0/24 10.10.10.2	路由信息设定
Switch(config)# ip route 2.2.2.0/24 10.10.10.2	路由信息设定
Switch(config)# monitor session 1 destination interface tunnel1	指定镜像会话的目的端口
Switch(config)# monitor session 1 source interface eth-0-1 both	指定镜像会话的源端口
Switch(config)# end	退出全局配置模式
Switch# show monitor session 1	显示会话 1 的配置信息
Switch#show running-config interface tunnel 1	显示 tunnel 配置信息

IV. 命令验证

```
SwitchA# show monitor mac escape
Session   1
----------
Status             : Valid
Type               : Local Session
Source Ports       :
  Receive Only     :
  Transmit Only    :
  Both             : eth-0-1
Source VLANs       :
  Receive Only     :
  Transmit Only    :
  Both             :
Destination Port   : tunnel1
SwitchA# show running-config interface tunnel 1
Building configuration...
!
interface tunnel1
 tunnel source eth-0-2
 tunnel multi-destination 1.1.1.1
 tunnel multi-destination 2.2.2.2
 tunnel gre key 3333
```

```
 tunnel multi-dst-gre extend-header
 tunnel mode multi-dst-gre
!
```

3.6 CPU 镜像目的口配置

3.6.1 简介

用户可以将某一端口或某一 VLAN 收、发的报文复制一份，从设备的另一个端口上发送出去，在这个端口连上测试仪或其他报文收集设备，可达到对原始报文进行捕获和分析的目的。当该端口无法连接上测试仪或者其他报文收集设备，或者设备资源紧张时，需要将被复制报文发送到 CPU 并被保存下来，便于用户或程序员快速分析报文。将报文复制一份送到 CPU 是一种解决硬件资源紧张的问题。目前镜像报文上送 CPU 的速率是系统默认限制速率，也可以用户指定限速速率。

3.6.2 配置

1. 配置 cpu 为镜像目的端口，配置 eth-0-1 为镜像源，方向为 both；配置 mirror cpu 的内存存储包的大小为 100，单位为包的个数；配置 mirror cpu 的限速速率为 128pps

Switch# configure terminal	进入配置模式
Switch(config)# monitor session 1 destination cpu	配置 cpu 为 session 1 镜像目的口
Switch(config)# monitor session 1 source interface eth-0-1 both	配置 eth-0-1 为 session 1 的镜像源，方向为 both(缺省值也为 both)
Switch(config)# monitor cpu set packet buffer 100	配置 mirror cpu 的内存存储空间大小为 100 个包，最多 100 个包。
Switch(config)# cpu-traffic-limit reason mirror-to-cpu rate 128	配置 mirror 到 cpu 的包的速率为 128pps
Switch# exit	退出全局配置模式

2. 配置 mirror cpu 的抓包策略为 drop，其中 replace 为默认值。

Switch(config)# monitor cpu capture strategy drop	配置 mirror cpu 的抓包策略为 drop。(即：当内存空间写满之后，丢弃新包)
Switch(config)# monitor cpu capture strategy replace	配置 mirror cpu 的抓包策略为 replace。(即：当内存空间写满之后，新包替换最旧包)

3.6.3 命令验证

1.示例中创建了会话 1 用以监控源端口 eth-0-1 的流量，并通过 show 命令查看 mirror to cpu 的报文。可以使用显示会话命令查看配置：

```
Switch# show monitor session 1
DUT1# show monitor session 1
Session   1
----------
Status             : Valid
Type               : Cpu Session
Source Ports       :
  Receive Only     :
  Transmit Only    :
  Both             : eth-0-1
Source VLANs       :
  Receive Only     :
  Transmit Only    :
  Both             :
Destination Port   : cpu
```

2. 查看报文 mirror 到 cpu 后内存存储的包

```
DUT1# show monitor cpu packet all
-----------------show all mirror to cpu packet info-----------------
packet: 1
Source port: eth-0-1
MACDA:264e.ad52.d800, MACSA:0000.0000.1111
vlan tag:100
IPv4 Packet, IP Protocol is 0
IPDA:3.3.3.3, IPSA: 10.0.0.2
Data length: 47
Data:
 264e ad52 d800 0000 0000 1111 8100 0064
 0800 4500 001d 0001 0000 4000 6ad9 0a00
 0002 0303 0303 6365 6e74 6563 796f  75
```

3. 查看配置 mirror cpu 内存 buffer 大小

```
DUT1# show monitor cpu packet buffer
--------------------show packet buffer size ---------------------
The mirror-to-cpu packet buffer size of user set is: 100
```

4. 查看配置 mirror cpu 的报文上 cpu 的速率

```
DUT1# show cpu traffic-limit | include mirror-to-cpu
mirror-to-cpu              128           0
```

5. 查看 mirror cpu 报文的存储文件

```
DUT1# ls flash:/mirror
Directory of flash:/mirror

total 8
```

```
-rw-r-----  1 2287 Dec 23 01:16 MirCpuPkt-2016-12-23-01-15-54.txt
-rw-r-----  1 2568 Jan  3 11:41 MirCpuPkt-2017-01-03-11-41-33.txt
14.8T bytes total (7.9T bytes free)
DUT1# more flash:/mirror/ MirCpuPkt-2017-01-03-11-41-33.txt
sequence  srcPort
1          eth-0-1
++++++++1483443444:648884
8c 1d cd 93 51 00 00 00 00 00 11 11 08 00 45 00
00 26 00 01 00 00 40 00 72 d0 01 01 01 01 03 03
03 03 63 65 6e 74 65 63 79 6f 75 63 65 6e 74 65
63 79 6f 75
--------
sequence  srcPort
2          eth-0-1
++++++++1483443445:546440
8c 1d cd 93 51 00 00 00 00 00 11 11 08 00 45 00
00 26 00 01 00 00 40 00 72 d0 01 01 01 01 03 03
03 03 63 65 6e 74 65 63 79 6f 75 63 65 6e 74 65
63 79 6f 75
```

6.在转换成 pcap 文件后，可以通过 wireshark 打开

```
DUT1#ls flash:/mirror
Directory of flash:/mirror

total 12
-rw-r-----  1 2287 Dec 23 01:16 MirCpuPkt-2016-12-23-01-15-54.txt
-rw-r-----  1 2568 Jan  3 11:41 MirCpuPkt-2017-01-03-11-41-33.txt
-rw-r--r--  1  704 Jan  3 13:07 test.pcap
14.8T bytes total (7.9T bytes free)
```

7.查看 mirror cpu 的抓包策略

```
DUT1# show monitor cpu capture strategy
The capture strategy of cpu mirror is: replace (add new packet and remove oldest
 packet when buffer is full)
```

3.7 CPU 镜像源配置

3.7.1 简介

用户可以将 CPU 作为镜像源配置，包含 ingress 方向和 egress 方向以及 both。当需要将上报 cpu 报文或者 cpu 下发的报文镜像复制某一个端口时，可以启用 CPU 镜像源配置，值得注意的是镜像复制出来的报文是 cpu-traffic-limit 限速之前的。目前只允许 session 1 可以配置 cpu mirror source。

3.7.2 配置

1. 将 CPU 设置为镜像源端口，在配置模式下进行：

Switch# configure terminal	进入配置模式
Switch(config)# monitor session 1 source cpu both	配置 cpu 为 session 1 镜像源， 方向为 both
Switch(config)# monitor session 1 destination interface eth-0-1	配置物理口 eth-0-1 为 session 1 的镜像目的端口
Switch# exit	退出全局配置模式

3.7.3 命令验证

配置 cpu 为 session 1 的镜像源，配置 eth-0-1 为 session 1 的镜像目的端口

```
DUT1# show monitor session 1
Session   1
----------
Status              : Valid
Type                : Cpu Session
Source Ports        :
  Receive Only      :
  Transmit Only     :
  Both              : cpu
Source VLANs        :
  Receive Only      :
  Transmit Only     :
  Both              :
Destination Port    : eth-0-1
```

3.8 设备管理配置

3.8.1 简介

用户可以通过管理端口管理交换机。交换机有 2 类管理端口：以太网口和串口。

3.8.2 配置串口

I. 配置

交换机的默认串口配置如下：

- 波特率为 **115200**
- 数据位为 **8**
- 停止位为 **1**
- **无**奇偶校验

在配置交换机之前，请先确认已经将交换机串口与 PC 或其他终端的串口相连，且 PC 或终端的串口配置与上述交换机串口默认配置一致。当登录到交换机上后，可以修改串口配置参数。

Switch# configure terminal	进入全局配置模式
Switch(config)# line console 0	进入串口配置模式
Switch(config-line)# speed 19200	设置串口波特率

II. 命令验证

完成上述配置后，串口参数已经被修改，此时 PC 或终端无法再通过串口配置交换机。必须修改 PC 或终端的串口属性，将波特率从 115200 修改为 19200，才能够重新连上交换机进行配置。

3.8.3 配置带外管理端口

为了通过带外管理端口配置交换机，必须先通过串口为带外管理端口配置管理 IP 地址。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# management ip address A.B.C.D/M	配置交换机管理口 IPv4 地址 A.B.C.D – 管理口 IPv4 地址 M - 子网掩码
Switch(config)# management ipv6 address A:B::C/M	配置交换机管理口 IPv6 地址 A:B::C – 管理口 IPv6 地址 M - 子网掩码
Switch(config)# exit	退出
Switch# show management ip address	验证设置的管理口 IPv4 地址
Switch# show management ipv6 address	验证设置的管理口 IPv6 地址

II. 命令验证

完成上述配置后，可在命令行中输入“**show management ip address**”或“**show management ipv6 address**”来查看配置的管理口 IP 地址。也可以通过 PC 执行 ping A.B.C.D 指令来验证该 IP 地址。

Switch# show management ip address

```
Management IP address is:  A.B.C.D/M
```

```
  Gateway: 0.0.0.0
Switch # show management ipv6 address
  Management IPv6 address is:  2001:1000::1/96
  Gateway: ::
```

3.8.4 配置温度管理

交换机支持温度告警管理功能。用户可以设置 3 个温度阀值：低温告警阀值，高温告警阀值，超高温断电保护阀值。当交换机温度低于低温告警阀值，或者高于高温告警阀值，交换机将自动产生告警信息。当交换机温度高于超高温断电保护阀值，交换机将通过自动切断电源来保护系统。

I. 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# temperature 5 70 90	设置新的温度阀值，单位为摄氏度
Switch(config)#exit	退出
Switch#show environment	验证设置的温度阀值

II. 命令验证

Switch# show environment

```
-------------------------------------------------------
Sensor status (Degree Centigrade):
Index Temperature  Lower_alarm  Upper_alarm Critical_limit
1     50           5            75          90
```

3.8.5 配置风扇管理

交换机支持自动管理风扇。当风扇盘不在位或者风扇坏掉，交换机能自动产生告警信息。如果风扇盘支持风扇速度调节，交换机将根据系统内部实时温度值自动调节风扇转速。交换机风扇速度调节也有 3 个温度阀值：Tlow=50 度, Thigh=65 度, Tcrit=80 度。当实时温度<Tlow 时，风扇将停止转动；当 Tlow<=实时温度<Thigh 时，风扇将以 30%的速率转动；当 Thigh<=实时温度<Tcrit 时，风扇将以 70%的速率转动；当 Tcrit<=实时温度时，风扇将全速转动。并且这里风扇自动调节还支持温度迟滞 Thyst=2 度。当之前的温度高于某阀值，风扇转速上升一个级别，现在温度又下降到低于该阀值时，风扇转速不会立即下降一个级别，必须等到实时温度比该阀值还低 Thyst（2 度）时，才会调节风扇转速，下降一个级别。举例如下：

当前温度为 58 摄氏度，风扇转速为 30%；(Tlow<58<Thigh)。

当温度上升到 65 摄氏度时，风扇转速自动调节为 70%；（Thigh==65)

当温度又下降到 63 摄氏度时，风扇转速仍旧为 70%；(Thigh-Thyst ==63)

当温度下降到 62 摄氏度时，风扇转速降为 3%；(62<Thigh-Thyst)

I. 配置

Tlow、Thigh、Tcrit 和 Thyst 以及对应的风扇转速都是系统预定义的，不支持用户调节。

II. 命令验证

Switch# show environment

```
Fan tray status:
Index     Status
1         PRESENT
FanIndex   Status SpeedRate Mode
1-1        OK      30%      Auto
1-2        OK      30%      Auto
1-3        OK      30%      Auto
1-4        OK      30%      Auto
-------------------------------------------------------
```

3.8.6 配置电源管理

交换机支持自动电源管理。当某个电源坏掉（双电源模式时）或者电源风扇坏掉，交换机能够自动发出告警信息。当电源模块拔插时，交换机也会发出通告信息。

I. 命令验证

用户可以通过命令行指令来查看电源的运行状态

Switch# show environment

```
-------------------------------------------------------
Power status:
Index     Status    Power     Type      Fans      Control
1         PRESENT   OK        AC        -         -
2         ABSENT    -         -         -         -
3         PRESENT   OK        DC(PoE)   -         -
-------------------------------------------------------
```

3.8.7 配置光模块

交换机支持管理光模块信息，这些管理信息包括基本信息和诊断信息。其中基本信息包括光模块类型、生产厂商名称、序列号、产品号以及相应支持的光波长和链路长度。诊断信息包括光模块的实时温度、电压、电流、发送光功率和接收光功率以及这些实时信息对应的厂商预定义正常工作范围、提醒阀值和告警阀值。当光模块拔插或者实时信息超出正常工作范围，交换机将自动发出通告或告警信息。

I. 命令验证

用户可以通过命令行指令来查看电源的运行状态

Switch# show transceiver detail

```
Port eth-1-2 transceiver info:
Transceiver Type: 10G Base-SR
```

```
Transceiver Vendor Name : OEM
Transceiver PN          : SFP-10GB-SR
Transceiver S/N         : 201033PST1077C
Transceiver Output Wavelength: 850 nm
Supported Link Type and Length:
      Link Length for 50/125um multi-mode fiber: 80 m
      Link Length for 62.5/125um multi-mode fiber: 30 m
-------------------------------------------------------------------------------
Transceiver is internally calibrated.
mA: milliamperes, dBm: decibels (milliwatts), NA or N/A: not applicable.
++ : high alarm, +  : high warning, -  : low warning, -- : low alarm.
The threshold values are calibrated.
-------------------------------------------------------------------------------
             High Alarm   High Warn   Low Warn    Low Alarm
Temperature   Threshold    Threshold    Threshold     Threshold
Port  (Celsius)       (Celsius)     (Celsius)     (Celsius)       (Celsius)
--------- ------------------  ----------  ----------  ----------  ----------
eth-1-2  25.92       95.00         90.00      -20.00        -25.00
-------------------------------------------------------------------------------------
---------------
                   High Alarm   High Warn   Low Warn    Low Alarm
       Voltage     Threshold     Threshold   Threshold     Threshold
Port     (Volts)       (Volts)       (Volts)       (Volts)        (Volts)
--------- ----------------- ----------------  ----------------  ---------------  --
-------------
eth-1-2   3.32        3.80         3.70        2.90          2.80
-------------------------------------------------------------------------------------
----------------
                    High Alarm   High Warn   Low Warn    Low Alarm
         Current     Threshold     Threshold   Threshold     Threshold
Port   (milliamperes)  (mA)          (mA)         (mA)         (mA)
--------- ------------------  ---------------  ----------------  -----------------
-------------
eth-1-2    6.41        20.00       18.00       1.00           0.50
-------------------------------------------------------------------------------------
------------------
        Optical      High Alarm   High Warn   Low Warn    Low Alarm
     Transmit Power   Threshold    Threshold    Threshold     Threshold
Port   (dBm)          (dBm)        (dBm)         (dBm)        (dBm)
--------- ------------------  ---------------  ----------------  ----------------
---------------
eth-1-2    -2.41        2.01        1.00         -6.99         -7.96
-------------------------------------------------------------------------------------
--------------------
        Optical      High Alarm   High Warn    Low Warn    Low Alarm
     Receive Power    Threshold    Threshold     Threshold     Threshold
Port    (dBm)          (dBm)        (dBm)         (dBm)         (dBm)
--------- ------------------  ---------------  -----------------  ----------------
----------------
eth-1-2     -12      -     1.00        0.00        -19.00      -20.00
-------------------------------------------------------------------------------------
--------------------
```

3.8.8 升级 Bootrom 程序

交换机支持在线升级 Bootrom 程序，当升级完后，必须重启才能生效。

I. 配置

Switch# copy mgmt-if tftp://10.10.29.160/ bootrom.bin flash:/boot/	从 TFTP 服务器拷贝 Bootrom 程序到本地 flash 存储介质上
Switch# configure termina	进入全局配置模式
Switch(config)# update bootrom flash:/boot/bootrom.bin	升级指定的 Bootrom 程序
Switch(config)# exit	退出
Switch# reboot	重启系统

II. 命令验证

当完成上述配置，系统重启结束，可查看系统当前运行的 bootrom 版本号。

```
Switch# show version
……
EPLD Version is 1
BootRom Version is 3.0.2
```

3.8.9 升级 EPLD 程序

交换机支持在线升级 EPLD 程序，当升级完成后，必须断电重启系统，否则系统将无法正常工作。

I. 配置

Switch# copy mgmt-if tftp://10.10.29.160/ vme_v1.0 flash:/boot/ vme_v1.0	从 tftp 服务器拷贝 EPLD 程序到本地 flash 存储介质上
Switch# configure termina	进入全局配置模式
Switch(config)# update epld flash:/boot/ vme_v1.0	升级指定 EPLD 程序
Switch(config)# exit	退出
Switch# reboot	重启系统

II. 命令验证

当完成上述配置，系统重启结束，可查看系统当前运行的 EPLD 版本号

Switch# show version

.....

```
EPLD Version is 1
BootRom Version is 3.0.2
```

3.9 Bootrom 配置

3.9.1 简介

U-boot 的主要功能是简单地初始化板子和在启动时加载系统镜像。在 U-boot 模式下，你可以使用一些必要的命令。

U-boot 既能从 TFTP 服务器上加载系统镜像，又能从硬盘里加载，例如 flash。如果你在从 TFTP 服务器上启动系统，你可以配置本地设备和指定 TFTP 服务器的 IP 地址。

3.9.2 从 TFTP 服务器上加载镜像

I. 配置

步骤 1 从 TFTP 服务器上加载镜像 OS-ms-v3.1.9.it.r.bin 启动系统，具体如下。

bootrom:> setenv bootcmd boot_tftp OS-ms-v3.1.9.it.r.bin	从 TFTP 服务器上加载镜像 OS-ms-v3.1.9.it.r.bin 启动系统
bootrom:> saveenv	在本地保存配置
bootrom:> reset	重启板子

步骤 2 不需要密码从 TFTP 服务器上加载镜像 OS-ms-v3.1.9.it.r.bin 启动系统，具体如下。

bootrom:> setenv bootcmd boot_tftp_nopass OS-ms-v3.1.9.it.r.bin	不需要密码从 TFTP 服务器上加载镜像 OS-ms-v3.1.9.it.r.bin 启动系统
bootrom:> saveenv	在本地保存配置
bootrom:> reset	重启板子

步骤 3 从 TFTP 服务器上加载镜像 OS-ms-v3.1.9.it.r.bin 后直接重启板子，具体如下。

bootrom:> boot_tftp OS-ms-v3.1.9.it.r.bin	从 TFTP 服务器上加载镜像 OS-ms-v3.1.9.it.r.bin 后直接重启板子

步骤 4 不需要密码从 TFTP 服务器上加载镜像 OS-ms-v3.1.9.it.r.bin 后直接重启板子，具体如下。

bootrom:> boot_tftp_nopass OS-ms-v3.1.9.it.r.bin	不需要密码从 TFTP 服务器上加载镜像 OS-ms-v3.1.9.it.r.bin 后直接重启板子

II. 命令验证

在以上配置命令之后，你可以验证配置信息。

```
bootrom:> reset
……..
TFTP from server 10.10.29.160; our IP address is 10.10.29.118
Filename 'OS-ms-v3.1.9.it.r.bin'.
Load address: 0xaa00000
Loading: octeth0: Up 100 Mbps Full duplex (port  0)
#################################################################
         ######################
done
Bytes transferred = 12314539 (bbe7ab hex), 1829 Kbytes/sec
```

3.9.3 从 Flash 上加载镜像

I. 配置

步骤 1 从 flash 加载镜像 OS-ms-v3.1.9.it.r.bin 启动系统，具体如下。

bootrom:> setenv bootcmd boot_flash OS-ms-v3.1.9.it.r.bin	从 flash 加载镜像 OS-ms-v3.1.9.it.r.bin 启动系统
bootrom:> saveenv	在本地保存配置
bootrom:> reset	重启板子

步骤 2 从 flash 加载镜像 OS-ms-v3.1.9.it.r.bin 启动系统，并将恢复系统默认登录密码配置，具体如下。

bootrom:> setenv bootcmd boot_flash_nopass OS-ms-v3.1.9.it.r.bin	不需要密码从 flash 加载镜像 OS-ms-v3.1.9.it.r.bin 启动系统
bootrom:> saveenv	在本地保存配置.
bootrom:> reset	重启板子
Do you want to revert to the default config file ? [Y|N|E]:Y	Y：恢复默认配置文件 N：仅恢复默认登录密码配置 E：退出设置

步骤 3 从 flash 加载镜像 OS-ms-v3.1.9.it.r.bin 后直接启动系统，具体如下。

bootrom:> boot_flash OS-ms-v3.1.9.it.r.bin	从 flash 加载镜像 OS-ms-v3.1.9.it.r.bin 后直接启动系统

步骤 4 从 flash 加载镜像 OS-ms-v3.1.9.it.r.bin from flash 后直接启动系统，并将恢复系统默认登录密码配置，具体如下。

bootrom:> boot_flash_nopass OS-ms-v3.1.9.it.r.bin	不需要密码从 flash 加载镜像 OS-ms-v3.1.9.it.r.bin from flash 后直接启动系统
Do you want to revert to the default config file ? [Y|N|E]:Y	Y：恢复默认配置文件 N：仅恢复默认登录密码配置 E：退出设置

II. 命令验证

在以上配置命令之后，你可以验证配置信息。

bootrom:> reset

.....

```
Do you want to  revert  to the default config file ? [Y|N|E]:Y
### JFFS2 loading '/boot/OS-ms-v3.1.9.it.r.bin' to 0xaa00000
Scanning JFFS2 FS: . done.
### JFFS2 load complete: 12314539 bytes loaded to 0xaa00000
## Booting image at 0aa00000 ...
   Verifying Checksum ... OK
   Uncompressing Kernel Image ... OK
```

.....

3.9.4 配置 Boot IP

I. 配置

步骤 1 设置本地设备的 IP，具体如下。

bootrom:> setenv ipaddr 10.10.29.101	设置本地设备的 IP
bootrom:> saveenv	在本地保存配置

步骤 2 指定 TFTP 服务器 IP，具体如下。

bootrom:> setenv serverip 10.10.29.160	指定 TFTP 服务器 IP
bootrom:> saveenv	在本地保存配置

II. 命令验证

在以上配置命令之后，你可以验证配置信息。

bootrom:> printenv

```
printenv
bootdelay=5
baudrate=9600
download_baudrate=9600
……
stderr=serial
ipaddr=10.10.29.101
ipserver=10.10.29.160
Environment size: 856/2044 bytes
```

3.9.5 在线升级 Bootrom

I. 配置

bootrom:> upgrade_uboot bootrom.bin	从 TFTP 服务器在线升级 Bootrom

II. 命令验证

在以上配置命令之后，你可以验证配置信息。

bootrom:> version

```
version
Bootrom 3.0.3 (Development build) (Build time: Aug  4 2011 - 11:47:06)
```

3.9.6 设定 bootrom 的网关

I. 配置

步骤 1 设置本地设备的网关，具体如下。

bootrom:> setenv gatewayip 10.10.37.1	设定交换机 bootrom 的网关地址
bootrom:> saveenv	在本地保存配置

步骤 2 设置本地设备的子网掩码，具体如下。

bootrom:> setenv netmask 255.255.255.0	设定子网掩码
bootrom:> saveenv	在本地保存配置

II. 命令验证

在以上配置命令之后，你可以验证配置信息：

bootrom:> printenv

```
printenv
bootdelay=5
baudrate=9600
download_baudrate=9600
……
stderr=serial
gatewayip=10.10.38.1
netmask=255.255.255.0
Environment size: 856/2044 bytes
```

3.10 启动诊断配置

3.10.1 简介

启动诊断可以在交换机重新启动后，帮助用户诊断交换机的各个硬件组件是否工作正常。其中诊断项包括：EPLD，EEPROM，PHY，MAC 等。

3.10.2 配置

配置启动诊断的流程如下表所示。

Switch# configure terminal	进入配置模式
Switch(config)# diagnostic bootup level minimal	设置诊断等级为 minimal
Switch(config)# exit	推出诊断模式
Switch# show diagnostic bootup level	查看配置的诊断等级是否正确
Switch# reboot	重启系统

3.10.3 命令验证

下面的例子显示了如何查看启动诊断的结果

Switch# show diagnostic bootup result detail

```
##########################################################
```

```
Item Name                    Attribute Result Time(usec)
1    EPLD TEST               C         Pass   57
2    EEPROM0 TEST            C         Pass   101262
3    PHY TEST                C         Pass   1161
4    FAN TEST                C         Pass   4668
5    SENSOR TEST             C         Pass   5472
6    PSU TEST                C         Pass   1370
7    L2 UCAST FUNC TEST      C         Pass   40126
```

3.11 Bootstrap 配置

3.11.1 简介

Bootstrap 是一种智能初始化配置方法。在设置启用 Bootstrap 功能后，交换机启动时发现没有 startup-config.conf 文件或 bootstrap 功能开关开启，则开始从 tftp 服务器上下载配置文件或 image 文件。如果发现需要下载不同版本的 image 文件，则需要重新启动系统。

需要注意的是我们是通过 python 脚本文件控制交换机下载的 image 文件和配置文件。交换机将会从 python 格式的脚本文件里，找到自己需要下载的文件。脚本文件的名称为 bootstrap.py，需要事先根据需求配置好，需要填入的信息如下：

```
options = {
  # tftp server ip
  "hostname": "192.168.1.254",

  # new target system image name
  "target_system_image": "XXXXXX.bin",

  # new target system image md5sum
  "image_md5sum": "f7ea31029b33d3f77d7c2156a814e6d7",

  # tftp server config path
  #config_sw=1 get config, config_sw=0 no get config
  "config_sw": 1,
  "config_path": "/tftpboot/",

  # tftp server target system image path
  "target_image_path": "/tftpboot/",
```

```
    "destination_path": "/mnt/flash/boot/",
}
```

其中红色字体部分需要用户自行设置，

hostname：tftp 服务器的 ip 地址。

target_system_image：需要升级的版本文件名称。

image_md5sum：image 文件的 MD5 值，若为空则表示不进行 image 的 MD5 校验，否则进行 image 的 MD5 校验。

config_sw：下载配置文件的开关，1 表示下载配置文件，0 表示不下载配置文件。

config_path：下载配置文件的路径，注意这里路径不需要填写配置文件名称。

target_image_path：下载 image 文件的路径，同样这里不需要填写 image 文件名称。

配置好 python 文件后，当 python 脚本中 target_system_image 和交换机中版本一致时，不进行 image 下载操作，只做配置文件下载并更新配置操作。当版本不一致时，先进行 image 文件下载，更新 image 版本后重启，待重启后进行配置下载更新操作。

注意：当完成了一次配置更新后，如果还需要重复应用该配置文件则需要手动删除 /mnt/flash/boot/increment-config.cfg 文件，再进行 bootstrap 功能。当没有 startup-config.conf 文件即空配置启动时会自动删除 increment-config.cfg 文件。

3.11.2 拓扑

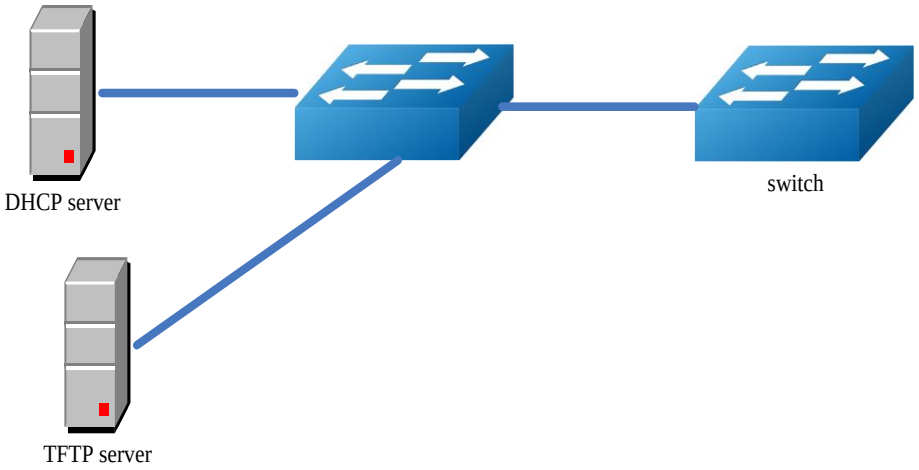


图 1-3 Bootstrap 拓扑

上图为测试 Bootstrap 的网络拓扑，需要两台交换机和两台 pc 构建测试环境。switch 是我们启用 Bootstrap 功能的交换机。需要注意的是, 上图中 DHCP server 提供的 TFTP server 地址必须是 switch 可以直接连接或者通过路由器连接的。

3.11.3 配置

配置 Bootstrap

Switch#configure terminal	进入全局配置模式
Switch(config)#bootstrap enable	设置启用 Bootstrap
Switch (config)#exit	退出接口配置模式

默认是关闭 Bootstrap 功能的，所以当 startup-config.conf 文件不存在或开启了 bootstrap 开关时，交换机才会在启动时开始 Bootstrap 工作流程。也可以手动删除 startup-config.conf 文件，这样在下次启动时，Bootstrap 就会工作。

具体的配置步骤如下：

1.配置 DHCP server，必须要设置 option 66：tftp-server name 和 option67：bootfile-name 选项；其中 option 66 字段必须设置为 IP 地址格式，如 10.0.0.1 或者 http://10.0.0.1。option 67 字段为下载 python 脚本文件的路径名称，如/tftpboot/boostrap.py。

2.通过 DHCP server 端的信息从 tftp 服务器上下载脚本文件 bootstrap.py，该 python 脚本内容需根据客户需求事先定义好，将需要升级的 image 文件，配置文件放到 python 脚本中对应的 tftp server 上。

Image 文件名需要和 python 脚本文件中填入的 target_system_image 一致。

配置文件名称的格式：设备 SN 号.cfg，例如：U50R9390071.cfg

也可以为：设备 MAC 地址.cfg 例如：6cec5a084e93.cfg

注意：设备 SN 号区分大小写，设备 MAC 地址为设备管理口 MAC 地址且字母需全为小写。（可用 show management interface 命令查看）

3.确保交换机没有 startup-config.conf 文件或者开启了 bootstrap 功能开关。

注意：当存在 startup-config.conf 文件但开启了 bootstrap 开关时，管理口需配成 DHCP 模式后再重启，空配置启动时无此限制。

4.启动或重启系统。

3.11.4 命令验证

检查 Bootstrap 配置

Switch# show running-config

```
!
bootstrap enable
!
```

```
line con 0
 no line-password
 no login
line vty 0 7
 exec-timeout 35791 0
 privilege level 4
 no line-password
 no login
!
end
```

3.12 重启记录

3.12.1 简介

Centec 交换机支持显示重启记录，从重启记录中可以区分出来板子是掉电重启，还是手动重启，或者是其他原因导致的重启。用户也可以通过一条命令来清除重启记录。

3.12.2 命令验证

如下命令显示重启记录

```
Switch# show reboot-info
Times      Reboot Type     Reboot Time(DST)
1          MANUAL          2000/01/01 01:21:35
2          MANUAL          2000/01/01 02:07:52
3          MANUAL          2000/01/01 02:24:59
4          MANUAL          2000/01/01 03:28:58
5          MANUAL          2000/01/01 03:43:02
6          MANUAL          2000/01/01 03:49:51
7          MANUAL          2000/01/01 04:01:23
8          MANUAL          2000/01/01 04:42:40
9          MANUAL          2000/01/01 04:49:27
10         MANUAL          2000/01/01 20:59:20
```

如下命令清除重启记录

```
Switch(config)# reset reboot-info
```

3.12.3 注意

使用该命令最多显示 10 条重启记录，如果要查看更多的重启记录，可以在如下的文件中查看：flash:/reboot-info/reboot_info.log

显示结果说明如下：

重启类型	说明
POWER	断电重启

MANUAL	系统下手动 reboot/reload 重启
HIGH-TMPR	高温异常重启
BHMDOG	BHM 看门狗重启，用于监控系统各个功能模块
LCMDOG	LCM 看门狗重启，用于监控 LC
SCHEDULE	定时重启
SNMP-RELOAD	SNMP 重启
HALFAIL	HAGT 与 HSRV 通讯异常重启，需要 stack 功能开启
ABNORMAL	系统非正常方式重启，包括 shell 下的 reboot
CTCINTR	按键重启
LCATTACH	LC 匹配异常重启
OTHER	其他重启

4 网络管理配置指导

4.1 网络诊断配置

4.1.1 简介

Ping 是一个计算机网络的管理工具，用于测试一台主机通过 IP 协议的可达性和衡量每次从源到目的主机的时间(round trip tiem)。它的名字来源于主动声纳的术语。

Ping 通过向目的主机发送 ICMP echo 请求报文，等待 ICMP 回应来实现。在运作过程中，它测量每一次发送到接收到响应的时间间隔(round trip time)，并且记录所有的丢包。测试结果会用一个统计汇总数据来显示接收到的所有报文，包括最小，最大和平均的 round-trip time，有时会打印平均的标准偏差值。

Traceroute 是一个在 IP 网上用于测量路由选路和报文传输时间的工具。

Traceroute 向目的主机发送一个 ICMP 序列报文，通过 TTL 参数跟踪通过的中间路由。中间路由器减少通过报文的 TTL 参数值，当 TTL 值为 0 时丢弃报文并回送一个 ICMP 错误消息(ICMP Timer Exceeded)给发送源。

4.1.2 配置

Ping 内部接口的 IP 地址

DUT# ping 10.10.29.247	Ping 内部接口的 IPv4 地址 10.10.29.247
DUT# ping ipv6 2001:1000::1	Ping 内部接口的 IPv6 地址 2001:1000::1

Ping 管理口的 IP

DUT# ping mgmt-if 10.10.29.247	Ping 带外管理口的 IPv4 地址 10.10.29.247
DUT# ping mgmt-if ipv6 2001:1000::1	Ping 带外管理口的 IPv6 地址 2001:1000::1

Ping VRF 实例的 IP

DUT# ping vrf vrf1 10.10.10.1	Ping VRF 实例的 IP 10.10.10.1

Traceroute 内部接口 IP

DUT# traceroute 1.1.1.2	Traceroute 内部接口的 IP 1.1.1.2
Switch# traceroute ipv6 2001:1000::1	Traceroute 内部接口的 IP 2001:1000::1

4.1.3 命令验证

```
Switch # ping mgmt-if 192.168.100.101
PING 192.168.100.101 (192.168.100.101) 56(84) bytes of data.
64 bytes from 192.168.100.101: icmp_seq=0 ttl=64 time=0.092 ms
64 bytes from 192.168.100.101: icmp_seq=1 ttl=64 time=0.081 ms
64 bytes from 192.168.100.101: icmp_seq=2 ttl=64 time=0.693 ms
64 bytes from 192.168.100.101: icmp_seq=3 ttl=64 time=0.071 ms
64 bytes from 192.168.100.101: icmp_seq=4 ttl=64 time=1.10 ms
--- 192.168.100.101 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4054ms
rtt min/avg/max/mdev = 0.071/0.408/1.104/0.421 ms, pipe 2
Switch# traceroute 1.1.1.2
traceroute to 1.1.1.2 (1.1.1.2), 30 hops max, 38 byte packets
 1  1.1.1.2 (1.1.1.2)  112.465 ms  102.257 ms  131.948 ms
Switch # ping mgmt-if ipv6 2001:1000::1
PING 2001:1000::1(2001:1000::1) 56 data bytes
64 bytes from 2001:1000::1: icmp_seq=1 ttl=64 time=0.291 ms
64 bytes from 2001:1000::1: icmp_seq=2 ttl=64 time=0.262 ms
64 bytes from 2001:1000::1: icmp_seq=3 ttl=64 time=0.264 ms
64 bytes from 2001:1000::1: icmp_seq=4 ttl=64 time=0.270 ms
64 bytes from 2001:1000::1: icmp_seq=5 ttl=64 time=0.274 ms
--- 2001:1000::1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3997ms
rtt min/avg/max/mdev = 0.262/0.272/0.291/0.014 ms
Switch #
```

4.2 NTP 配置

4.2.1 简介

NTP 是一个具有冗余能力的分层时间分布系统。NTP 测量内网延迟和设备上运行它的算法的延误。使用这样的技术，NTP 可以使 LAN 内的设备时间同步，精度达到毫秒级，WAN 上的设备时间同步，精度达到百毫秒级。NTP 时间分布树的分层特性使用户

能通过一个等级（层级）选择需要的精度。一台时间服务器，放置在树的高端（低层级），提供了高精度的 UTC 标准时间。

主机可作为时间服务器，他们提供了他们认为是正确的时间到其他主机。主机也可作为客户端，向服务器请求时间同步。主机也可既充当客户端又当服务器，因为这些主机是在一个链路上，正确的时间从一个主机转发到另一个主机上。作为这个链路的一部分，首先一台主机作为一个客户端从另一台作为时间服务器的主机获取正确的时间。然后作为其他主机同步时间的时间服务器。

配置 NTP 客户端之前请确认 NTP 服务器已开启 NTP 服务。

4.2.2 配置

I. 配置接口 vlan10

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 配置模式
Switch(config-vlan)# vlan 10	添加 VLAN 10 到数据库
Switch(config-vlan)# exit	退出 VLAN 配置模式
Switch(config)# interface eth-0-26	进入接口配置模式
Switch(config-if)# switch access vlan 10	添加端口到 vlan 10
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface vlan10	进入 VLAN 配置模式
Switch(config-if)# ip address 6.6.6.5/24	设置 IP 地址
Switch(config-if)# exit	退出 VLAN 配置模式

II. 配置 NTP 客户端

Switch(config)# ntp key 1 serverkey	使能 trustedkey
Switch(config)# ntp server 6.6.6.6 key 1	配置 NTP 服务器的 IP 地址
Switch(config)# ntp authentication enable	使能 authentication
Switch(config)# ntp trustedkey 1	一旦使能 authentication，客户端交换机仅发送 time-of-day 请求到信任 NTP 服务器
Switch(config)# ntp ace 6.6.6.6 none	配置 ntp ace

III. 配置 NTP 服务器

步骤 1 显示接口 eth1 的 IP 地址。

[root@localhost octeon]# ifconfig eth1

```
eth1      Link encap:Ethernet  HWaddr 00:08:C7:89:4B:AA
          inet addr:6.6.6.6  Bcast:6.6.6.255  Mask:255.255.255.0
          inet6 addr: fe80::208:c7ff:fe89:4baa/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3453 errors:1 dropped:0 overruns:0 frame:1
          TX packets:3459 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:368070 (359.4 KiB)  TX bytes:318042 (310.5 KiB)
```

步骤 2 通过 ping 检查网络连接。

[root@localhost octeon]# ping 6.6.6.5

```
PING 6.6.6.5 (6.6.6.5) 56(84) bytes of data.
64 bytes from 6.6.6.5: icmp_seq=0 ttl=64 time=0.951 ms
64 bytes from 6.6.6.5: icmp_seq=1 ttl=64 time=0.811 ms
64 bytes from 6.6.6.5: icmp_seq=2 ttl=64 time=0.790 ms
```

步骤 3 配置 ntp.conf。

[root@localhost octeon]# vi /etc/ntp.conf

```
server  127.127.1.0 # local clock
fudge   127.127.1.0 stratum 5
#
# Drift file.  Put this in a directory which the daemon can write to.
# No symbolic links allowed, either, since the daemon updates the file
# by creating a temporary in the same directory and then rename()'ing
# it to the file.
#
driftfile /var/lib/ntp/drift
broadcastdelay  0.008
broadcast 6.6.6.255
#
# PLEASE DO NOT USE THE DEFAULT VALUES HERE. Pick your own, or remote
# systems might be able to reset your clock at will. Note also that
# ntpd is started with a -A flag, disabling authentication, that
# will have to be removed as well.
#
#disable auth
keys        /etc/ntp/keys
trustedkey 1
```

步骤 4 配置 keys。

[root@localhost octeon]# vi /etc/ntp/keys

```
#
```

```
# PLEASE DO NOT USE THE DEFAULT VALUES HERE. Pick your own, or remote
# systems might be able to reset your clock at will. Note also that
# ntpd is started with a -A flag, disabling authentication, that
# will have to be removed as well.
#
1 M serverkey
```

步骤 5 启动 ntpd 服务器。

[root@localhost octeon]# ntpd

4.2.3 命令验证

```
Switch# show ntp
Current NTP configuration:
============================================================
NTP access control list:
  6.6.6.6 none
Unicast peer:
Unicast server:
  6.6.6.6 key 1
Authentication: enabled
Local reference clock:
Switch# show ntp status
Current NTP status:
============================================================
clock is synchronized
stratum:         7
reference clock: 6.6.6.6
frequency:       17.365 ppm
precision:       2**20
reference time:  d14797dd.70b196a2  ( 1:54:37.440 UTC   Thu Apr  7 2011)
root delay:      0.787 ms
root dispersion: 23.993 ms
peer dispersion: 57.717 ms
clock offset:    -0.231 ms
stability:       6.222 ppm
Switch# show ntp associations
Current NTP associations:
   remote           refid      st   when poll reach   delay   offset  disp
==============================================================================
*6.6.6.6         127.127.1.0     6   50  128   37      0.778   -0.234  71.945
synchronized, + candidate, # selected, x falsetick, . excess, - outlier
```

注意说明

如果用户不想使用 authentication 选项，可以在 ntp.conf 文件上去使能 auth 以及在设备上去使能 ntp authentication。

Ntp 的服务器端 startum 号必须小于当前客户端的 startum 号。

4.3 Phy Loopback 管理

4.3.1 简介

Phy loopback 是一个私有的模块，实现物理层的环回功能。它包含两个级别的环回：一种是通过 phy 硬件实现环回（包括 internal 和 external 两种模式），另一种是 port 级别的环回，通过芯片实现。

Phy loopback 只能配置在物理口上：

- 如果配置为 external phy 模式，所有进入此端口的报文被环回回去。
- 如果配置为 internal phy 模式，所有期望从此端口出去的报文被环回到另外一个指定的端口。
- 如果配置为 port loopback 模式，所有进入此端口的报文被环回回去，此模式还可以指定是否进行源、目的 MAC 的交换，如果交换 MAC，芯片会重新计算 CRC 校验和。

4.3.2 配置 external phy 环回模式

I. 拓扑

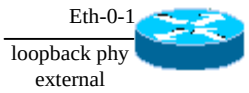


图4-1 external phy topo

II. 配置

Switch# configure terminal	进入全局配置模式
Switch (config)# interface eth-0-1	进入端口配置模式
Switch (config-if)# no shutdown	配置端口管理 up
Switch (config-if)# loopback phy external	配置端口为 external phy 环回模式
Switch (config-if)# end	退出到特权模式
Switch# show phy loopback	查看配置

4.3.3 配置 internal phy 环回模式

I. 配置

Switch # configure terminal	进入全局配置模式

Switch (config)# interface eth-0-2	进入端口配置模式
Switch (config-if)# no shutdown	配置端口管理 up
Switch (config-if)# exit	退出到全局配置模式
Switch (config)# interface eth-0-1	进入端口配置模式
Switch (config-if)# no shutdown	配置端口管理 up
Switch (config-if)# loopback phy internal eth-0-2	配置端口为 internal phy 环回模式， 并指定 interface 2 为目的端口
Switch (config-if)# end	退出到特权模式
Switch# show phy loopback	查看配置

4.3.4 配置 port level 环回模式

I. 配置

Switch # configure terminal	进入全局配置模式
Switch (config)# interface eth-0-1	进入端口配置模式
Switch (config-if)# no shutdown	配置端口管理 up
Switch (config-if)# loopback port mac-address swap	配置端口为 port level 环回模式，并且指定进行源、目的 MAC 交换
Switch (config-if)# end	退出到特权模式
Switch# show phy loopback	查看配置

4.3.5 命令验证

Switch# show phy loopback

```
Interface  Type       DestIntf   SwapMac
-------------------------------------------
eth-0-1    external   -          -
-------------------------------------------
```

4.3.6 L2 ping 配置

L2 ping 是一个用于检测交换机间的连通性的工具。Window、Linux 上的 IP ping 是通过 ICMP 协议实现，工作在 3 层网络上的，而 L2 ping 工作在二层网络。

当系统发出 L2 Ping 请求时，以 ether type 0x9009 为标志的协议报文将进入二层网络，当通过二层网络到达对端指定目的端口时，如果该端口上使能了 l2 ping response，对端系统就会回复 l2 ping 请求。

I. 配置

switch2

Switch2 # configure terminal	进入全局配置模式
Switch2 (config)# interface eth-0-2	进入端口配置模式
Switch2 (config-if)# no shutdown	配置端口管理 up
Switch2 (config-if)# l2 ping response enable	使能 l2 ping 回复功能
Switch2 (config-if)# end	退出到特权模式

switch1

Switch1# configure terminal	进入全局配置模式
Switch1 (config)# interface eth-0-1	进入端口配置模式
Switch1 (config-if)# no shutdown	配置端口管理 up
Switch1 (config-if)# end	退出到特权模式
Switch1# l2 ping 001e.0808.58f1 interface eth-0-1 count 10 interval 1000 timeout 2000	001e.0808.58f1 是对端端口 eth-0-2 的接口地址 用户可以指定 ping 的次数、间隔、以及超时时间

I. 命令验证

Switch1# l2 ping 001e.0808.58f1 interface eth-0-9 count 10 interval 1000 timeout 2000

```
Sending 10 L2 ping message(s):
64 bytes from 001e.0808.58f1: sequence = 0, time = 10ms
64 bytes from 001e.0808.58f1: sequence = 1, time = 15ms
64 bytes from 001e.0808.58f1: sequence = 2, time = 13ms
64 bytes from 001e.0808.58f1: sequence = 3, time = 12ms
64 bytes from 001e.0808.58f1: sequence = 4, time = 20ms
64 bytes from 001e.0808.58f1: sequence = 5, time = 21ms
64 bytes from 001e.0808.58f1: sequence = 6, time = 12ms
64 bytes from 001e.0808.58f1: sequence = 7, time = 16ms
64 bytes from 001e.0808.58f1: sequence = 8, time = 14ms
64 bytes from 001e.0808.58f1: sequence = 9, time = 17ms
L2 ping completed.
```

```
---------------------------------
10 packet(s) transmitted, 10 received, 0 % packet loss
```

4.4 RMON 管理

4.4.1 简介

RMON 是一个 Internet 工程任务组（IETF）标准的监测规范，允许不同的网络代理和控制台系统交换网的监测数据。用户可以结合 RMON 和交换机中的简单网络管理协议（SNMP）代理来监控网络中流经交换机的数据流量。 RMON 是一种标准的监测规范，它定义了一套统计与 RMON 兼容的控制台系统或网络探头一起提供全面的网络故障诊断，规划和性能优化的信息。

4.4.2 配置

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入端口模式
Switch(config-if)# rmon collection stats 1 owner test	创建一条统计。统计 ID 为 1，用于记录端口的统计
Switch(config-if)# rmon collection history 1 buckets 100 interval 1000 owner test	在端口创建一条统计历史记录编号为 1，每 1000 秒记录一次端口统计值，一共保留 100 次
Switch(config-if)# exit	退出端口模式
Switch(config)# rmon event 1 log trap public description test_event owner test	创建一条事件，事件 ID 为 1。如果触发该事件系统会发送 log 和 trap
Switch(config)# rmon alarm 1 etherStatsEntry.6.1 interval 1000 delta rising-threshold 1000 event 1 falling-threshold 1 event 1 owner test	创建一条警告，关注 ETHERSTATSBROADCASTPKTS 这个值，每 1000 秒统计一次，如果超过 1000 或者低于 1 都会触发事件 1

4.4.3 命令验证

Switch# show rmon statistics

```
  Rmon collection index 1
        Statistics ifindex = 1, Owner: test
        Input packets 0, octets 0, dropped 0
        Broadcast packets 0, multicast packets 0, CRC alignment errors 0,
collisions 0
        Undersized packets 0, oversized packets 0, fragments 0, jabbers 0
```

```
        # of packets received of length (in octets):
        64: 0,  65-127: 0, 128-255: 0
        256-511: 0,  512-1023: 0, 1024-max: 0
```

Switch# show rmon history

```
History index = 1
        Data source ifindex = 1
        Buckets requested = 100
        Buckets granted = 100
        Interval = 1000
        Owner: test
```

Switch# show rmon event

```
Event Index = 1
        Description: test_event
        Event type Log & Trap
        Event community name: public
        Last Time Sent = 00:00:00
        Owner: test
```

Switch# show rmon alarm

```
Alarm Index = 1
Alarm status = VALID
Alarm Interval = 1000
Alarm Type is Delta
Alarm Value = 00
Alarm Rising Threshold = 1000
Alarm Rising Event = 1
Alarm Falling Threshold = 1
Alarm Falling Event = 1
Alarm Owner is test
```

4.5 SNMP 网络管理

4.5.1 简介

SNMP 是管理进程（NMS）和代理进程（Agent）之间的通信协议。它规定了在网络环境中对设备进行监视和管理的标准化管理框架、通信的公共语言、相应的安全和访问控制机制。网络管理员使用 SNMP 功能可以查询设备信息、修改设备的参数值、监控设备状态、自动发现网络故障、生成报告等。

SNMP 具有以下技术优点：

- 基于 TCP/IP 互联网的标准协议，传输层协议一般采用 UDP。
- 自动化网络管理。网络管理员可以利用 SNMP 平台在网络上的节点检索信息、修改信息、发现故障、完成故障诊断、进行容量规划和生成报告。

- 屏蔽不同设备的物理差异，实现对不同厂商产品的自动化管理。SNMP 只提供最基本的功能集，使得管理任务与被管设备的物理特性和实际网络类型相对独立，从而实现对不同厂商设备的管理。
- 简单的请求—应答方式和主动通告方式相结合，并有超时和重传机制。
- 报文种类少，报文格式简单，方便解析，易于实现。
- SNMPv3 版本提供了认证和加密安全机制，以及基于用户和视图的访问控制功能，增强了安全性。

4.5.2 参考

SNMP 基于以下 RFC：

SNMPv1：在 RFC1157 中定义

SNMPv2C：在 RFC1901 中定义

SNMPv3：在 RFC2273 至 2275 中定义

4.5.3 术语

以下简单描述了 SNMP 协议的条目和概念。

Agent

Agent 是网络设备中的一个应用模块，用于维护被管理设备的信息数据并响应 NMS 的请求，把管理数据汇报给发送请求的 NMS。Agent 接收到 NMS 的请求信息后，完成查询或修改操作，并把操作结果发送给 NMS，完成响应。同时，当设备发生故障或者其他事件的时候，Agent 会主动发送 Trap 信息给 NMS，通知设备当前的状态变化。

Management Information Base (MIB)

任何一个被管理的资源都表示成一个对象，称为被管理的对象。MIB 是被管理对象的集合。它定义了被管理对象的一系列属性：对象的名称、对象的访问权限和对象的数据类型等。每个 Agent 都有自己的 MIB。MIB 也可以看作是 NMS 和 Agent 之间的一个接口，通过这个接口，NMS 可以对 Agent 中的每一个被管理对象进行读/写操作，从而达到管理和监控设备的目的。

Engine ID

一个网络节点的唯一 ID。

Trap

Trap 是 Agent 主动向 NMS 发送的信息，用于报告一些紧急的重要事件（如被管理设备重新启动等）。Trap 报文有两种：通用 Trap 和企业自定义 Trap。设备支持的通用 Trap 包括 authentication、coldstart、linkdown、linkup 和 warmstart 五种，其它均为企业自定义 Trap。企业自定义 Trap 由模块生成。因为 Trap 信息通常较多，会占用设备内存，从

而影响设备性能，所以建议用户根据需要开启指定模块的 Trap 功能，生成相应的 Trap 报文。

4.5.4 拓扑

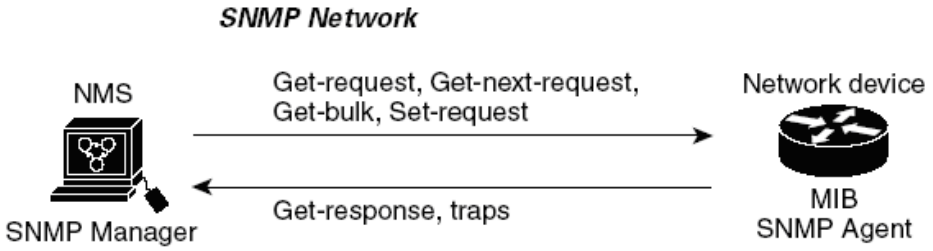


图4-2 SNMP 网络

4.5.5 启用 SNMP

I. 配置

在特权 EXEC 模式，启用 SNMP 服务

Switch# configure terminal	进入配置模式
Switch(config)# snmp-server enable	启用 SNMP
Switch(config)# end	退出配置模式
Switch# show running-config	显示配置

I. 命令验证

Switch# show running-config

```
snmp-server enable
```

4.5.6 团体字符串配置

您可以使用 SNMP 团体字符串来定义的 SNMP 管理者和代理之间的关系。团体字符串的行为就像一个密码，以允许访问代理交换机上。您可以指定一个或多个团体字符。

- 一个 MIB 视图，它定义了所有给定团体可访问的 MIB 对象子集。
- 设置访问的 MIB 对象的读、写权限。

在特权 EXEC 模式，开始按照下列步骤来配置交换机上的一个团体字符串，以下步骤配置完成后，就可以实现 SNMP 的基本读写功能。

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# snmp-server view DUT included 1	配置一个视图名字 “DUT”（可选）
Switch(config)# snmp-server community public read-write (view DUT)	配置团体名字 “public” 读写权限，可访问的视图为 “DUT”.括号内为可选字段
Switch(config)# end	退出配置模式

II. 命令验证

Switch# show running-config

```
snmp-server enable
snmp-server view DUT included .1
snmp-server community public read-only view DUT
```

4.5.7 SNMPv3 Groups, Users and Accesses 配置

你可以为 SNMP 服务器指定一个 (engine ID)，创建一个 SNMP 组，在 SNMP 组中加入用户、设置权限。

在特权 EXEC 模式，开始按照下列步骤操作，在交换机上配置 SNMP。

I. 配置

Switch# configure terminal	进入配置模式
Switch (config)# snmp-server engineID 8000123456	配置 engineID
Switch(config)# snmp-server usm-user usr1 authentication md5 mypassword privacy des yourpassword	配置用户名和密码以及验证类型
Switch(config)# snmp-server group grp1 user usr1 security-model usm	创建 SNMP 组
Switch(config)# snmp-server access grp1 security-model usm noauth	设置组内成员的权限
Switch(config)# end	退出全局模式

II. 命令验证

Switch# show running-config

```
snmp-server engineID 8000123456
snmp-server usm-user usr1 authentication md5 mypassword privacy des yourpassword
snmp-server group grp1 user usr1 security-model usm
snmp-server access grp1 security-model usm noauth
```

4.5.8 SNMPv1 和 SNMPv2 的 notifications 配置

在特权 EXEC 模式，开始在交换机上配置的 SNMP 按照下列步骤操作。

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# snmp-server trap enable all	开启所有 Trap
Switch(config)# snmp-server trap target-address 10.0.0.2 community public	配置目的 IPv4 地址以及团体名 Public
Switch(config)# snmp-server trap target-address 2001:1000::1 community public	配置目的 IPv6 地址以及团体名 Public
Switch(config)# end	退出配置模式

II. 命令验证

Switch# show running-config

```
snmp-server trap target-address 10.0.0.2 community public
snmp-server trap target-address 2001:1000::1 community public
snmp-server trap enable vrrp
snmp-server trap enable igmp snooping
snmp-server trap enable ospf
snmp-server trap enable pim
snmp-server trap enable stp
snmp-server trap enable system
snmp-server trap enable coldstart
snmp-server trap enable warmstart
snmp-server trap enable linkdown
snmp-server trap enable linkup
```

4.5.9 SNMPv3 的 notifications 配置

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# snmp-server trap enable all	开启所有 Trap
Switch(config)# snmp-server notify notif1 tag tmptag trap	创建一个 Trap 消息条目

Switch(config)# snmp-server target-address targ1 param parm1 10.0.0.2 taglist tmptag	配置目的 IPv4 地址以及团体名 Public
Switch(config)# snmp-server target-address t1 param p1 2001:1000::1 taglist tag1	配置目的 IPv6 地址以及团体名 Public
Switch(config)# snmp-server target-params parm1 user usr1 security-model v3 message-processing v3 noauth	加入一个用户到 SNMP 组内
Switch(config)# end	退出配置模式
Switch# show running-config	检查配置

II. 命令验证

Switch# show running-config

```
snmp-server notify notif1 tag tmptag trap
snmp-server target-address t1 param p1  2001:1000::1 taglist tag1
snmp-server target-address targ1 param parm1  10.0.0.2 taglist tmptag
snmp-server target-params parm1 user usr1 security-model v3 message-processing v3
noauth
snmp-server trap enable vrrp
snmp-server trap enable igmp snooping
snmp-server trap enable ospf
snmp-server trap enable pim
snmp-server trap enable stp
snmp-server trap enable system
snmp-server trap enable coldstart
snmp-server trap enable warmstart
snmp-server trap enable linkdown
snmp-server trap enable linkup
```

4.6 Sflow 配置

4.6.1 简介

Sflow 即 Sampled Flow，是一种监视进入设备流量的技术。它在监视设备上应用，通过一种采样机制以一定速率采样，然后将采样信息送到监视 server。在 server 端可以参看多个 agent 的流量情况。

Sflow 有两种类型的采样信息：一种是端口的统计信息，一种是被采样报文的头部。

4.6.2 术语

Sflow：Sampled flow

4.6.3 拓扑图

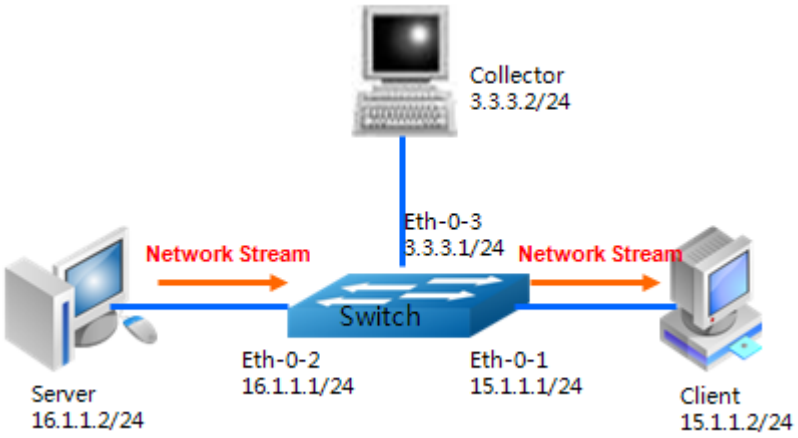


图4-3 Sflow 拓扑

4.6.4 配置

默认配置

Feature	Default Setting
Global sflow	disabled
sflow on port	disable
Collector udp port	6343
counter interval time	20 seconds

Sflow 配置

本节包含基本的配置 Sflow 的例子。所有进入端口 eth-0-1 的报文将会以一定速率采样，然后发送给 collector PC 3.3.3.2。

Switch# configure terminal	进入配置模式
Switch(config)# sflow enable	全局使能 Sflow
Switch(config)# sflow counter interval 20	配置基于统计的采样间隔
Switch(config)# sflow agent ip 3.3.3.1	配置代理地址
Switch(config)# sflow collector 3.3.3.2 6342	配置 collector 地址
Switch(config)# sflow collector 2001:1000::1	配置 collector 地址

Switch(config)# interface eth-0-1	进入端口模式
Switch(config-if)# sflow flow-sampling rate 8192	配置基于报文的采样速率
Switch(config-if)# sflow flow-sampling enable input	端口上使能基于报文的采样功能
Switch(config-if)# sflow counter-sampling enable	端口上使能基于统计的采样功能
Switch(config-if)# no switchport	将端口切换到 3 层口
Switch(config-if)# ip address 15.1.1.1/24	配置端口的 IP 地址
Switch(config-if)# exit	退出到 config 模式
Switch(config)# interface eth-0-2	进入端口模式
Switch(config-if)#no switchport	切换到 3 层口
Switch(config-if)# ip address 16.1.1.1/24	配置端口的 IP 地址
Switch(config-if)# exit	退出到 config 模式
Switch(config)# interface eth-0-3	进入端口模式
Switch(config-if)# no switchport	切换到 3 层口
Switch(config-if)# ip address 3.1.1.1/24	配置端口的 IP 地址

4.6.5 命令验证

用如下命令查看 sflow 配置：

Switch# show sflow

```
sFlow Global Information:
 Agent IP address              : 2.2.2.1
 Agent IPv6 address            : 2026::2
 Counter Sampling Interval     : 20 seconds
 Collector 1:
  Address: 3.3.3.2
  Port: 6342
 Collector 2:
  Address: 2001:1000::1
  Port: 6343
sFlow Port Information:
                                    Flow-Sample   Flow-Sample
 Port        Counter     Flow       Direction     Rate
 ------------------------------------------------------
 eth-0-1     Enable      Enable     Input         8192
```

4.7 LLDP 配置

4.7.1 简介

链路层发现协议 LLDP（Link Layer Discovery Protocol）是 IEEE 802.1ab 中定义的第二层发现协议。第二层发现（Layer 2 Discovery）可以准确定位了设备附带有哪些接口，以及设备之间相互连接等二层信息，例如端口的 VLAN 属性和支持的协议类型等，并显示出了客户端、交换机、路由器和应用服务器以及网络服务器之间的路径。这些详细的信息对快速获取相连设备的拓扑状态、设备间的配置冲突、查询网络失败的根源将很有帮助。

4.7.2 术语

LLDP：Link Layer Discovery Protocol

4.7.3 配置

基本配置

Switch# configure terminal	进入配置模式
Switch(config)# lldp enable	全局使能 LLDP
Switch(config)# interface eth-0-9	进入端口模式
Switch(config)# no shutdown	打开端口
Switch(config-if)# no lldp tlv 8021-org-specific vlan-name	取消选择 IEEE 802.1 tlv 集中 Vlan Name TLV
Switch(config-if)# lldp tlv med location-id ecs-elin 1234567890	选择并配置 MED tlv 集中 Location ID TLV
Switch(config-if)# lldp enable txrx	端口使能 LLDP，并配置模式为 TXRX

状态配置

Switch# configure terminal	进入配置模式
Switch(config)# lldp timer msg-tx-interval 40	配置 LLDP 报文传输间隔为 40 秒
Switch(config)# lldp timer tx-delay 3	配置 LLDP 报文传输延迟为 3 秒
Switch(config)# lldp timer reinitDelay 1	配置 LLDP 重新使能延迟为 1 秒

4.7.4 命令验证

用如下命令查看 LLDP 配置：

```
Switch# show lldp local config
LLDP global configuration:
============================================================
LLDP function global enabled : YES
LLDP msgTxHold     : 4
LLDP msgTxInterval : 40
LLDP reinitDelay   : 1
LLDP txDelay       : 3
Switch# show lldp local config interface eth-0-9
LLDP configuration on interface eth-0-9 :
============================================================
LLDP admin status : TXRX
Basic optional TLV Enabled:
    Port Description TLV
    System Name TLV
    System Description TLV
    System Capabilities TLV
    Management Address TLV
IEEE 802.1 TLV Enabled:
    Port Vlan ID TLV
    Port and Protocol Vlan ID TLV
    Protocol Identity TLV
IEEE 802.3 TLV Enabled:
    MAC/PHY Configuration/Status TLV
    Power Via MDI TLV
    Link Aggregation TLV
    Maximum Frame Size TLV
LLDP-MED TLV Enabled:
    Med Capabilities TLV
    Network Policy TLV
    Location Identification TLV
    Extended Power-via-MDI TLV
    Inventory TLV
Switch# show running-config
!
lldp enable
lldp timer msg-tx-interval 40
lldp timer reinit-delay 1
lldp timer tx-delay 3
。。。
interface eth-0-9
lldp enable txrx
 no lldp tlv 8021-org-specific vlan-name
 lldp tlv med location-id ecs-elin 1234567890
!
Switch# show lldp neighbor
Remote LLDP Information
============================================================
```

```
Chassis ID type: Mac address
Chassis ID     : 48:16:be:a4:d7:09
Port ID type   : Interface Name
Port ID        : eth-0-9
TTL : 160
Expired time: 134
…
Location Identification :
ECS ELIN: 123456789
```

5 组播配置指导

5.1 IP Multicast-Routing 配置

5.1.1 简介

随着 Internet 网络的不断发展，网络数据、语音、视频信息等多种交互业务与日俱增。另外，新兴的电子商务、网上会议、网上拍卖、视频点播、远程教学等对带宽和实时数据交互要求较高的服务逐渐兴起，这些服务对信息安全性、可计费性、网络带宽提出了更高的要求。

当网络中需要某信息的用户量不确定时，单播和广播方式的效率会很低，IP 组播技术的出现改变了这一现状。当网络中的某些用户需要特定信息时，组播信息发送者（即组播源）仅发送一次信息，借助组播路由协议为组播数据包建立树型路由，被传递的信息在距离用户端尽可能近的节点才开始复制和分发。

通过组播路由协议，多个接收者能跨越不同网络接收到组播数据。

- IGMP(Internet Group Management Protocol，因特网组管理协议)是 TCP/IP 协议族中负责 IP 组播成员管理的协议。它用来在 IP 主机和与其直接相邻的组播路由器之间建立、维护组播组成员关系。
- PIM（Protocol Independent Multicast，协议无关组播），用于组播路由器或多层交换机之间。为 IP 组播提供路由的单播路由协议可以是静态路由、RIP、OSPF、IS-IS、BGP 等，组播路由和单播路由协议无关，只要单播路由协议能产生路由表项即可。借助 RPF（Reverse Path Forwarding，逆向路径转发）机制，PIM 实现了在网络中传递组播信息。为了描述上的方便，我们把由支持 PIM 协议的组播路由器所组成的网络称为 PIM 组播域，PIM 有两种模式：密集模式和稀疏模式，我们目前只支持稀疏模式。

5.1.2 配置

我们默认能支持限制 2048 条组播路由表。

Switch# configure terminal	进入配置模式
Switch(config)# ip multicast route-limit 1000	配置最大组播限制条目

5.1.3 检查配置

Switch# show ip mroute 192.168.47.2

```
IP Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder installed
Timers: Uptime/Stat Expiry
Interface State: Interface (TTL)
(192.168.47.2, 238.255.0.1), uptime 00:00:03, stat expires 00:03:27
Owner PIM-SM, Flags: TF
  Incoming interface: eth-0-3
  Outgoing interface list:
    Register (1)
    eth-0-1 (1)
(192.168.47.2, 238.255.0.2), uptime 00:00:02, stat expires 00:03:28
Owner PIM-SM, Flags: TF
  Incoming interface: eth-0-3
  Outgoing interface list:
    Register (1)
    eth-0-2 (1)
```

5.2 IGMP 配置

5.2.1 简介

参与 IP 组播的主机、路由器、多层交换机必须具备 IGMP 功能。该协议定义了查询器和主机角色：

- 网络设备的查询器发送查询消息给网络中特定组来发现组播中的成员。
- 主机发送 IGMP 报告报文(响应查询报文)来通知查询者主机要加入相应的组播组列表中。
- 一个组播组的成员是动态的，主机可以随时加入和离开。在一个多播组成员的位置或数量上没有限制。

一个主机可作为不止一个组播组的成员，在同一时间，成员在组播组内活跃，它可以改变从组到组、时间到时间。一个组播组，可以持续很长一段时间，也可以非常短暂。

IGMP 报文使用下面的组播地址：

- IGMP 普通组查询以 224.0.0.1 为目的地址(在一个子网中的所有系统)。
- IGMP 特定组的查询以特定组 IP 地址为目的查询。
- IGMP 组成员发送 Report 报文给特定的组播 IP 地址。
- IGMP 版本 2(IGMPv2)离开组播组时，发送离开消息给 224.0.0.2。

5.2.2 参考

IGMP 模块是基于以下 RFC

- RFC 1112
- RFC 2236
- RFC 3376

5.2.3 配置

IGMP 的使能是依赖于组播路由协议的使能，当接口上使能 PIM 或者其他组播路由协议，IGMP 将会在接口上自动启用，反之亦然。但是请注意，IGMP 在工作之前，IP 组播路由必须在全局模式启用。系统支持动态学习 IGMP 组记录，也可以配置静态 IGMP 组记录。

启用 IGMP

Switch# configure terminal	进入配置模式
Switch(config)# ip multicast-routing	全局模式下启用组播路由
Switch(config)# interface eth-0-1	进入接口 Eth-0-1
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.10.10/24	设置 IP 地址
Switch(config-if)# ip pim sparse-mode	接口上启用 PIM-SM

配置 IGMP 接口参数

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	接入接口模式
Switch(config-if)# ip igmp version 2	设置 IGMP 版本
Switch(config-if)# ip igmp query-interval 120	设置 IGMP 查询时间间隔
Switch(config-if)# ip igmp query-max-response-time 12	设置 IGMP 查询最大响应时间
Switch(config-if)# ip igmp robustness-variable 3	设置 IGMP 的鲁棒参数
Switch(config-if)# ip igmp last-member-query-count 3	设置 IGMP 的最后一个成员查询计数
Switch(config-if)# ip igmp last-member-	设置 IGMP 的最后一个成员查询间隔

query-interval 2000	

配置最大 IGMP 组数目

可以全局配置最大 IGMP 组数目或者接口模式下最大 IGMP 组数目。

Switch# configure terminal	进入配置模式
Switch(config)# ip igmp limit 2000	设置全局最大 IGMP 组数目
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip igmp limit 1000	设置接口下最大 IGMP 组数目

配置静态 IGMP 组

可以在接口模式下配置静态 IGMP 组。

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip igmp static-group 228.1.1.1	配置静态 IGMP 组

配置 IGMP 代理

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip pim sparse-mode	在接口上启用 PIM-SM
Switch(config-if)# ip igmp proxy-service	设置接口为 IGMP 代理上游口
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip pim sparse-mode	接口上启用 PIM-SM
Switch(config-if)# ip igmp mroute-proxy eth-0-1	设置 eth-0-2 为 IGMP 代理下游口，IGMP 代理上游口为 eth-0-1

5.2.4 检查配置

显示 IGMP 接口信息

Switch# show ip igmp interface

```
Interface eth-0-1 (Index 1)
 IGMP Inactive, Version 2 (default) proxy-service
 IGMP host version 2
 IGMP global limit is 2000
 IGMP global limit states count is currently 0
 IGMP interface limit is 1000
 IGMP interface has 0 group-record states
 IGMP activity: 0 joins, 0 leaves
 IGMP query interval is 120 seconds
 IGMP querier timeout is 366 seconds
 IGMP max query response time is 12 seconds
 Last member query response interval is 2000 milliseconds
 Group Membership interval is 372 seconds
 Last memeber query count is 3
 Robustness Variable is 3
Interface eth-0-2 (Index 2)
 IGMP Inactive, Version 2 (default)
 IGMP mroute-proxy interface is eth-0-1
 IGMP global limit is 2000
 IGMP global limit states count is currently 0
 IGMP interface limit is 16384
 IGMP interface has 0 group-record states
 IGMP activity: 0 joins, 0 leaves
 IGMP query interval is 125 seconds
 IGMP querier timeout is 255 seconds
 IGMP max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
 Last memeber query count is 2
 Robustness Variable is 2
```

显示 IGMP 组信息

Switch# show ip igmp groups

```
IGMP Connected Group Membership
Group Address    Interface        Uptime   Expires Last Reporter
228.1.1.1        eth-0-1           00:00:05 stopped  -
```

5.3 PIM-SM 配置

5.3.1 简介

协议无关组播稀疏模式(PIM-SM)是一个组播路由协议，用来将稀疏分散的组播设备联系起来协同工作。它将有助于分散的网络节点节约带宽和通过发送单一流量到多个接受着来降低网络流量。

PIM-SM 使用接收者发起成员的 IP 组播模型，支持共享和最短路径树，并使用软状态机制，以适应不断变化的网络条件。它依赖于单播路由协议来建立和维护路由器间的组播路由。

5.3.2 参考

在 PIM-SM 模块是基于以下的 IETF 标准：

RFC 4601

5.3.3 术语

以下是 PIM-SM 协议概念的简要描述：

- **汇聚点(RP)**：RP（Rendezvous Point）在 SM 模式中作为组播的汇聚点，发送者和接收者在 RP 处进行汇聚。对于所有的组播路由器，必须知道某个组播组对应哪个 RP。

  所有的组播数据需要在 RP 上注册，然后所有需要组播数据的接收者通过向 RP 发送 JOIN 报文来请求数据。源的注册机制就是让 RP 知道现在网络内有什么源的数据。

- **组播路由信息库(MRIB)**：组播路由表是从单播路由表的获得的。在 PIM-SM 中，MRIB 是用来决定向何处发送加入/剪枝消息。它还提供了目的网络的路由度量。发送和处理的 Assert 消息时将使用这些度量。

- **反向路径转发（RPF)**：反向路径转发是指路由器在接受数据包从源 A 通过接口 IF1 时，只有 IF1 是到达源 A 的出接口时才会接受这个包。反向路径转发通过使用单播路由表来决定入端口是否正确。这个数据包将被转发是由于单播路由表表明了接口 IF1 是到达源 A 的最短路径。单播路由表为组播数据选择最短路径。

- **组播树状态信息库(TIB)**：组播树状态信息库是组播路由器上保存所有组播转发树信息的一个信息库，通过收到 PIM 加入/剪枝消息，Assert 消息和 IGMP 消息建立起来。

- **上游 Upstream**：朝向树根，树根可能是源或 RP。

- **下游 Downstream**：远离树根，树根可能是源或 RP。

- **基于源的树**：基于源的树的转发路径是到达源的最短转发路径，如果单播路由度量是跳数，基于源的树的转发路径的跳数最小，如果单播路由度量是延迟，基于源的树的转发路径的延迟最小。

  对于每个组播源，有一个对应的组播转发树直接将源和接收者连接起来。所有发往指定组的流量沿着对应的转发树进行转发。

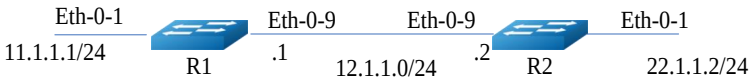
- **共享树：**共享树依赖于汇聚点(RP)，所有流量从源都发文那个汇聚点，然后汇聚点再将流量发送给接收者。对于每一个组播组来说，不管有多少个源，只有一个转发树。共享树是单向的，流量只会从 RP 流向接收者。如果一个源要发送组播数据，首先组播数据要被发到 RP，然后在从 RP 发送到接收者。
- **自举路由器(BSR)：**当一个组播源开始发送组播数据或者一个接收者开始发送加入信息到 RP，组播路由器必须知道汇聚点的信息。自举路由器负责在 PIM-SM 网络启动后，收集网络内的 RP 信息，为每个组选举出 RP，然后将 RP 集（即组-RP 映射数据库）发布到整个 PIM-SM 网络。
- **数据流从源到接收者：**发送 Hello 消息：PIM 路由器定期的发送 Hello 消息来发现 PIM 路由器邻居。Hello 消息是组播报文，使用 224.0.0.13 这个地址。PIM 路由器对 Hello 消息进行响应，Hello 消息中的 Hold 时间来决定信息的有效时间。
- **选举指定路由器：**在一个多路访问的网络中如果有多个组播路由器，只能有一个组播路由器被选为指定路由器，负责为本地网络的组播接收者往 RP 发送加入/剪枝消息。
- **RP 发现：**PIM-SM 通过自举路由器来产生自举消息，然后发布 RP 信息给所有的组播路由器。组播路由器接收和保存自举消息，当 DR 从直连 host 收到一个 IGMP 报文或组播数据，DR 计算出该组播组的 RP，然后发送加入/剪枝到 RP 或者封装 reigster 报文到 RP。在小网络环境下可以静态指定 RP。
- **加入共享树：**要加入一个多播组，主机发送一个 IGMP 消息给上游路由器，组播路由器向 RP 方向的上游的 PIM 邻居发送加入报文。当组播路由器接收到下游设备的加入请求后，检查本地的组播组是否存在。如果存在，说明加入消息被送到共享树，收到消息的接口被成为 outgoing 的接口.如果不存在，条目将被创建，收到消息接口的被加入到 outgoing 中并再次向 RP 方向的上游的 PIM 邻居发送加入报文。
- **组播源注册：**与组播源 S 直接相连的路由器接收到该组播报文后，就将该报文封装成 Register 注册报文，并单播发送给对应的 RP。当 RP 接收到来自组播源 S 的注册消息后，一方面解封装注册消息并将组播信息沿着 RPT 树转发到接收者，另一方面朝组播源 S 逐跳发送（S，G）加入消息，从而让 RP 和组播源 S 之间的所有路由器上都生成了（S，G）表项，这些沿途经过的路由器就形成了 SPT 树的一个分支。SPT 源树以组播源 S 为根，以 RP 为目的地组播源 S 发出的组播信息沿着已经建立好的 SPT 树到达 RP，然后由 RP 将信息沿着 RPT 共享树进行转发。
- **发送注册停止消息：**当 RP 从组播源接收到注册报文后也收到未封装的组播报文，将发送注册停止消息给组播源一侧的 DR，当 DR 收到注册停止消息后将不再发送注册消息给 RP 了。
- **剪枝端口：**接收者侧的组播路由器向 RP 方向的上游的 PIM 邻居发送剪枝报文，当上联组播路由器收到剪枝报文后，将收到剪枝报文的端口从转发端口中删除，当本路由器上没有其他接收者后会继续向 RP 方向的上游的 PIM 邻居发送剪枝报文。
- **转发组播数据：**PIM-SM 路由器将组播数据发往那些已经明确表示加入组播组的接收者。组播路由器将进行 RPF 检查，只有检查通过的组播数据包才将通过出端口发送出去。

5.3.4 配置通用 PIM Sparse-mode

I. 配置

PIM-SM 是一个软状态协议。主要要求是，所需的接口上启用 PIM-SM 协议，并正确配置的 RP 信息，通过静态或动态的方法。所有组播组的 IGMP 报告/离开和 PIM 加入/剪枝消息保持动态。目前，我们只支持一个 RP 的所有组播组（224.0.0.0 / 4）。

本节提供了两个相关的场景，PIM-SM 配置的例子。下面的例子中使用的网络拓扑如下；



配置静态 RP

以上例子中 R1 是 RP，所有的路由器都配置静态 RP：

- 每个路由器配置静态 RP 地址 *11.1.1.1*。
- 所有接口上必须启用 PIM-SM 功能。

R1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 11.1.1.1/24	配置 IP 地址
Switch(config-if)# ip pim sparse-mode	在接口上启用 PIM-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 12.1.1.1/24	配置 IP 地址
Switch(config-if)# ip pim sparse-mode	在接口上启用 PIM-SM
Switch(config-if)# exit	退出接口模式

Switch(config)# ip route 22.1.1.0/24 12.1.1.2	配置静态单播路由
Switch(config)# ip pim rp-address 11.1.1.1	配置静态 RP 地址

R2

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 22.1.1.2/24	配置 IP 地址
Switch(config-if)# ip pim sparse-mode	在接口上启用 PIM-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 12.1.1.2/24	配置 IP 地址
Switch(config-if)# ip pim sparse-mode	在接口上启用 PIM-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# ip route 11.1.1.0/24 12.1.1.1	配置静态单播路由
Switch(config)# ip pim rp-address 11.1.1.1	配置静态 RP 地址

I. 检查配置

所有的路由器配置使用相同的 RP 地址 11.1.1.1，使用以下命令来验证 RP 的配置，接口的详细信息和组播路由表。

RP 详细说明

在 R1 上，显示 PIM 稀疏模式 RP 映射的命令表明 11.1.1.1 是对所有组播组 224.0.0.0 / 4 静态配置的 RP。所有其他路由器都会有类似的输出：

R1# show ip pim sparse-mode rp mapping

```
PIM group-to-RP mappings
Group(s): 224.0.0.0/4, Static
```

```
    RP: 11.1.1.1
         Uptime: 00:08:21
```

接口的详细信息

显示 R1 接口的组播信息。

R1# show ip pim sparse-mode interface

```
Address          Interface VIFindex Ver/   Nbr    DR     DR          HoldTime
                                    Mode   Count  Prior
11.1.1.1         eth-0-1   2        v2/S   0      1      11.1.1.1        105
12.1.1.1         eth-0-9   0        v2/S   1      1      12.1.1.2        105
```

IP 组播路由表

显示 PIM-SM 的组播路由表。

R1# show ip pim sparse-mode mroute detail

```
IP Multicast Routing Table
(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
FCR Entries: 0
(*, 224.1.1.1) Uptime: 00:01:32
  RP: 11.1.1.1, RPF nbr: None, RPF idx: None
  Upstream:
   State: JOINED, SPT Switch: Enabled, JT: off
   Macro state: Join Desired,
  Downstream:
   eth-0-9:
     State: JOINED, ET Expiry: 179 secs, PPT: off
     Assert State: NO INFO, AT: off
      Winner: 0.0.0.0, Metric: 4294967295, Pref: 4294967295, RPT bit: on
     Macro state: Could Assert, Assert Track
  Join Olist:
   eth-0-9
```

R2# show ip pim sparse-mode mroute detail

```
IP Multicast Routing Table
(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
FCR Entries: 0
(*, 224.1.1.1) Uptime: 00:00:43
  RP: 11.1.1.1, RPF nbr: 12.1.1.1, RPF idx: eth-0-9
  Upstream:
   State: JOINED, SPT Switch: Enabled, JT Expiry: 18 secs
   Macro state: Join Desired,
  Downstream:
```

```
  eth-0-1:
    State: NO INFO, ET: off, PPT: off
    Assert State: NO INFO, AT: off
     Winner: 0.0.0.0, Metric: 4294967295, Pref: 4294967295, RPT bit: on
    Macro state: Could Assert, Assert Track
 Local Olist:
  eth-0-1
```

5.3.5 配置动态 RP

在小型并且简单的网络中，组播信息量少，全网络仅依靠一个 RP 进行信息转发即可，此时可以在 SM 域中各路由器上静态指定 RP 位置。但是更多的情况下，PIM-SM 网络规模都很大，通过 RP 转发的组播信息量巨大，为了缓解 RP 的负担同时优化共享树的拓扑结构，不同组播组应该对应不同的 RP，此时就需要自举机制来动态选举 RP。

I. 配置

以下是动态 RP 的详细配置：

R1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 11.1.1.1/24	配置 IP 地址
Switch(config-if)# ip pim sparse-mode	在接口上启用 PIM-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 12.1.1.1/24	配置 IP 地址
Switch(config-if)# ip pim sparse-mode	进入配置模式
Switch(config-if)# exit	进入接口模式
Switch(config)# ip route 22.1.1.0/24 12.1.1.2	配置静态单播路由
Switch(config)# ip pim rp-candidate eth-0-1	配置候选 RP 接口

R2

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 22.1.1.2/24	配置 IP 地址
Switch(config-if)# ip pim sparse-mode	在接口上启用 PIM-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 12.1.1.2/24	配置 IP 地址
Switch(config-if)# ip pim sparse-mode	在接口上启用 PIM-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# ip route 11.1.1.0/24 12.1.1.1	配置静态单播路由
Switch(config)# ip pim rp-candidate eth-0-9	配置候选 RP 接口
Switch(config)# ip pim bsr-candidate eth-0-9	配置候选 BSR 接口

选择最高优先级的路由器为 RP。如果有两个或多个路由器的优先级相同，在 BSR 机制的一个哈希函数是用来选择的 RP，以确保在 PIM 域的所有路由器对同一组相同的 RP。使用 **ip pim rp-candidate IFNAME PRIORITY** 命令来改变候选 RP 的默认的优先级。

I. 检查配置

PIM-SM 的组-RP 的 Mapping 关系

使用 **show ip pim sparse-mode rp mapping** 命令，来显示组-RP 的映射的详细信息，输出内容是候选 RP 信息。对组的范围 224.0.0.0 / 4 的组有两个候选 RP。候选 RP 11.1.1.1 默认的优先级 192，而候选 RP 12.1.1.2 的优先级被配置为 2。由于候选 RP 12.1.1.2 由于具有更高的优先权，它被选中作为组播组 224.0.0.0/24 的 RP。

R2# show ip pim sparse-mode rp mapping

```
PIM group-to-RP mappings
This system is the bootstrap router (v2)
```

```
Group(s): 224.0.0.0/4
  RP: 12.1.1.2
    Info source: 12.1.1.2, via bootstrap, priority 2
         Uptime: 01:55:20, expires: 00:02:17
  RP: 11.1.1.1
    Info source: 11.1.1.1, via bootstrap, priority 192
         Uptime: 01:55:23, expires: 00:02:13
```

RP 详细显示

要显示特定组的 RP 路由器的信息，使用下面的命令。此输出显示，12.1.1.2 已经选择 224.1.1.1 的组播组的 RP。

R2# show ip pim sparse-mode rp-hash 224.1.1.1

```
    RP: 12.1.1.2
    Info source: 12.1.1.2, via bootstrap
```

RP 信息后达到域中的所有 PIM 路由器，各种状态机保持所有路由从组成员的加入/剪枝的结果。要显示接口的详细信息和组播路由表的信息，请参见以上配置 RP 的静态部分。

5.3.6 配置自举路由器

每个组播组需要有一个为它服务的 RP，这个 RP 作为基于组播组的分发树的根。为了组播数据能从发送者到达接收者，在一个组播域内的组播路由器需要使用同样的组播组-RP 的映射。为了选择指定组播组的 RP，组播路由器需要维护一系列的组播组-RP 的映射关系，这被称为 RP 集。自举路由器的机制就是用来让在同一个组播域内的组播路由器能够学习到这个 RP 集。

BSR 是 PIM-SM 网络里的管理核心，主要负责：

- 负责收集网络中 Candidate-RP（C-RP）发来的 Advertisement 宣告信息。
- 为每个组播组选择部分 C-RP 信息以组成 RP-Set 集（即组播组和 RP 的映射数据库）。
- 发布到整个 PIM-SM 网络，从而使网络内的所有路由器（包括 DR）都会知道 RP 的位置。

在一个 PIM 域中，需要配置一个或多个候选 BSR，候选 BSR 之间通过自动选举，产生自举路由器 BSR，负责收集并发布 RP 信息。下面简单描述一下候选 BSR 之间的自动选举：

- 在将路由器配置为候选 BSR 时，必须同时指定一个启动了 PIM-SM 的接口。
- 每个候选 BSR 开始都认为自己是本 PIM-SM 的 BSR，并使用这个接口的 IP 地址作为 BSR 地址，发送自举报文（Bootstrap message）。
- 当候选 BSR 收到其它路由器发来的自举报文时，它将新收到的自举报文的 BSR 地址与自己的 BSR 地址进行比较，比较标准包括优先级和 IP 地址，优先级相同的情况下，较大的 IP 地址被认为是更好的。如果前者更好，则将这个新的 BSR 地址替

换自己的 BSR 地址，并且不再认为自己是 BSR。否则，保留自己的 BSR 地址，继续将自己视为 BSR。

- 备选 RP 将自己的 RP 信息报告给自举路由器，然后自举路由器将汇聚的 RP 集通过自举报文发布到整个组播域的所有路由器。

I. 拓扑

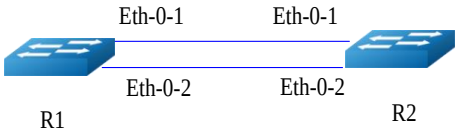


图5-1 BSR 拓扑

II. 配置

Router 1

Switch# configure terminal	进入配置模式
Switch(config)# ip pim bsr-candidate eth-0-1	指定 BSR 的候选接口，默认优先级 64

Router 2

Switch# configure terminal	进入配置模式
Switch(config)# ip pim bsr-candidate eth-0-1 10 25	配置 HASH 掩码长度为 10 优先级 25 的 BSR 候选接口
Switch(config)# ip pim rp-candidate eth-0-1 priority 0	配置优先级为 0 的 RP 候选接口

通过命令 **ip pim unicast-bsm** 配置接口以单播方式发送和接收 BSM 消息。

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip pim dr-priority 10	配置接口 DR 的优先级
Switch(config-if)# ip pim unicast-bsm	配置接口以单播方式发送和接收 BSM 消息

I. 检查配置

检查候选 BSR 路由器

Switch# show ip pim sparse-mode bsr-router

```
PIMv2 Bootstrap information
This system is the Bootstrap Router (BSR)
BSR address: 20.0.1.21
Uptime: 00:37:12, BSR Priority: 64, Hash mask length: 10
Next bootstrap message in 00:00:04
Role: Candidate BSR
State: Elected BSR
```

检查候选 BSR 路由器

Switch# show ip pim sparse-mode bsr-router

```
PIMv2 Bootstrap information
BSR address: 20.0.1.21
Uptime: 00:02:39, BSR Priority: 64, Hash mask length: 10
Expires: 00:00:03
Role: Candidate BSR
State: Pending BSR
Switch# show ip pim sparse-mode bsr-router
PIMv2 Bootstrap information
BSR address: 20.0.1.21
Uptime: 00:40:20, BSR Priority: 64, Hash mask length: 10
Expires: 00:02:07
Role: Candidate BSR
State: Candidate BSR
```

在 E-BSR 上检查 RP

Switch# sh ip pim sparse-mode rp mapping

```
PIM Group-to-RP Mappings
This system is the Bootstrap Router (v2)
Group(s): 224.0.0.0/4
RP: 20.0.1.11
Info source: 20.0.1.11, via bootstrap, priority 0
Uptime: 00:00:30, expires: 00:02:04
```

在 C-BSR 上检查 RP

Switch# show ip pim sparse-mode rp mapping

```
PIM Group-to-RP Mappings
Group(s): 224.0.0.0/4
RP: 20.0.1.11
Info source: 20.0.1.21, via bootstrap, priority 0
Uptime: 00:00:12, expires: 00:02:18
```

5.3.7 配置 PIM-SSM

PIM-SSM 是借助 PIM-SM 的部分技术和 IGMPv3 来实现的，其建立组播转发树的过程与 PIM-SM 创建 SPT 树的过程相似，即接收者 DR 在知道组播数据源的具体位置

后，直接向组播数据源发送 Join 消息，将组播数据流发送到接收者。

默认情况下，SSM 组播组地址的范围为 232.0.0.0~232.255.255.255。当用户加入的组播组属于 SSM 组地址范围内，通过 PIM-SSM 的进行处理；当用户加入的组播组不属于 SSM 组地址范围，通过 PIM-SM 的进行处理。

PIM-SSM 的特点是网络用户能够预先知道组播源的具体位置。因此用户在加入组播组时，可以明确指定从哪些源接收信息。组成员端 DR 了解到用户的需求后，直接向组播源的方向发送 Join 消息。Join 消息逐跳向上传输，在源与组成员之间建立 SPT。

PIM-SSM 只使用了 PIM-SM 的部分技术：无需维护 RP、无需构建 RPT、无需注册组播源，可以直接在源与组成员之间建立 SPT。

PIM-SSM 可以跟 PIM-SM 在组播路由器上一起工作。PIM-SSM 默认是 disable 的。

Switch# configure terminal	进入配置模式
Switch(config)# ip pim ssm default	使能 PIM-SSM
Switch(config)# ip pim ssm range ipacl	根据指定的 acl 来设置 PIM-SSM 的组范围

5.4 PIM-DM 配置

5.4.1 简介

协议无关组播密集模式(PIM-DM)是一个组播路由协议，用来将密集分布的组播设备联系起来协同工作。它将有助于分散的网络节点节约带宽和通过发送单一流量到多个接收者来降低网络流量。

PIM-DM 设想当一个组播源开始发送组播流的时候，所有的下游系统都期望接受这个组播流。刚开始组播流被泛洪到整个网络。当泛洪的时候，PIM-DM 使用 RPF 来防止组播流的环路。如果某些网络区域没有该组播组的接收成员，PIM-DM 会把转发分支通过剪枝来删除掉。

剪枝状态有一个生命周期，当生命周期超时后，组播数据将再一次开始转发，每个(S,G)对应的组播组都有自己的剪枝状态。当某个组播组有新的接收者出现在已经被剪枝的区域里，路由器会通过朝组播源发送”graft”消息来把剪枝状态转换成转发路径。

5.4.2 参考

在 PIM-DM 模块是基于以下的 IETF 标准：

RFC 3973

5.4.3 配置通用 PIM dense-mode

I. 拓扑

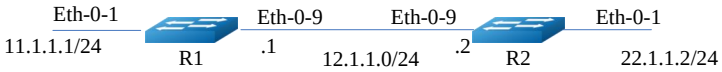


图5-2 配置 PIM dense-mode

II. 配置

PIM-DM 是一个软状态协议。主要要求是在所需的接口上启用 PIM-DM 协议。所有组播组的状态通过 IGMP 报告/离开和 PIM 消息来动态的维护。

本节提供了两个 PIM-DM 配置的相关的场景。下面的例子中使用的网络拓扑如上图；

组播流从 R1 的 eth-0-1 口进来，接收者来与 R2 的 eth-0-1 相连。

下面是配置的举例：

Configuring R1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入 eth-0-1 的接口模式
Switch(config-if)# no shutdown	启用端口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 11.1.1.1/24	配置接口的 ip 地址
Switch(config-if)# ip pim dense-mode	使能接口的 pim dm 功能
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入 eth-0-9 的接口模式
Switch(config-if)# no shutdown	启用端口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 12.1.1.1/24	配置接口的 ip 地址

Switch(config-if)# ip pim dense-mode	使能接口的 pim dm 功能
Switch(config-if)# exit	退出接口模式
Switch(config)# ip route 22.1.1.0/24 12.1.1.2	配置一条静态路由

Configuring R2

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入 eth-0-1 的接口模式
Switch(config-if)# no shutdown	启用端口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 22.1.1.2/24	配置接口的 ip 地址
Switch(config-if)# ip pim dense-mode	使能接口的 pim dm 功能
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入 eth-0-9 的接口模式
Switch(config-if)# no shutdown	启用端口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 12.1.1.2/24	配置接口的 ip 地址
Switch(config-if)# ip pim dense-mode	使能接口的 pim dm 功能
Switch(config-if)# exit	退出接口模式
Switch(config)# ip route 11.1.1.0/24 12.1.1.1	配置一条静态路由

I. 检查配置

使用下面的命令来检查接口配置和路由表信息。

接口的详细信息

用 show ip pim dense-mode interface 来显示 R1 上接口的详细信息.

R1# show ip pim dense-mode interface

```
Address          Interface VIFIndex Ver/   Nbr
                                    Mode   Count
11.1.1.1         eth-0-1   0        v2/D   0
12.1.1.1         eth-0-9   1        v2/D   1
```

邻居的详细信息

用 show ip pim dense-mode neighbor 来显示 R1 上邻居的详细信息

R1# show ip pim dense -mode neighbor

```
Neighbor-Address Interface          Uptime/Expires    Ver
12.1.1.2         eth-0-9            00:01:00/00:01:44 v2
```

组播路由表的信息

用 show ip pim dense-mode mroute detail 来显示 PIM-DM 组播路由表的信息

R1# show ip pim dense-mode mroute

```
PIM-DM Multicast Routing Table
(11.1.1.2, 225.1.1.1)
  Source directly connected on eth-0-1
  State-Refresh Originator State: Originator
  Upstream IF: eth-0-1
    Upstream State: Forwarding
    Assert State: NoInfo
  Downstream IF List:
    eth-0-9, in 'olist':
      Downstream State: NoInfo
      Assert State: NoInfo
```

R2# show ip pim dense-mode mroute

```
PIM-DM Multicast Routing Table
(11.1.1.2, 225.1.1.1)
  RPF Neighbor: none
  Upstream IF: eth-0-9
    Upstream State: AckPending
    Assert State: NoInfo
  Downstream IF List:
    eth-0-1, in 'olist':
      Downstream State: NoInfo
      Assert State: NoInfo
```

5.5 配置 IGMP Snooping

5.5.1 简介

IGMP Snooping（Internet Group Management Protocol Snooping，IGMP 侦听）是运行在二层以太网交换机上的组播约束机制，用于管理和控制组播组。

二层交换机通过 IGMP Snooping 来控制组播流量的泛洪。当二层以太网交换收到主机和路由器之间传递的 IGMP 报文时，IGMP Snooping 将对 IGMP 报文所带的信息进行分析，将端口和 MAC 组播地址建立起映射关系，并根据这样的映射关系转发组播数据。

组播路由器定期发送通用组查询来维护组播组成员关系。所有接收者将将发送 IGMP 报告报文来响应这个查询，交换机通过这个监听 IGMP 报告报文来建立转发表项。

二层的组播组可以通过 IGMP 报文动态建立，也可以静态配置。静态配置的组播组将覆盖动态学的组播组。

5.5.2 配置启用 IGMP Snooping

IGMP Snooping 可以在全局模式下启用或者每个 VLAN 下启用。假如 IGMP Snooping 在全局模式下关闭，即使你在每个 VLAN 下启用 IGMP Snooping 也是无效的。假如 IGMP Snooping 在全局模式下开启.可以在某个 VLAN 下关闭 IGMP Snooping，另一方面，全局配置可以覆盖每个 VLAN 配置。默认情况下，IGMPSnooping 在全局模式下和每个 VLAN 上使能。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)# ip igmp snooping	全局模式下启用 IGMP Snooping
Switch(config)#ip igmp snooping vlan 1	在单 VLAN 模式下启用 IGMP Snooping
Switch # show ip igmp snooping vlan 1	检查配置

II. 命令验证

Switch # show ip igmp snooping vlan 1

```
Global Igmp Snooping Configuration
-------------------------------------------------
Igmp Snooping                                :Enabled
Igmp Snooping Fast-Leave                     :Disabled
Igmp Snooping Version                        :2
Igmp Snooping Robustness Variable            :2
Igmp Snooping Max-Member-Number              :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Report-Suppression             :Enabled
Vlan 1
------------
Igmp Snooping                                :Enabled
Igmp Snooping Fast-Leave                     :Disabled
Igmp Snooping Report-Suppression             :Enabled
Igmp Snooping Version                        :2
Igmp Snooping Robustness Variable            :2
Igmp Snooping Max-Member-Number              :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Group Access-list              :N/A
Igmp Snooping Mrouter Port                   :
Igmp Snooping Mrouter Port Aging Interval(sec) :255
```

5.5.3 配置 IGMP Snooping 快速离开

正常情况下，IGMP Snooping 在接收到 IGMP 离开报文后不会直接将端口从组播组中删除，而是发送 IGMP 特定组查询报文，如果等待一段时间后没有得到响应，才将该端口从组播组中删除。启动快速删除功能后，IGMP Snooping 收到 IGMP 离开报文时，直接将端口从组播组中删除。当端口下只有一个用户时，快速删除可以节省带宽。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)#ip igmp snooping fast-leave	全局模式下启用快速离开功能
Switch(config)#ip igmp snooping vlan 1 fast-leave	在 VLAN 模式下启用快速离开功能
Switch# show ip igmp snooping vlan 1	检查配置

II. 命令验证

Switch # show ip igmp snooping vlan 1

```
Global Igmp Snooping Configuration
------------------------------------------------
Igmp Snooping                          :Enabled
Igmp Snooping Fast-Leave               :Enabled
Igmp Snooping Version                  :2
Igmp Snooping Robustness Variable      :2
Igmp Snooping Max-Member-Number        :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Report-Suppression       :Enabled
Vlan 1
-----------
Igmp Snooping                          :Enabled
Igmp Snooping Fast-Leave               :Enabled
Igmp Snooping Report-Suppression       :Enabled
Igmp Snooping Version                  :2
Igmp Snooping Robustness Variable      :2
Igmp Snooping Max-Member-Number        :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Group Access-list        :N/A
Igmp Snooping Mrouter Port             :
Igmp Snooping Mrouter Port Aging Interval(sec) :255
```

5.5.4 配置 IGMP Snooping 查询参数

三层交换机在所连接的网段上周期性的发送 IGMP 通用查询报文，通过解析返回的 IGMP 主机报告报文，获知该网段内哪些组播组有成员。组播路由器周期性地发送查询报文，当得到某一组成员的 IGMP 主机报告报文的时候，刷新该网段相应的组成员关系信息。

I. 配置

Switch #configure terminal	进入配置模式
Switch(config)# ip igmp snooping query-interval 100	设置查询时间间隔是 100 秒
Switch(config)# ip igmp snooping query-max-response-time 5	设置查询的最大响应时间 5 秒
Switch(config)#ip igmp snooping last-member-query-interval 2000	设置当仅存最后一个成员时的查询间隔
Switch(config)#ip igmp snooping vlan 1 querier address 10.10.10.1	在 VLAN1 上配置 IGMP Snooping 的查询地址
Switch(config)#ip igmp snooping vlan 1 querier	在 VLAN1 上启用 IGMP Snooping 的查询功能
Switch(config)#ip igmp snooping vlan 1 query-interval 200	在 VLAN1 上设置查询时间间隔是 200 秒
Switch(config)#ip igmp snooping vlan 1 query-max-response-time 5	在 VLAN1 上设置查询的最大响应时间 5 秒
Switch(config)#ip igmp snooping vlan 1 querier-timeout 100	在 VLAN1 上设置查询超时时间 100 秒
Switch(config)#ip igmp snooping vlan 1 last-member-query-interval 2000	在 VLAN1 上设置特定组的查询间隔 2000 秒
Switch(config)# ip igmp snooping vlan 1 discard-unknown	在 VLAN1 上丢弃未知组播报文
Switch(config)# ip igmp snooping discard-unknown	在全局模式下设置丢弃未知组播报文

II. 命令验证

Switch # show ip igmp snooping querier

```
Global Igmp Snooping Querier Configuration
-------------------------------------------------
Version                          :2
Last-Member-Query-Interval (msec) :2000
Last-Member-Query-Count           :2
Max-Query-Response-Time (sec)     :5
Query-Interval (sec)              :100
Global Source-Address             :0.0.0.0
TCN Query Count                   :2
TCN Query Interval (sec)          :10
TCN Query Max Respose Time (sec)  :5
```

```
Vlan 1:   IGMP snooping querier status
-----------------------------------------
Elected querier is : 0.0.0.0
-----------------------------------------
Admin state                          :Enabled
Admin version                        :2
Operational state                    :Non-Querier
Querier operational address          :10.10.10.1
Querier configure address            :10.10.10.1
Last-Member-Query-Interval (msec)    :2000
Last-Member-Query-Count              :2
Max-Query-Response-Time (sec)        :5
Query-Interval (sec)                 :200
Querier-Timeout (sec)                :100
```

5.5.5 配置 IGMP Snooping 组播路由端口

组播路由端口是交换机上连接到组播路由器的端口，可以动态学习或者静态配置。当某个 VLAN 的端口上收到 IGMP 通用组查询报文或者是 PIMv2 Hello 报文，该端口成为这个 VLAN 的组播路由端口。所有从组播路由端口上收到的的 IGMP 查询报文要在所属 VLAN 内广播。所有 VLAN 上收到 IGMP 报告/离开报文也将从组播路由端口转发(报文抑制关闭的情况下)，另外所有从该 VLAN 上收到的组播流量将从组播路由端口转发。

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# ip igmp snooping report-suppression	启用 IGMP Snooping 的报告抑制功能
Switch(config)# ip igmp snooping vlan 1 mrouter interface eth-0-1	配置静态组播路由端口
Switch(config)# ip igmp snooping vlan 1 report-suppression	在 VLAN1 上启用报告抑制功能
Switch(config)# ip igmp snooping vlan 1 mrouter-aging-interval 200	配置动态组播路由端口老化时间

II. 命令验证

Switch# show ip igmp snooping vlan 1

```
Global Igmp Snooping Configuration
-----------------------------------------------
Igmp Snooping                              :Enabled
Igmp Snooping Fast-Leave                   :Disabled
Igmp Snooping Version                      :2
Igmp Snooping Robustness Variable          :2
Igmp Snooping Max-Member-Number            :2048
```

```
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Report-Suppression          :Enabled
Vlan 1
-----------
Igmp Snooping                             :Enabled
Igmp Snooping Fast-Leave                  :Disabled
Igmp Snooping Report-Suppression          :Enabled
Igmp Snooping Version                     :2
Igmp Snooping Robustness Variable         :2
Igmp Snooping Max-Member-Number           :2048
Igmp Snooping Unknown Multicast Behavior  :Flood
Igmp Snooping Group Access-list           :N/A
Igmp Snooping Mrouter Port                :eth-0-1
Igmp Snooping Mrouter Port Aging Interval(sec) :200
```

5.5.6 配置 IGMP Snooping 查询 TCN

可以通过配置 TCN 的时间间隔以及查询次数来适应 STP 收敛拓扑后的组播组学习以及更新。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)# ip igmp snooping querier tcn query-count 5	设置 TCN 的查询次数
Switch(config)# ip igmp snooping querier tcn query-interval 20	设置 TCN 的查询时间间隔 20 秒

II. 命令验证

Switch # show ip igmp snooping querier

```
Global Igmp Snooping Querier Configuration
-------------------------------------------------
Version                          :2
Last-Member-Query-Interval (msec) :1000
Max-Query-Response-Time (sec)     :10
Query-Interval (sec)              :125
Global Source-Address             :0.0.0.0
TCN Query Count                   :5
TCN Query Interval (sec)          :20
Vlan 1:   IGMP snooping querier status
-----------------------------------------
Elected querier is : 0.0.0.0
-----------------------------------------
Admin state                       :Disabled
Admin version                     :2
Operational state                 :Non-Querier
Querier operational address       :0.0.0.0
```

```
Querier configure address          :N/A
Last-Member-Query-Interval (msec) :1000
Max-Query-Response-Time (sec)      :10
Query-Interval (sec)               :125
Querier-Timeout (sec)              :255
```

5.5.7 配置 IGMP Snooping 报告抑制

交换机使用 IGMP 报告抑制来同一个 IGMP 报文重复发送给组播路由器。当 IGMP 路由器抑制使能时(默认)，交换机将第一个 IGMP 报告报文发送给组播路由器，其余同样的 IGMP 报告报文将不再发送给组播路由器。这样就阻止了重复 IGMP 报告报文发送给组播路由器了。

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# ip igmp snooping report-suppression	在全局模式下启用报告抑制
Switch(config)# ip igmp snooping vlan 1 report-suppression	在 VLAN1 模式下启用报告抑制

II. 命令验证

Switch # show ip igmp snooping

```
Global Igmp Snooping Configuration
-------------------------------------------------
Igmp Snooping                                :Enabled
Igmp Snooping Fast-Leave                     :Disabled
Igmp Snooping Version                        :2
Igmp Snooping Robustness Variable            :2
Igmp Snooping Max-Member-Number              :2048
Igmp Snooping Unknown Multicast Behavior     :Flood
Igmp Snooping Report-Suppression             :Enabled
Vlan 1
-----------
Igmp Snooping                                :Enabled
Igmp Snooping Fast-Leave                     :Disabled
Igmp Snooping Report-Suppression             :Enabled
Igmp Snooping Version                        :2
Igmp Snooping Robustness Variable            :2
Igmp Snooping Max-Member-Number              :2048
Igmp Snooping Unknown Multicast Behavior     :Flood
Igmp Snooping Group Access-list              :N/A
Igmp Snooping Mrouter Port                   :
Igmp Snooping Mrouter Port Aging Interval(sec) :255
```

5.5.8 配置静态组播组

交换机在二层端口上收到 IGMP 报文时会建立 IGMP Snooping 的组记录。目前系统中也支持静态配置 IGMP Snoopoing 的组记录，在静态配置时需要指定组地址，二层端口，以及二层端口所属的 VLAN。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)# ip igmp snooping vlan 1 static-group 229.1.1.1 interface eth-0-2	配置静态组播组 229.1.1.1，成员端口是 vlan1 的 eth-0-2

II. 命令验证

Switch# show ip igmp snooping groups

```
VLAN    Interface       Group-Address       Uptime      Expires-time
1       eth-0-2         229.1.1.1           00:01:08    stopped
```

5.5.9 限制和配置指导

VRRP，RIP，OSPF 等协议使用了组播 IP，因此在使能了 IGMP Snooping 的网络中，要避免使用这样的组播 IP，这些组播 IP 是，它们映射出来的 MAC 和被协议模块使用的组播 IP 映射出来的 MAC 一致。

VRRP 使用了 224.0.0.18，因此组播 MAC 0100.5E00.0012 映射出的组播 IP 在 IGMP Snooping 和 VRRP 的网络中避免使用。

RIP 使用了 224.0.0.9，因此组播 MAC 0100.5E00.0009 映射出的组播 IP 在 IGMP Snooping 和 RIP 的网络中避免使用。

OSPF 使用了 224.0.0.5，因此组播 MAC 0100.5E00.0005 映射出的组播 IP 在 IGMP Snooping 和 OSPF 的网络中避免使用。

5.6 配置 MVR

5.6.1 简介

在传统的组播点播方式下，汇聚组播路由器下连一些接入交换机，接入交换机上连接了分布在不同 VLAN 中的用户。当这些不同 VLAN 的用户点播相同 Group 的节目时，汇聚的组播路由器需要为每个 VLAN 内的用户复制一份数据，每个 VLAN 的组播流量都要占用接入交换机的带宽。这样即增加了汇聚路由器的负担，也浪费接入设备的带宽。

MVR(组播 VLAN 注册)功能能够很好的解决这个问题。在靠近用户侧的接入交换机上启用组播 VLAN，汇聚路由器只需把组播数据在源 VLAN 内发送给接入交换机，而不

必在每个用户 VLAN 内都复制一份，接入交换机收到组播数据后再根据用户请求进行复制，给每个 VLAN 内的用户发送一份组播数据。从而节省了网络带宽，也减轻了三层设备的负担。

MVR 依赖于 IGMP Snooping 进行工作，而且只有 MVR 全局配置的 Group 才会生效。如果在 MVR 的下游口上接收的 IGMP 报文中组播组不在 MVR 全局 Group 中，该报文将被忽略。通过在 MVR 的下游口上接收的 IGMP 报告/离开报文来维护接收者信息，MVR 上游口收到组播数据后根据下游口的组播组信息来决定将组播数据从哪些 VLAN 的端口转发出去。

5.6.2 术语

MVR：组播 VLAN 注册

Source vlan：组播 VLAN 的源 VLAN

Source port：MVR 网络中的上游口，连接组播路由器的端口

Receiver port：MVR 网络中的下游口，连接接收者的端口

5.6.3 拓扑

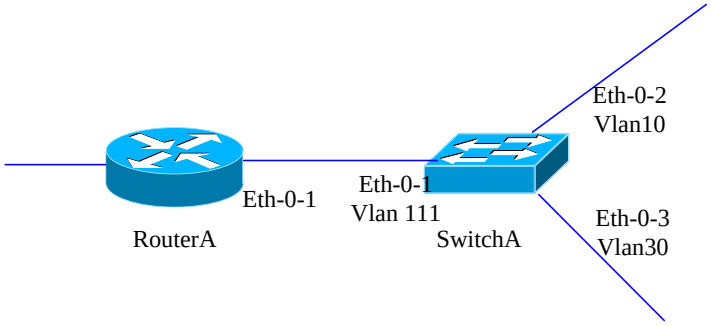


图5-3 组播 VLAN 拓扑

5.6.4 配置

目的

在 Router A 的 eth-0-1 上启用 IGMP&PIM-SM。

配置 Switch A：eth-0-1 属于 vlan111，eth-0-2 属于 vlan10，eth-0-3 属于 vlan30。

在 Switch A 启用 MVR，从 Router A 到 Switch A 上拷贝一份组播流，在 Switch A 上再将这个组播流进行复制，从 eth-0-2 和 eth-0-3 发送出去。

Router A

在配置接口上启用 IGMP&PIM-SM。

RouterA# configure terminal	进入配置模式
RouterA(config)# interface eth-0-1	进入接口模式
RouterA(config-if)# no switchport	设置端口为三层端口
RouterA(config-if)# no shutdown	使能端口
RouterA(config-if)# ip address 12.12.12.12/24	配置 IP 地址
RouterA(config-if)# ip pim sparse-mode	启用 PIM-SM 协议
RouterA(config-if)# end	返回全局模式

Switch A

配置 eth-0-1 属于 vlan111，eth-0-2 属于 vlan10，eth-0-3 属于 vlan30。

SwitchA# configure terminal	进入配置模式
SwitchA(config)# vlan database	进入 VLAN 模式
SwitchA(config-vlan)# vlan 111,10,30	创建 vlan 111，10，30
SwitchA(config-vlan)# quit	退出 VLAN 模式
SwitchA(config)# interface vlan 111	进入 VLAN 接口模式
SwitchA(config-if)# exit	退出 VLAN 接口模式
SwitchA(config)# interface vlan 10	进入 VLAN 接口模式
SwitchA(config-if)# exit	退出 VLAN 接口模式
SwitchA(config)# interface vlan 30	进入 VLAN 接口模式
SwitchA(config-if)# exit	退出 VLAN 接口模式
SwitchA(config)# interface eth-0-1	进入接口模式
SwitchA(config-if)# switchport access vlan111	设置端口属于 VLAN111
SwitchA(config)# interface eth-0-2	进入接口模式
SwitchA(config-if)# switchport access vlan10	设置端口属于 VLAN10
SwitchA(config)# interface eth-0-3	进入接口模式
SwitchA(config-if)# switchport access vlan30	设置端口属于 VLAN30
SwitchA(config-if)# end	退出接口模式

在 switch A 启用 MVR，这样从 Router A 到 Switch A 只会拷贝一份组播流，在 Switch A 上再将这个组播流从 eth-0-2 和 eth-0-3 发送出去。

SwitchA # configure terminal	进入配置模式
SwitchA(config)# no ip multicast-routing	关闭 IP 组播路由
SwitchA(config)# mvr	启用 MVR
SwitchA(config)# mvr vlan 111	创建 MVR 的 VLAN
SwitchA(config)# mvr group 238.255.0.1 64	创建组播组
SwitchA(config)# mvr source-address 12.12.12.1	配置 MVR 源地址
SwitchA(config)# interface eth-0-1	进入接口模式
SwitchA(config-if)# mvr type source	配置接口为 MVR 的源端口
SwitchA(config)# interface eth-0-2	进入接口模式
SwitchA(config-if)# mvr type receiver vlan 10	设置接口为 MVR 的接收端口
SwitchA(config)# interface eth-0-3	进入接口模式
SwitchA(config-if)# mvr type receiver vlan 30	设置接口为 MVR 的接收端口
SwitchA(config-if)# end	退出接口模式

5.6.5 命令验证

Router A

RouterA # show ip igmp groups

```
IGMP Connected Group Membership
Group Address    Interface            Uptime   Expires  Last Reporter
238.255.0.1      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.2      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.3      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.4      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.5      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.6      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.7      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.8      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.9      eth-0-1              00:01:16 00:03:49 12.12.12.1
238.255.0.10     eth-0-1              00:01:16 00:03:49 12.12.12.1
```

```
.............
238.255.0.64     eth-0-1               00:01:16 00:03:49 12.12.12.1
```

Switch A

SwitchA# show mvr

```
MVR Running: TRUE
MVR Multicast VLAN: 111
MVR Source-address: 12.12.12.1
MVR Max Multicast Groups: 1024
MVR Hw Rt Limit: 508
MVR Current Multicast Groups: 255
```

SwitchA# show mvr groups

```
VLAN    Interface       Group-Address       Uptime      Expires-time
10      eth-0-2         238.255.0.1         00:03:23    00:02:03
10      eth-0-2         238.255.0.2         00:02:16    00:02:03
10      eth-0-2         238.255.0.3         00:02:16    00:02:03
10      eth-0-2         238.255.0.4         00:02:16    00:02:03
10      eth-0-2         238.255.0.5         00:02:16    00:02:03
10      eth-0-2         238.255.0.6         00:02:16    00:02:04
10      eth-0-2         238.255.0.7         00:02:16    00:02:04
10      eth-0-2         238.255.0.8         00:02:16    00:02:04
10      eth-0-2         238.255.0.9         00:02:16    00:02:04
10      eth-0-2         238.255.0.10        00:02:16    00:02:04
......
10      eth-0-2         238.255.0.64       00:01:50    00:02:2
```

6 安全性配置指导

6.1 端口安全配置

6.1.1 简介

端口安全功能是用来限制一个特定的接口上可靠 MAC 地址的数量。该接口将只向前转发源 MAC 地址匹配这些安全地址的数据包。MAC 地址可以手动创建，或自动学习。MAC 地址的数量达到安全 MAC 地址数量的限制后，新的 MAC 地址在接口上不能学到。如果接口又接收到新数据包，且数据包源 MAC 地址与任何安全地址是不同的，它被视为违反安全。

端口安全将 MAC 地址绑定到端口，源 MAC 不是这些地址的报文从端口进入后不会被转发。如果安全 MAC 地址在接口上已学习到，但该 MAC 地址又试图从别的接口学习或配置到别的接口，这也被认为是违反安全。

支持两种类型的安全 MAC 地址是：

- 静态安全 MAC 地址：这是手动配置的 MAC 地址。
- 动态安全 MAC 地址：这是动态学习。

如果发生违反安全，要转发的数据包将被丢弃。

6.1.2 配置

按下列步骤配置端口安全。

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# switchport	设置端口为二层端口
Switch(config-if)# switchport port-security	启用端口安全功能
Switch(config-if)# switchport port-security maximum 3	设置最大的安全条目
Switch(config-if)# switchport port-security mac-address 0000.1111.2222 vlan 1	绑定 mac 地址到接口

Switch(config-if)# switchport port-security mac-address 0000.aaaa.bbbb vlan 1	绑定 mac 地址到接口
Switch(config-if)# switchport port-security violation restrict	设置端口的限制模式
Switch(config-if)# end	退出接口模式
Switch# show port-security	检查配置

6.1.3 命令验证

Switch# show port-security

```
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolationMode
               (Count)        (Count)
------------------------------------------------------------
  eth-0-1           3             2       restrict
Switch# show port-security address-table
          Secure MAC address table
------------------------------------------------------
Vlan    Mac Address       Type                Ports
----    -----------       -------             -------
1       0000.1111.2222    SecureConfigured    eth-0-1
1       0000.aaaa.bbbb    SecureConfigured    eth-0-1
```

Switch# show port-security interface eth-0-1

```
Port security                     : enabled
Violation mode                    : discard packet and log
Maximum MAC addresses             : 3
Total MAC addresses               : 2
Static configured MAC addresses   : 2
```

6.2 VLAN 安全配置

6.2.1 简介

VLAN 安全通过限制 VLAN 内 MAC 地址的数量达到保护 VLAN 的目的。MAC 地址可以是用户手动添加的，也可以是自动学习的。Vlan 内 MAC 地址达到限制数量后，未知源 MAC 的报文就会被丢弃（可指定行为）。

系统支持两种类型的 MAC 地址：

- 静态 MAC 地址：手工配置的 MAC 地址
- 动态 MAC 地址：通过动态学习的 MAC 地址

用户可以指定当 VLAN 内 MAC 达到限制数量时的如下行为之一：

- Discard：丢弃未知源 MAC 地址的报文

- Warn：丢弃未知的源 MAC 地址报文，并且在 LOG 中提示
- Forward：报文正常转发，但是 MAC 不会进行学习。

系统还支持开、关 VLAN 内 MAC 地址学习功能。

6.2.2 配置 VLAN MAC 地址限制

通过以下操作可以配置如何限制 MAC 地址。

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config)# vlan 2	创建 VLAN2
Switch(config-vlan)# vlan 2 mac-limit maximum 100	配置 VLAN2 的最大 MAC 地址数量
Switch(config-vlan)# vlan 2 mac-limit action discard	配置学满地址后的行为为丢弃
Switch(config-vlan)#end	退出 VLAN 模式
Switch #show vlan-security	检查配置

6.2.3 配置 VLAN MAC 地址学习

通过以下操作关闭 VLAN 的 MAC 地址学习功能。

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN 模式
Switch(config)# vlan 2	创建 VLAN
Switch(config-vlan)# vlan 2 mac learning disable	取消 MAC 地址学习
Switch(config-vlan)#end	退出 VLAN 模式
Switch #show vlan-security	检查配置

6.2.4 命令验证

Switch# show vlan-security

```
Vlan  learning-en  max-mac-count  cur-mac-count  action
-------------------------------------------------------
2     Disable      100            0              Discard
```

6.3 Time-Range 配置

6.3.1 简介

Time range 定义了一段时间，这段时间可以是绝对时间，也可以是相对的周期性时间。Time range 本身没有意义，通常被用在基于时间的协议或者应用中（比如 acl）。在实际应用中，它可以表示在这段时间内，某些规则或操作有效。Time range 定义的时间依赖于系统时钟。

6.3.2 配置

配置绝对时间段

Switch# configure terminal	进入全局配置模式
Switch(config)# time-range test-absolute	创建 time-range，进入时间段配置模式
Switch(config-tm-range)# absolute start 1:1:2 jan 1 2012 end 1:1:3 jan 7 2012	创建 绝对时间
Switch(config-tm-range)# end	退出时间段配置模式

配置周期时间段

Switch# configure terminal	进入全局配置模式
Switch(config)# time-range test-periodic	创建 time-range，进入时间段配置模式
Switch(config-tm-range)# periodic 1:1 mon to 1:1 wed	创建周期时间
Switch(config-tm-range)# end	退出时间段配置模式

6.3.3 命令验证

DUT1# show time-range

```
time-range test-absolute
   absolute start 01:01:02 Jan 01 2012 end 01:01:03 Jan 07 2012
time-range test-periodic
```

```
periodic 01:01 Mon to 01:01 Wed
```

6.4 访问控制列表配置

6.4.1 简介

ACL（Access Control List，访问控制列表）主要用来实现流识别、访问控制功能。网络设备为了过滤数据包，需要配置一系列的匹配规则，以识别需要过滤的报文。在识别出特定的报文之后，才能根据预先设定的策略允许或禁止相应的数据包通过。ACL通过一系列的匹配条件对数据包进行分类，这些条件可以是数据包的源地址、目的地址、端口号等。

6.4.2 术语

下面简要介绍用于描述 ACL 相关的术语和概念：

访问控制条目（ACE）

每一个 ACE 包括一个动作元素（允许或者拒绝）和一个基于标准过滤元素，例如源地址、目的地址、协议、特定协议参数等等。

MAC ACL

MAC ACL 可以根据 MAC-SA 和 MAC-DA 过滤报文，MAC 地址可以配置掩码，或者配置为主机 MAC。MAC ACL 也可以根据其他二层字段过滤报文，例如 COS、VLAN-ID、INNER-COS、INNER-VLAN-ID、L2 type、L3 type。

IPv4 ACL

IPv4 ACL 可以根据 IP-SA 和 IP-DA 过滤报文，IP 地址可以配置掩码或者配置为主机 IP 地址。IPv4 ACL 也可以根据其他三层字段过滤报文，例如 DSCP、L4 Protocol 字段以及其他字段（TCP 端口、UDP 端口等等）。

时间段

定义一个时间周期，在这个时间段内，ACE 是有效的。

6.4.3 限制

如果进入的报文只有一个 VLAN tag，字段 inner-cos 和 inner-vlan-id 将被默认设置为 0，因此在报文有一个 vlan tag 还是有两个 vlan tag 的情况下，inner-vlan-id 和 inner-cos 的配置将产生不同的作用。

6.4.4 配置

下面这个例子中，在端口 eth-0-1 上使用 MAC ACL，允许源 MAC 地址为 0000.0000.1111 的报文通过，拒绝其他报文。在 eth-0-2 使用 IPv4 ACL，允许源 IP 地址为 1.1.1.1/24 的报文通过，拒绝其他报文。

ACL 配置细则

Switch# configure terminal	进入全局配置模式
Switch(config)# mac access-list mac	创建并进入 MAC ACL 配置模式
Switch(config-mac-acl)# permit src-mac host 0000.0000.1111 dest-mac any	添加条目，设置允许源 MAC 地址为 0000.0000.1111 帧通过
Switch(config-mac-acl)# deny src-mac any dest-mac any	添加条目，设置拒绝任何 MAC 帧通过
Switch(config-mac-acl)# exit	退出 ACL 配置模式
Switch(config)# ip access-list ipv4	创建并进入 IPv4 ACL 配置模式
Switch(config-ip-acl)# permit any 1.1.1.1 0.0.0.255 any	添加条目，设置允许源 IP 地址为 1.1.1.1 0.0.0.255 帧通过
Switch(config-ip-acl)# deny any any any	添加条目，设置拒绝任何帧通过
Switch(config-ip-acl)# exit	退出 ACL 配置模式

接口配置细则

Switch# configure terminal	进入全局配置模式
Switch(config)# class-map cmap1	创建分类映射表 cmap1, 并且进入分类映射表配置模式
Switch(config-cmap)# match access-group mac	将 mac ACL 加入 cmap1
Switch(config-cmap)# exit	退出分类映射表配置模式
Switch(config)# policy-map pmap1	创建策略表 pmap1, 并且进入策略表配置模式
Switch(config-pmap)# class cmap1	将流分类映射表 cmap1 加入策略映射表 pmap1, 并且进入策略表中的分类映射表配置模式
Switch(config-pmap-c)# exit	退出策略表中的分类映射表配置模式

Switch(config-pmap)# exit	退出策略表配置模式
Switch(config)# interface eth-0-1	进入要应用此 ACL 的端口配置模式
Switch(config-if)# service-policy input pmap1	将策略表 pmap1 应用到接口, 策略表 pmap1 中引用了 cmap1, cmap1 引用了 mac ACL, 因此这个接口就应用的 mac ACL
Switch(config-if)# exit	退出接口配置模式
Switch(config)# class-map cmap2	创建分类映射表 cmap2, 并且进入分类映射表配置模式
Switch(config-cmap)# match access-group ipv4	将 ACL ipv4 加入 cmap2
Switch(config-cmap)# exit	退出分类映射表配置模式
Switch(config)# policy-map pmap2	创建策略表 pmap2, 并且进入策略表配置模式
Switch(config-pmap)# class cmap2	将流分类映射表 cmap2 加入策略映射表 pmap2, 并且进入策略表中的分类映射表配置模式
Switch(config-pmap-c)# exit	退出策略表中的分类映射表配置模式
Switch(config-pmap)# exit	退出策略表配置模式
Switch(config-if)# interface eth-0-2	进入要应用此 ACL 的端口配置模式
Switch(config-if)# service-policy input pmap2	将策略表 pmap1 应用到接口, 策略表 pmap2 中引用了 cmap2, cmap2 引用了 ACL ipv4, 因此这个接口就应用的 ACL ipv4

6.4.5 命令验证

使用命令 show running-config，屏幕回显内容如下所示。

Switch# show running-config

```
mac access-list mac
  10 permit src-mac host 0000.0000.1111 dest-mac any
  20 deny src-mac any dest-mac any
!
ip access-list ipv4
  10 permit any 1.1.1.0 0.0.0.255 any
  20 deny any any any
!
```

```
class-map match-any cmap1
 match access-group mac
!
class-map match-any cmap2
 match access-group ipv4
!
policy-map pmap1
 class cmap1
!
policy-map pmap2
 class cmap2
!
interface eth-0-1
 service-policy input pmap1
!
interface eth-0-2
 service-policy input pmap2
!
```

6.5 扩展 ACL 配置

6.5.1 简介

扩展 IPV4 ACL 包含 MAC ACE 和 IP ACE，MAC ACE 匹配所有非 IPV6 和非 MPLS 报文，IP ACE 匹配所有 IPV4 报文。

6.5.2 术语

下面介绍了扩展 ACL 有关的术语和概念：

扩展 IPV4 ACL：包含 MAC ACE 和 IP ACE。

MAC ACE 可以根据 MAC-SA 和 MAC-DA 过滤报文，MAC 地址可以配置掩码，或者配置为主机 MAC；也可以根据其他二层字段过滤报文，例如 COS、VLAN-ID、INNER-COS、INNER-VLAN-ID、L2 type、L3 type。

IPv4 ACE 可以根据 IP-SA 和 IP-DA 过滤报文，IP 地址可以配置掩码或者配置为主机 IP 地址；也可以根据其他三层字段过滤报文，例如 DSCP、L4 Protocol 字段以及其他字段（TCP 端口、UDP 端口等等）。

用户可以通过 MAC ACE 和 IP ACE 各种组合，以及不同的顺序实现不同的需求。

6.5.3 配置

下面的例子描述如何通过扩展 IPV4 ACL 实现在端口 eth-0-1 上允许源 MAC 为 0.0.1111 报文 COS 为 2 的报文，允许所有 TCP 的报文，禁止其他报文进入系统。

Extend ACL 配置细则

Switch# configure terminal	进入全局配置模式
Switch(config)# ip access-list ipxacl extend	定义一个名称为 ipxacl 的扩展 IPV4 ACL
Switch(config-ex-ip-acl)# permit src-mac host 0000.0000.1111 dest-mac any cos 2	添加一条允许源 MAC0.0.1111、cos 2 报文的 ACE
Switch(config-ex-ip-acl)# permit tcp any any	添加一条允许 TCP 报文的 ACE
Switch(config-ex-ip-acl)# deny src-mac any dest-mac any	添加一条拒绝所有报文的 ACE
Switch(config-ex-ip-acl)# end	退出到特权模式

接口配置细则

Switch# configure terminal	进入全局配置模式
Switch(config)# class-map cmap	创建分类映射表 cmap, 并且进入分类映射表配置模式
Switch(config-cmap)# match access-group ipxacl	将 ipxacl ACL 加入 cmap
Switch(config-cmap)# exit	退出分类映射表配置模式
Switch(config)# policy-map pmap	创建策略表 pmap1, 并且进入策略表配置模式
Switch(config-pmap)# class cmap	将流分类映射表 cmap1 加入策略映射表 pmap1, 并且进入策略表中的分类映射表配置模式
Switch(config-pmap-c)# exit	退出策略表中的分类映射表配置模式
Switch(config-pmap)# exit	退出策略表配置模式
Switch(config)# interface eth-0-1	进入要应用此 ACL 的端口配置模式
Switch(config-if)# service-policy input pmap	将策略表 pmap1 应用到接口, 测量表 pmap1 中引用了 cmap1, cmap1 引用了 ACL mac, 因此这个接口就应用的 ACL mac
Switch(config-if)# exit	退出接口配置模式

6.5.4 命令验证

使用如下命令验证配置结果。

Switch# show running-config

```
ip access-list ipxacl extend
  10 permit src-mac host 0000.0000.1111 dest-mac any cos 2
  20 permit tcp any any
  30 deny src-mac any dest-mac any
!
class-map match-any cmap
 match access-group ipxacl
!
policy-map pmap
 class cmap
!
interface eth-0-1
 service-policy input pmap
!
Switch# show access-list ip
ip access-list ipxacl extend
  10 permit src-mac host 0000.0000.1111 dest-mac any cos 2
  20 permit tcp any any
  30 deny src-mac any dest-mac any
```

6.6 访问控制列表 v6 配置

6.6.1 简介

ACLv6（Access Control List，访问控制列表）主要用来实现 IPv6 流识别、访问控制功能。网络设备为了过滤数据包，需要配置一系列的匹配规则，以识别需要过滤的报文。在识别出特定的报文之后，才能根据预先设定的策略允许或禁止相应的数据包通过。ACL 通过一系列的匹配条件对数据包进行分类，这些条件可以是数据包的源地址、目的地址、端口号等。

6.6.2 术语

下面简要介绍用于描述 ACLv6 相关的术语和概念：

访问控制条目（ACE）

每一个 ACE 包括一个动作元素（允许或者拒绝）和一个基于标准过滤元素，例如源地址、目的地址、协议、特定协议参数等等。

IPv6 ACL

IPv6 ACL 可以根据 IP-SA 和 IP-DA 过滤报文，IP 地址可以配置掩码或者配置为主机 IP 地址。IPv6 ACL 也可以根据其他三层字段过滤报文，例如 L4 Protocol 字段以及其他字段（TCP 端口、UDP 端口等等）。

时间段

定义一个时间周期，在这个时间段内，ACE 是有效的。

6.6.3 限制

在全局启用 IPv6 后，IPv6 报文将不被 MAC ACL 所影响。

6.6.4 配置

下面这个例子中，在端口 eth-0-1 上使用 MAC ACL，允许源 MAC 地址为 0000.0000.1111 的非 IPv6 报文通过，拒绝其他非 IPv6 报文。在 eth-0-2 使用 IPv6 ACL，允许源 IP 地址为 2001::/64 的报文通过，拒绝其他报文。

ACL 配置细则

Switch# configure terminal	进入全局配置模式
Switch# ipv6 enable	全局启用 IPv6 功能
Switch(config)# mac access-list mac	创建并进入 MAC ACL 配置模式
Switch(config-mac-acl)# permit src-mac host 0000.0000.1111 dest-mac any	添加条目，设置允许目的 MAC 地址为 0000.0000.1111 帧通过
Switch(config-mac-acl)# deny src-mac any dest-mac any	添加条目，设置拒绝任何 MAC 帧通过
Switch(config-mac-acl)# exit	退出 ACL 配置模式
Switch(config)# ipv6 access-list ipv6	创建并进入 IPv6 ACL 配置模式
Switch(config-ipv6-acl)# permit any 2001::/64 any	添加条目，设置允许源 IPv6 地址为 2001::/64 帧通过
Switch(config-ipv6-acl)# deny any any any	添加条目，设置拒绝任何帧通过
Switch(config-ipv6-acl)# exit	退出 ACL 配置模式

接口配置细则

Switch# configure terminal	进入全局配置模式
Switch(config)# class-map cmap1	创建分类映射表 cmap1, 并且进入分类映射表配置模式
Switch(config-cmap)# match access-group mac	将 mac ACL 加入 cmap1
Switch(config-cmap)# exit	退出分类映射表配置模式
Switch(config)# policy-map pmap1	创建策略表 pmap1, 并且进入策略表配置模式
Switch(config-pmap)# class cmap1	将流分类映射表 cmap1 加入策略映射表 pmap1, 并且进入策略表中的分类映射表配置模式
Switch(config-pmap-c)# exit	退出策略表中的分类映射表配置模式
Switch(config-pmap)# exit	退出策略表配置模式
Switch(config)# interface eth-0-1	进入要应用此 ACL 的端口配置模式
Switch(config-if)# service-policy input pmap1	将策略表 pmap1 应用到接口, 策略表 pmap1 中引用了 cmap1, cmap1 引用了 mac ACL, 因此这个接口就应用的 mac ACL
Switch(config-if)# exit	退出接口配置模式
Switch(config)# class-map cmap2	创建分类映射表 cmap2, 并且进入分类映射表配置模式
Switch(config-cmap)# match access-group ipv6	将 ACL ipv6 加入 cmap2
Switch(config-cmap)# exit	退出分类映射表配置模式
Switch(config)# policy-map pmap2	创建策略表 pmap2, 并且进入策略表配置模式
Switch(config-pmap)# class cmap2	将流分类映射表 cmap2 加入策略映射表 pmap2, 并且进入策略表中的分类映射表配置模式
Switch(config-pmap-c)# exit	退出策略表中的分类映射表配置模式
Switch(config-pmap)# exit	退出策略表配置模式
Switch(config-if)# interface eth-0-2	进入要应用此 ACL 的端口配置模式
Switch(config-if)# service-policy input	将策略表 pmap1 应用到接口, 策略表

pmap2	pmap2 中引用了 cmap2, cmap2 引用了 ACL ipv4, 因此这个接口就应用的 ACL ipv4

6.6.5 命令验证

使用命令 show running-config，屏幕回显内容如下所示。

Switch# show running-config

```
mac access-list mac
  10 permit src-mac host 0000.0000.1111 dest-mac any
  20 deny src-mac any dest-mac any
!
ipv6 access-list ipv6
  10 permit any 2001::/64 any
  20 deny any any any
!
class-map match-any cmap1
 match access-group mac
!
class-map match-any cmap2
 match access-group ipv4
!
policy-map pmap1
 class cmap1
!
policy-map pmap2
 class cmap2
!
interface eth-0-1
 service-policy input pmap1
!
interface eth-0-2
 service-policy input pmap2
!
```

6.7 Dot1x 配置

6.7.1 简介

IEEE 802 网络在实际部署中，不可避免的会出现未经授权的设备在物理上接入到网络中。

802.1x 协议提供一种基于端口的网络接入控制协议（port based network access control protocol）。“基于端口的网络接入控制”是指在局域网接入设备的端口这一级对所接入

的用户设备进行认证和控制。连接在端口上的用户设备如果能通过认证，就可以访问局域网中的资源；如果不能通过认证，则无法访问局域网中的资源。

使用 802.1x 的系统为典型的 Client/Server 体系结构，包括三个实体：

- 客户端—设备（PC）请求访问 LAN 和交换机服务，响应来自交换机的请求。工作站必须运行符合 802.1X 客户端软件，如 Linux 的 xsupplicant。
- 认证服务器—执行客户端的实际认证。认证服务器验证客户的身份，并通知交换机客户端是否具有访问 LAN 和交换机服务的权限。由于交换机作为代理，认证服务对客户端是透明的。在此版本中，支持可扩展身份验证协议（EAP）的远程身份验证拨号用户服务（RADIUS）服务器是唯一支持的认证服务器。 RADIUS 工作于客户机/服务器模式，服务器和多个 RADIUS 客户端之间交换安全的身份验证信息。
- 交换机（边缘交换机或无线接入点）—控制基于客户端的认证状态网络的物理访问。交换机作为客户端和认证服务器之间的中介（代理），从客户端请求身份信息，通过认证服务器检查这些信息，并将认证结果返回到客户端。交换机包含 RADIUS 客户端，负责 EAP 帧的封装和解封，以及与认证服务器交互。当交换机收到 EAPOL 帧并中继到身份验证服务器时，以太网报头被剥离，剩下的 EAP 帧则重新封装为 RADIUS 格式。 EAP 帧在封装期间不会被修改或审查，并且验证服务器必须支持 EAP 在本机的帧格式。当交换机接收到来自验证服务器的报文，将服务器的帧头去掉，然后将剩下的 EAP 帧封装为以太网报文格式并发送到客户端。我们还可以在路由端口上配置 dot1x。

6.7.2 拓扑

图6-1 dot1x 基本拓扑图

6.7.3 配置

在普通的 2 层端口上使能 dot1x，配置步骤如下表所示。

Switch# configure terminal	进入全局配置模式
Switch(config)# dot1x system-auth-ctrl	全局启用 dot1x 认证控制
Switch(config)# interface eth-0-25	指定要配置的接口，进入接口配置模式
Switch(config)# switchport mode access	设置 Eth-0-25 为 access 模式
Switch(config-if)# dot1x port-control	在接口上启用 dot1x 端口控制

auto	
Switch(config-if)# no shutdown	确定端口使能
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface vlan 1	进入 VLAN 1
Switch(config-if)# ip address 192.168.100.1/24	设置 VLAN1 的 IP 地址
Switch(config)# interface eth-0-26	进入接口配置模式.
Switch(config-if)# no switchport	配置接口为路由端口
Switch(config-if)# ip address 202.38.100.1/24	在此接口上配置 IP 地址
Switch(config-if)# no shutdown	确定端口使能
Switch(config-if)# exit	退出接口配置模式.
Switch(config)# radius-server host 202.38.100.7	为 RADIUS 服务器配置 IPv4 地址
Switch(config)# radius-server host 2001:1000::1	为 RADIUS 服务器配置 IPv6 地址
Switch(config)# radius-server key test	配置 RADIUS 服务器的共享密钥
Switch(config)# end	退出配置模式
Switch# show dot1x	验证管理 dot1x 的配置
Switch# show dot1x interface eth-0-25	验证在 eth-0-25 的 dot1x 的配置

若要在路由端口上启用 dot1x，交换机配置步骤如下表所示。

Switch# configure terminal	进入全局配置模式
Switch(config)# dot1x system-auth-ctrl	全局使能 dot1x 身份验证控制
Switch(config)# interface eth-0-25	进入接口配置模式
Switch(config-if)# no switchport	配置接口为路由端口
Switch(config-if)# ip address 192.168.100.1/24	在此接口上配置 IP 地址
Switch(config-if)# dot1x port-control auto	在接口上使能 dot1x 端口控制，允许端口访问协商认证
Switch(config-if)# no shutdown	确定端口使能

Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface eth-0-26	进入接口配置模式
Switch(config-if)# no switchport	配置接口为路由端口
Switch(config-if)# ip address 202.38.100.1/24	在此接口上配置 IP 地址
Switch(config-if)# no shutdown	确定端口使能
Switch(config-if)# exit	退出接口配置模式
Switch(config)# radius-server host 202.38.100.7	为 RADIUS 服务器配置 IPv4 地址
Switch(config)# radius-server host 2001:1000::1	为 RADIUS 服务器配置 IPv6 地址
Switch(config)# radius-server key test	配置 RADIUS 服务器的共享密钥
Switch(config)# end	退出配置模式
Switch# show dot1x	验证管理 dot1x 的配置
Switch# show dot1x interface eth-0-25	验证在 eth-0-25 的 dot1x 的配置

采用强制授权模式，交换机配置步骤如下表所示。

Switch# configure terminal	进入全局配置模式
Switch(config)# dot1x system-auth-ctrl	全局使能 dot1x 身份验证控制
Switch(config)# interface eth-0-25	进入接口配置模式
Switch(config-if)# dot1x port-control force-authorized	在接口上使能 dot1x 端口控制，强制状态一直被授权
Switch(config-if)# no shutdown	确定端口使能
Switch(config-if)# end	退出配置模式
Switch# show dot1x	验证管理 dot1x 的配置
Switch# show dot1x interface eth-0-25	验证在 eth-0-25 的 dot1x 的配置

可选参数设置步骤如下表所示。

Switch# configure terminal	进入全局配置模式

Switch(config)# radius-server deadtime 10	设置重新激活 RADIUS 服务器的等待时间
Switch(config)# radius-server retransmit 5	设置 RADIUS 请求发送到服务器的最大可以失败的次数
Switch(config)# radius-server timeout 10	设置 RADIUS 服务器无响应的超时时间
Switch(config)# interface eth-0-25	进入接口配置模式
Switch(config-if)# dot1x max-req 5	未经授权之前，指定重新验证尝试的次数
Switch(config-if)# dot1x protocol-version 1	设置协议版本
Switch(config-if)# dot1x quiet-period 120	在 HELD 状态下的静默时间
Switch(config-if)# dot1x reauthentication	在一个端口上使能重新认证
Switch(config-if)# dot1x timeout re-authperiod 1800	指定重新认证的时间间隔
Switch(config-if)# dot1x timeout server-timeout 60	指定认证服务器响应超时时间
Switch(config-if)# dot1x timeout supp-timeout 60	指定客户端响应超时时间
Switch(config-if)# dot1x timeout tx-period 60	指定向客户端请求身份信息的时间间隔

服务器软件设置步骤及参数，详细配置信息参见图 6-2，图 6-3 和图 6-4。

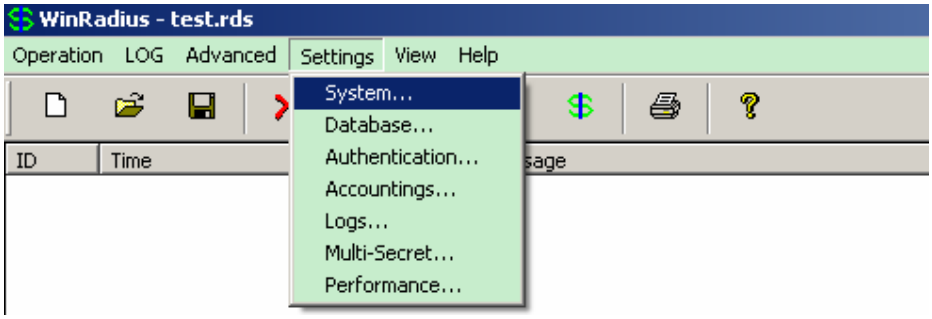


图6-2 选择 Setting-> System

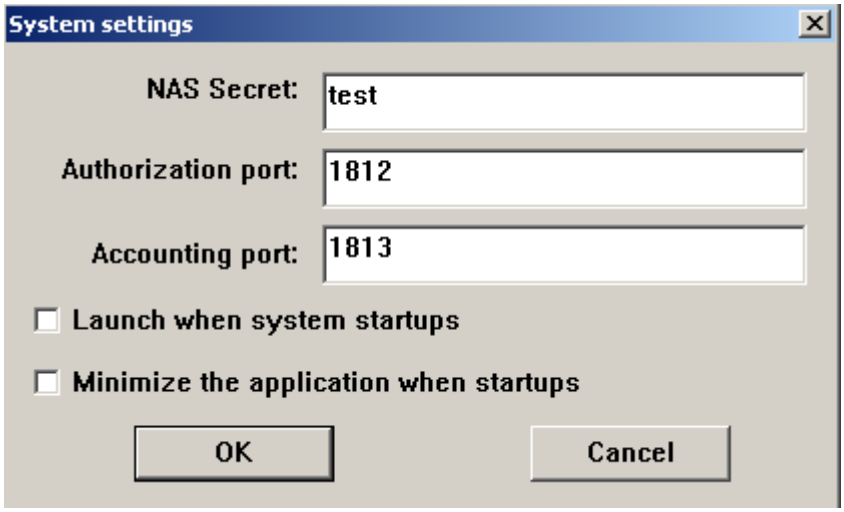


图6-3 配置 Radius 服务器的密码共享密钥、认证端口和计费端口

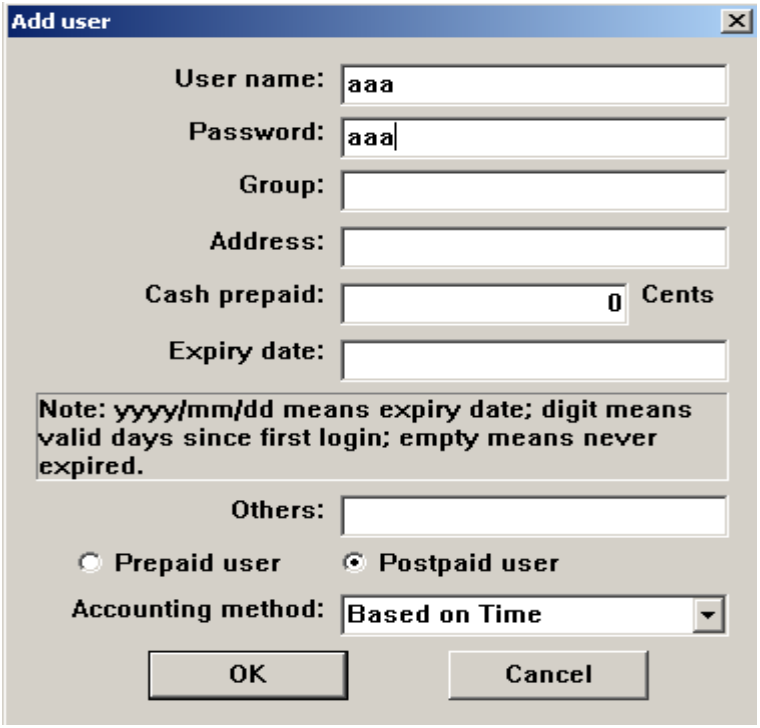


图6-4 在服务器端配置用户名和密码

6.7.4 命令验证

通过如下步骤，显示 dot1x 配置结果。

Switch# show dot1x

```
802.1X Port-Based Authentication Enabled
  RADIUS server address: 2001:1000::1:1812
  Next radius message ID: 0
  RADIUS server address: 202.38.100.7:1812
  Next radius message ID: 0
Switch# show dot1x interface eth-0-25
802.1X info for interface eth-0-25
  Supplicant name: aaa
  Supplicant address: 0011.11e1.9a3f
  portEnabled: true - portControl: Auto
  portStatus: Authorized - currentId: 42
  reAuthenticate: disabled
  reAuthPeriod: 3600
  abort:F fail:F start:F timeout:F success:T
  PAE: state: Authenticated - portMode: Auto
  PAE: reAuthCount: 0 - rxRespId: 0
  PAE: quietPeriod: 60 - reauthMax: 2 - txPeriod: 30
  BE: state: Idle - reqCount: 0 - idFromServer: 41
  BE: suppTimeout: 30 - serverTimeout: 30
  CD: adminControlledDirections: in - operControlledDirections: in
  CD: bridgeDetected: false
```

6.8 Guest VLAN 配置

6.8.1 简介

如果用户因为没有专用的认证客户端或者客户端版本过低等原因，导致无法认证成功，用户所在的端口会被加入 GuestVlan。GuestVlan 是一个不经认证也可以访问的 VLAN。在该 VLAN 内，用户可以进行例如客户端下载以及升级等操作。当用户利用这些资源，安装或者升级了认证客户端后，又可以进行正常的认证过程，从而访问其他的网络资源。开启 802.1x 特性、正确配置 GuestVlan 后，当设备从某一端口发送触发认证报文

（EAP-Request/Identity）超过设定的最大次数而没有收到客户端的任何回应报文后，该端口会被加入到 GuestVlan 内。此时用户发起认证，若认证失败，则端口仍然处于 guest vlan 中；如果认证成功，则端口返回到用户配置的 VLAN。

NOTE

Guest VLAN 的功能只能配置在 Access 端口上，不能作用于 3 层物理口(routed port)或者 trunk 端口上。

6.8.2 拓扑

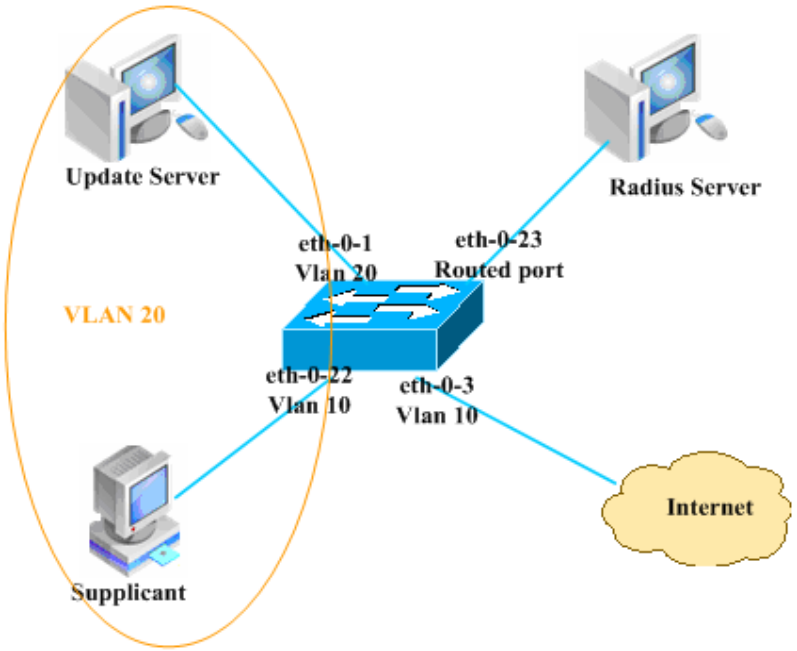


图6-5 Guest VLAN 拓扑

如图 6-5 所示，eth-0-22 是一个使能了 802.1x 功能的端口，它处于 VLAN 10 内。Update server 是用于客户端下载和升级的服务器，处于 VLAN 20 内。在 eth-0-22 上使能 guest vlan 特性，当设备从端口触发认证报文超过设定的最大次数而没有收到任何回

应报文后，端口被加入 guest VLAN 20 中。此时客户端和 update server 都在 VLAN 20 内，客户端可以访问 update server 并下载 802.1x 客户端。

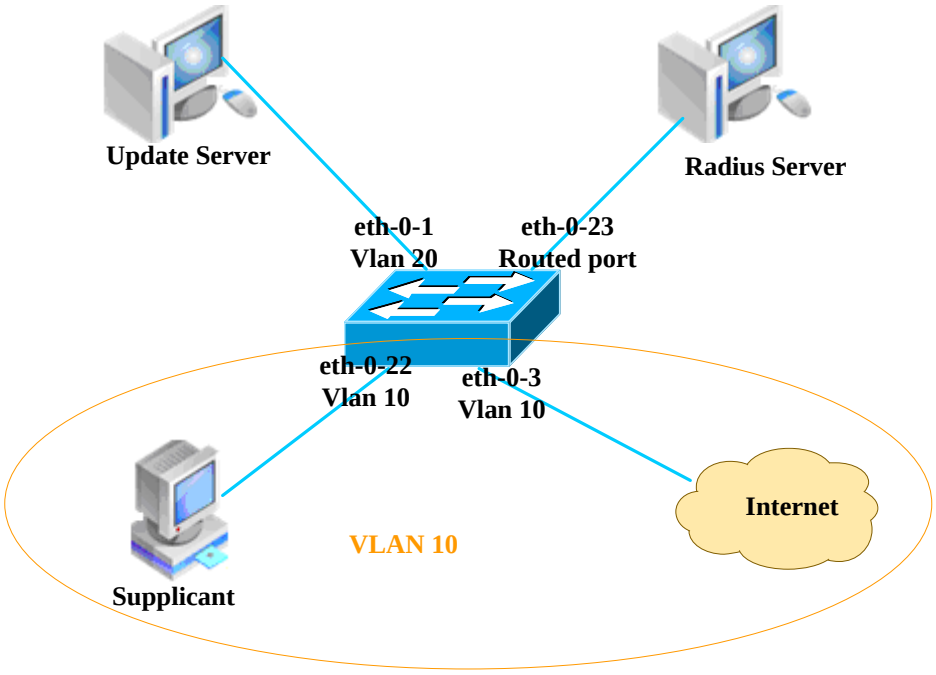


图6-6 认证成功之后的拓扑

连接认证服务器 radius server 的上行端口 eth-0-23 使一个 3 层物理口，它的 IP 地址为 202.38.100.1，radius server 的地址为 202.38.100.7。当认证成功之后，端口 eth-0-22 重新处于 VLAN 10 内，客户端可以访问 internet 了。

6.8.3 配置

配置交换机

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	进入 VLAN 配置模式
Switch(config-vlan)# vlan 10	创建 VLAN 10
Switch(config-vlan)# vlan 20	创建 VLAN 20
Switch(config-vlan)# exit	退出 VLAN 配置模式
Switch(config)# dot1x system-auth-ctrl	全局启用 dot1x 认证控制
Switch(config)# interface eth-0-22	指定要配置的接口，进入接口配置模式
Switch(config-if)# switchport mode access	设置接口为 access 模式

Switch(config-if)# switchport access vlan 10	设置接口允许 VLAN 10 通过
Switch(config-if)# dot1x port-control auto	在接口上使能 dot1x 端口控制，允许端口访问协商认证
Switch(config-if)# no shutdown	确定端口使能
Switch(config-if)# dot1x guest vlan 20	配置 guest VLAN 为 VLAN 20
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface eth-0-23	指定要配置的接口，进入接口配置模式
Switch(config-if)# no switchport	配置接口为路由端口
Switch(config-if)# ip address 202.38.100.1/24	在此接口上配置 IP 地址
Switch(config-if)# no shutdown	确定端口使能
Switch(config-if)# exit	退出接口配置模式
Switch(config)# radius-server host 202.38.100.7	为 RADIUS 服务器配置 IP 地址
Switch(config)# radius-server key test	配置 RADIUS 服务器的共享密钥
Switch(config)#end	退出配置模式
Switch # show dot1x	验证管理 dot1x 的配置
Switch # show dot1x interface eth-0-22	验证在 eth-0-22 的 dot1x 的配置

6.8.4 命令验证

步骤 1 在未配置 Guest VLAN 之前的初始状态如命令 **show running-config** 的屏显内容所示。

Switch# show running-config

```
dot1x system-auth-ctrl
radius-server host 202.38.100.7 key test
vlan database
 vlan 10,20
!
interface eth-0-22
 switchport access vlan 10
 dot1x port-control auto
 dot1x guest-vlan 20
!
interface eth-0-23
 no switchport
 ip address 202.38.100.1/24
```

```
!
Switch# show dOt1x interface eth-0-22
802.1X info for interface eth-0-22
  portEnabled: true - portControl: Auto
  portStatus: Unauthorized - currentId: 1
  reAuthenticate: disabled
  reAuthPeriod: 3600
  Guest VLAN:20
  abort:F fail:F start:F timeout:F success:F
  PAE: state: Connecting - portMode: Auto
  PAE: reAuthCount: 1 - rxRespId: 0
  PAE: quietPeriod: 60 - reauthMax: 2 - txPeriod: 30
  BE: state: Idle - reqCount: 0 - idFromServer: 19
  BE: suppTimeout: 30 - serverTimeout: 30
  CD: adminControlledDirections: in - operControlledDirections: in
  CD: bridgeDetected: false
Switch# show vlan brief
VLAN ID  Name            State   STP ID  DSCP    Member ports
                                                 (u)-Untagged, (t)-Tagged
======= ================ ======= ======= ======= ========================
1       default          ACTIVE  0       Disable eth-0-1(u) eth-0-2(u)
                                                 eth-0-3(u) eth-0-4(u)
                                                 eth-0-5(u) eth-0-6(u)
                                                 eth-0-7(u) eth-0-8(u)
                                                 eth-0-9(u) eth-0-10(u)
                                                 eth-0-11(u) eth-0-12(u)
                                                 eth-0-13(u) eth-0-14(u)
                                                 eth-0-15(u) eth-0-16(u)
                                                 eth-0-17(u) eth-0-18(u)
                                                 eth-0-19(u) eth-0-20(u)
                                                 eth-0-21(u) eth-0-24(u)
                                                 eth-0-25(u) eth-0-26(u)
                                                 eth-0-27(u) eth-0-28(u)
                                                 eth-0-29(u) eth-0-30(u)
                                                 eth-0-31(u) eth-0-32(u)
                                                 eth-0-33(u) eth-0-34(u)
                                                 eth-0-35(u) eth-0-36(u)
                                                 eth-0-37(u) eth-0-38(u)
                                                 eth-0-39(u) eth-0-40(u)
                                                 eth-0-41(u) eth-0-42(u)
                                                 eth-0-43(u) eth-0-44(u)
                                                 eth-0-45(u) eth-0-46(u)
                                                 eth-0-47(u) eth-0-48(u)
10      VLAN0010         ACTIVE  0       Disable eth-0-22(u)
20      VLAN0020         ACTIVE  0       Disable
```

步骤 2 Guest VLAN 客户端的状态信息如下面屏幕回显信息所示。

Switch# show dot1x interface eth-0-22

```
802.1X info for interface eth-0-22
  portEnabled: true - portControl: Auto
  portStatus: Unauthorized - currentId: 2
  reAuthenticate: disabled
```

```
  reAuthPeriod: 3600
  Guest VLAN:20(Port Authorized by guest vlan)
  abort:F fail:F start:F timeout:F success:F
  PAE: state: Connecting - portMode: Auto
  PAE: reAuthCount: 2 - rxRespId: 0
  PAE: quietPeriod: 60 - reauthMax: 2 - txPeriod: 30
  BE: state: Idle - reqCount: 0 - idFromServer: 19
  BE: suppTimeout: 30 - serverTimeout: 30
  CD: adminControlledDirections: in - operControlledDirections: in
  CD: bridgeDetected: false
Switch# show vlan brief
VLAN ID  Name            State   STP ID  DSCP    Member ports
                                                 (u)-Untagged, (t)-Tagged
======= ================ ======= ======= ======= ========================
1       default          ACTIVE  0       Disable eth-0-1(u) eth-0-2(u)
                                                 eth-0-3(u) eth-0-4(u)
                                                 eth-0-5(u) eth-0-6(u)
                                                 eth-0-7(u) eth-0-8(u)
                                                 eth-0-9(u) eth-0-10(u)
                                                 eth-0-11(u) eth-0-12(u)
                                                 eth-0-13(u) eth-0-14(u)
                                                 eth-0-15(u) eth-0-16(u)
                                                 eth-0-17(u) eth-0-18(u)
                                                 eth-0-19(u) eth-0-20(u)
                                                 eth-0-21(u) eth-0-24(u)
                                                 eth-0-25(u) eth-0-26(u)
                                                 eth-0-27(u) eth-0-28(u)
                                                 eth-0-29(u) eth-0-30(u)
                                                 eth-0-31(u) eth-0-32(u)
                                                 eth-0-33(u) eth-0-34(u)
                                                 eth-0-35(u) eth-0-36(u)
                                                 eth-0-37(u) eth-0-38(u)
                                                 eth-0-39(u) eth-0-40(u)
                                                 eth-0-41(u) eth-0-42(u)
                                                 eth-0-43(u) eth-0-44(u)
                                                 eth-0-45(u) eth-0-46(u)
                                                 eth-0-47(u) eth-0-48(u)
10      VLAN0010         ACTIVE  0       Disable
20      VLAN0020         ACTIVE  0       Disable eth-0-22(u)
```

步骤 3 Client is authenticated

Switch# show dot1x interface eth-0-22

```
802.1X info for interface eth-0-22
  Supplicant name: ychen
  Supplicant address: ae38.3288.f046
  portEnabled: true - portControl: Auto
  portStatus: Authorized - currentId: 6
  reAuthenticate: disabled
  reAuthPeriod: 3600
  Guest VLAN:20
  abort:F fail:F start:F timeout:F success:T
  PAE: state: Authenticated - portMode: Auto
```

```
  PAE: reAuthCount: 0 - rxRespId: 0
  PAE: quietPeriod: 60 - reauthMax: 2 - txPeriod: 30
  BE: state: Idle - reqCount: 0 - idFromServer: 5
  BE: suppTimeout: 30 - serverTimeout: 30
  CD: adminControlledDirections: in - operControlledDirections: in
  CD: bridgeDetected: false
Switch# show vlan brief
VLAN ID  Name            State   STP ID  DSCP    Member ports
                                                 (u)-Untagged, (t)-Tagged
======= ================ ======= ======= ======= ========================
1       default          ACTIVE  0       Disable eth-0-1(u) eth-0-2(u)
                                                 eth-0-3(u) eth-0-4(u)
                                                 eth-0-5(u) eth-0-6(u)
                                                 eth-0-7(u) eth-0-8(u)
                                                 eth-0-9(u) eth-0-10(u)
                                                 eth-0-11(u) eth-0-12(u)
                                                 eth-0-13(u) eth-0-14(u)
                                                 eth-0-15(u) eth-0-16(u)
                                                 eth-0-17(u) eth-0-18(u)
                                                 eth-0-19(u) eth-0-20(u)
                                                 eth-0-21(u) eth-0-24(u)
                                                 eth-0-25(u) eth-0-26(u)
                                                 eth-0-27(u) eth-0-28(u)
                                                 eth-0-29(u) eth-0-30(u)
                                                 eth-0-31(u) eth-0-32(u)
                                                 eth-0-33(u) eth-0-34(u)
                                                 eth-0-35(u) eth-0-36(u)
                                                 eth-0-37(u) eth-0-38(u)
                                                 eth-0-39(u) eth-0-40(u)
                                                 eth-0-41(u) eth-0-42(u)
                                                 eth-0-43(u) eth-0-44(u)
                                                 eth-0-45(u) eth-0-46(u)
                                                 eth-0-47(u) eth-0-48(u)
10      VLAN0010         ACTIVE  0       Disable eth-0-22(u)
20      VLAN0020         ACTIVE  0       Disable
```

Switch# show dot1x

```
802.1X Port-Based Authentication Enabled
  RADIUS server address: 202.38.100.7:1812
  Next radius message ID: 0
Switch# show dot1x statistics
=====================================
802.1X statistics for interface eth-0-22
  EAPOL Frames Rx: 52 - EAPOL Frames Tx: 4270
  EAPOL Start Frames Rx: 18 - EAPOL Logoff Frames Rx: 2
  EAP Rsp/Id Frames Rx: 29 - EAP Response Frames Rx: 3
  EAP Req/Id Frames Tx: 3196 - EAP Request Frames Tx: 3
  Invalid EAPOL Frames Rx: 0 - EAP Length Error Frames Rx: 0
  EAPOL Last Frame Version Rx: 2 - EAPOL Last Frame Src: ae38.3288.f046
```

6.9 ARP Inspection 配置

6.9.1 简介

缺省情况下，所有的 ARP 报文都将按照规则通过交换机。用户可以通过启用 ARP Inspection 功能监控 ARP 报文；该功能可以通过对 ARP 报文的有效性检查来过滤无效的 ARP 报文，也可以通过设置规则，让特定 ARP 报文通过，或者丢弃特定的 ARP 报文，以提高系统的安全性，并在一定程度上抑制 ARP 报文攻击。

ARP 检查是一个在网络中验证 ARP 报文的安全特性，可以检查日志、丢弃无效 IP 与 MAC 捆绑的 ARP 报文。这些能力可以保护网络免受人为攻击。ARP 检测确保只有有效的 ARP 请求和响应被执行，交换机执行的行为包括：在不信任端口上拦截所有 ARP 请求和响应。

在更新本地 ARP 缓存或者转发到特定目的地址的报文之前，需要验证每个检测的报文是否都是有效的。

丢弃无效 APR 报文：ARP 检测，是根据存在的 DHCP snooping 数据库中有效 IP 到 MAC 的绑定决定一个 ARP 报文的有效性。在信任端口上，交换机转发报文不需要任何检查，在不信任端口上，交换机只在有效情况下实现转发。

6.9.2 术语

下面简要介绍用于描述 ARP Inspection 相关的术语和概念。

DHCP Snooping

DHCP snooping 是一个在不可信主机和可信 DHCP 服务器之间执行防火墙的功能的安全特性，这个特性建立和维护 DHCP snooping 数据库，这个数据库包含租用 IP 地址的不信任主机信息。

Address Resolution Protocol (ARP)

ARP 通过映射 IP 地址和 MAC 地址，提供在二层广播域的 IP 通信。例如主机 B 想要发送信息到主机 A 上，但是没有主机 A 的 MAC 地址，主机 B 在广播域内产生一个广播报文对所有主机获取主机 A 的 MAC 地址。在广播域内的所有主机接收 ARP 请求，主机 A 返回它的 MAC 地址。

6.9.3 配置

创建 VLAN

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	配置 VLAN 数据库
Switch(config-vlan)# vlan 2	创建 VLAN 2

Switch(config-vlan)# exit	退出 VLAN 配置模式
Switch(config)# exit	退出全局配置模式

添加接口到 VLAN

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入端口配置模式，开始配置端口 eth-0-1
Switch(config-if)# switchport access vlan 2	添加端口到 VLAN2
Switch(config-if)# interface eth-0-2	开始配置端口 eth-0-2
Switch(config-if)# switchport access vlan 2	添加端口到 vlan 2
Switch(config-if)# interface eth-0-3	开始配置端口 eth-0-3
Switch(config-if)# switchport access vlan 2	添加端口到 VLAN 2
Switch(config-if)# interface eth-0-4	开始配置 eth-0-4
Switch(config-if)# switchport access vlan 2	添加端口到 VLAN 2
Switch(config-if)# exit	退出端口配置模式

配置 ARP 检查

Switch(config)# interface eth-0-1	进入端口配置模式，开始配置端口 eth-0-1
Switch(config-if)# ip arp inspection trust	配置端口为信任状态（通常把互联的交换机端口配置为可信任
Switch(config-if)# exit	退出端口配置模式
Switch(config)# ip arp inspection vlan 2	在 VLAN2 上使能 ARP 检查
Switch(config)# ip arp inspection vlan 2 logging acl-match matchlog	使能 VLAN2 上 ARP 检查匹配日志显示
Switch(config)# ip arp inspection validate src-mac ip dst-mac	在 ARP 报文中验证源 MAC 地址、IP、目的 MAC 地址

添加 ARP ACL

Switch(config)# arp access-list test	创建 ARP access-list of test
Switch(config-arp-acl)# deny request ip host 1.1.1.1 mac any	添加一个拒绝 1.1.1.1ARP 请求的 ACL 项目
Switch(config-arp-acl)# exit	退出 ARP ACL 配置模式
Switch(config)# ip arp inspection filter test vlan 2	在 VLAN2 上使能 ARP ACL
Switch(config)# exit	退出全局配置模式

6.9.4 命令验证

在交换机上检查 ARP 检测的配置是否正确，详细步骤参见如下描述。

Switch# show ip arp inspection

```
Source Mac Validation        : Enabled
Destination Mac Validation   : Enabled
IP Address Validation        : Enabled
Vlan     Configuration     ACL Match     Static ACL
================================================================
2        enabled             test
Vlan     ACL Logging     DHCP Logging
================================================================
2        deny              deny
Vlan     Forwarded      Dropped     DHCP Drops     ACL Drops
================================================================
2        0                0             0               0
Vlan     DHCP Permits     ACL Permits     Source MAC Failures
================================================================
2        0                  0                  0
Vlan   Dest MAC Failures   IP Validation Failures   Invalid Protocol Data
================================================================
2       0                      0                              0
```

在交换机上查看 ARP Inspection 的日志记录信息：

Switch# show ip arp inspection log

```
Total Log Buffer Size : 32
Syslog rate : 5 entries per 1 seconds.
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
```

```
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
1970-01-02 00:30:47 : Drop an ARP packet by ACL on vlan 2
```

6.10 DHCP Snooping 配置

6.10.1 简介

DHCP Snooping 是一种安全功能，如不受信任的主机和信任的 DHCP 服务器之间的防火墙行为，DHCP Snooping 功能执行如下：

- 验证 DHCP 消息接收来自不信任的源和过滤掉无效消息。
- 建立和维护 DHCP Snooping 绑定数据库，其中包含不信任主机租用的 IP 地址信息。
- 利用 DHCP Snooping 绑定数据库来验证来自不受信任的主机的后续请求。
- 其他的安全功能，如，动态 ARP 监测，也可以使用 DHCP Snooping 绑定数据库中存储的信息，每个 VLAN 的基础上启用 DHCP Snooping 功能，该功能在默认情况下在所有 VLAN 上都无效。你可以在一个单独的 VLAN 或者 VLAN 范围使能该功能，DHCP Snooping 功能在软件中实现，所有 DHCP 消息在芯片中被拦截直接发往 CPU 进行处理。

6.10.2 配置

配置 VLAN

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	配置 VLAN 数据库
Switch(config-vlan)# vlan 12	创建 VLAN 12
Switch(config-vlan)# exit	退出全局配置模式

配置接口 eth-0-12

Switch(config)# interface eth-0-12	进入接口配置模式
Switch(config-if)# switchport	设置为交换接口
Switch(config-if)# switchport access vlan 12	添加接口到 VLAN 12
Switch(config-if)# dhcp snooping trust	配置接口为信任状态

Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出全局配置模式

配置接口 eth-0-11

Switch(config)# interface eth-0-11	进入接口配置模式
Switch(config-if)# switchport	设置为交换接口
Switch(config-if)# switchport access vlan 12	添加接口到 VLAN 12
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出全局配置模式

配置 VLAN12 接口

Switch(config)# interface vlan 12	进入接口配置模式
Switch(config-if)# ip address 12.1.1.1/24	设置 VLAN 12 的 IP 地址
Switch(config-if)# exit	退出接口配置模式

配置 DHCP 特性

Switch(config)# dhcp snooping verify mac-address	检查 DHCP 用户上送的请求报文头 MAC 地址是否合法

使能 DHCP snooping 全局特性

Switch(config)# service dhcp enable	使能 dhcp 服务
Switch(config)# dhcp snooping	使能 dhcp snooping 特性
Switch(config)# dhcp snooping vlan 12	在 VLAN 12 上使能 dhcp snooping 特性

6.10.3 命令验证

步骤 1 根据如下步骤，检查接口配置是否正确。

Switch(config)# show running-config interface eth-0-12

```
!
interface eth-0-12
 dhcp snooping trust
 switchport access vlan 12
!
Switch(config)# show running-config interface eth-0-11
!
interface eth-0-11
 switchport access vlan 12
!
```

步骤 2 使用如下命令，检查 DHCP 服务状态。

Switch# show services

```
Networking services configuration:
Service Name         Status
============================================================
dhcp                   enable
```

步骤 3 使用如下命令，打印 dhcp snooping 配置，检查当前配置。

Switch# show dhcp snooping config

```
dhcp snooping service: enabled
dhcp snooping switch: enabled
Verification of hwaddr field: enabled
Insertion of relay agent information (option 82): disable
Relay agent information (option 82) on untrusted port: not allowed
dhcp snooping vlan 12
```

步骤 4 使用如下命令，检查 dhcp snooping 的统计信息。

Switch# show dhcp snooping statistics

```
DHCP snooping statistics:
============================================================
DHCP packets                               17
BOOTP packets                              0
Packets forwarded                         30
Packets invalid                            0
Packets MAC address verify failed      0
Packets dropped                            0
```

步骤 5 使用如下命令，显示 dhcp snooping 绑定信息。

Switch# show dhcp snooping binding all

```
DHCP snooping binding table:
VLAN MAC Address    Interface  Lease(s)   IP Address
```

```
=========================================================
12   0016.76a1.7ed9 eth-0-11   691190     12.1.1.65
```

6.11 IP Source Guard 配置

6.11.1 简介

通过 IP Source Guard 绑定功能，可以对端口转发的报文进行过滤控制，防止非法 IP 地址和 MAC 地址的报文通过端口，提高了端口的安全性。端口接收到报文后，通过查找 IP Source Guard 绑定表项，对报文进行如下处理：

步骤 1 对于 IP+Port 的绑定表项，如果报文中的源 IP 地址与绑定表项中记录的 IP 地址相同，端口将转发该报文；若不相同，则丢弃；

步骤 2 对于 IP+Port+MAC 的绑定表项，如果报文中的源 MAC 地址和源 IP 地址与绑定表项中记录的 MAC 地址和 IP 地址相同，端口将转发该报文；若不相同，则丢弃。

步骤 3 对于 IP+Port+MAC+VLAN 的绑定表项，如果报文中的源 MAC 地址，源 IP 地址和 VLAN 与绑定表项中记录的 MAC 地址，IP 地址和 VLAN 相同，端口将转发该报文；若不相同，则丢弃。

6.11.2 术语

以下是一些用来描述 IP source guard 的术语和概念的简要描述：

动态主机配置协议(DHCP)

动态主机配置协议 (DHCP) 是一个客户机/服务器的协议，它会自动提供 IP 地址以及其它相关的子网掩码和默认网关等信息给一个互联网协议（IP）的主机。

DHCP Snooping

DHCP Snooping 是一种安全功能，如不受信任的主机和信任的 DHCP 服务器之间的防火墙行为。此功能建立和维护 DHCP Snooping 绑定数据库，其中包含不可信主机租用的 IP 地址信息。

ACL

访问控制列表。

6.11.3 配置

配置 VLAN 信息

Switch# configure terminal	进入配置模式

Switch(config)# vlan database	进入 VLAN 模式
Switch(config-vlan)# vlan 3	创建 VLAN3
Switch(config-vlan)# exit	退出 VLAN 模式
Switch(config)# interface eth-0-16	进入接口模式
Switch(config-if)# switchport	设置端口为二层端口
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# switchport access vlan 3	设置端口属于 VLAN3
Switch(config-if)# exit	退出接口模式

配置 IP source guard

Switch(config)# ip source maximal binding number per-port 15	设置每个端口最大绑定的条目为 15 条
Switch(config)# ip source binding mac 1111.1111.1111 vlan 3 ip 10.0.0.2 interface eth-0-16	配置 IP Source Guard 绑定表项
Switch(config)# interface eth-0-16	进入接口模式
Switch(config-if)# ip verify source ip	在接口下使能 IP+Port 绑定检查
Switch(config-if)# exit	退出接口模式

删除配置

Switch(config)# no ip source binding mac 1111.1111.1111 vlan 3 ip 10.0.0.2 interface eth-0-16	删除单条 IP Source Guard 绑定表项
Switch(config)# no ip source binding entries interface eth-0-16	删除所有绑定到 eth-0-16 的表项
Switch(config)# no ip source binding entries vlan 3	删除所有绑定到 VLAN 3 的表项
Switch(config)# no ip source binding entries	清除所有的绑定表项

6.11.4 命令验证

Switch#show running-config interface eth-0-16

```
!
interface eth-0-16
 ip verify source ip
 switchport access vlan 3
```

6.12 私有 VLan 配置

6.12.1 简介

私有 vlan 属性在同一 vlan 内部实现的二层流量的隔离和互通。

可根据需要，提供灵活的组网方法。

6.12.2 拓扑

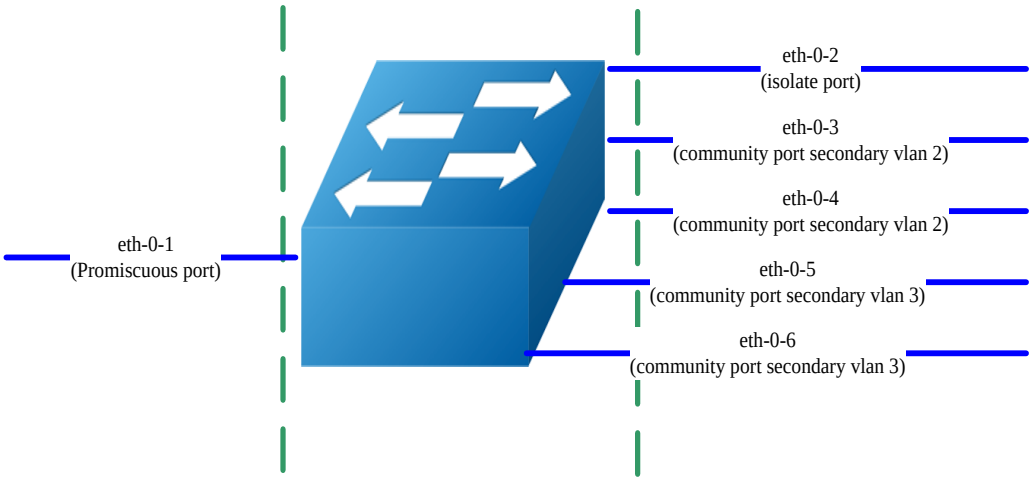


图6-7 私有 vlan 基本拓扑

所有端口在同一私有 vlan 中。

端口 1 是混杂端口，可与同一私有 vlan 中所有其他端口互通

端口 2 是隔离端口，它与同一私有 vlan 中所有其他端口都互相隔离，除了混杂端口（端口 1）

端口 3 和 4 是互通端口，属于子 vlan 2，端口 3 和 4 彼此可以互通，还可以和混杂端口互通。和同一私有 vlan 的其他端口都互相隔离。

端口 5 和 6 是互通端口，属于子 vlan 3，端口 5 和 6 彼此可以互通，还可以和混杂端口互通。和同一私有 vlan 的其他端口都互相隔离。

6.12.3 配置

交换机配置.

Switch# configure terminal	进入配置模式
Switch (config)# vlan database	进入 vlan 模式
Switch (config-vlan)# vlan 2	创建 vlan 2
Switch (config-vlan)# quit	退出 vlan 模式
Switch (config)# interface eth-0-1	进入接口模式
Switch (config-if)# switchport mode private-vlan promiscuous	配置私有 vlan 模式为混杂端口
Switch (config-if)# switchport private-vlan 2	配置主 vlan 2
Switch (config-if)# quit	退出接口模式
Switch (config)# interface eth-0-2	进入接口模式
Switch (config-if)# switchport mode private-vlan host	配置私有 vlan 模式为主机端口
Switch (config-if)# switchport private-vlan 2 isolate	配置私有 vlan 模式为隔离端口，配置主 vlan 为 2
Switch (config-if)# quit	退出接口模式
Switch (config)# interface eth-0-3	进入接口模式
Switch (config-if)# switchport mode private-vlan host	配置私有 vlan 模式为主机端口
Switch (config-if)# switchport private-vlan 2 community-vlan 2	配置私有 vlan 模式为互通端口，配置主 vlan 为 2， 子 vlan 为 2
Switch (config-if)# quit	退出接口模式
Switch (config)# interface eth-0-4	进入接口模式
Switch (config-if)# switchport mode private-vlan host	配置私有 vlan 模式为主机端口
Switch (config-if)# switchport private-vlan 2 community-vlan 2	配置私有 vlan 模式为隔离端口，配置主 vlan 为 2
Switch (config-if)# quit	退出接口模式
Switch (config)# interface eth-0-5	进入接口模式
Switch (config-if)# switchport mode private-vlan host	配置私有 vlan 模式为主机端口

Switch (config-if)# switchport private-vlan 2 community-vlan 3	配置私有 vlan 模式为隔离端口，配置主 vlan 为 3
Switch (config-if)# quit	退出接口模式
Switch (config)# interface eth-0-6	进入接口模式
Switch (config-if)# switchport mode private-vlan host	配置私有 vlan 模式为主机端口
Switch (config-if)# switchport private-vlan 2 community-vlan 3	配置私有 vlan 模式为隔离端口，配置主 vlan 为 3
Switch (config-if)# quit	退出接口模式

6.12.4 命令验证

显示结果如下：

switch # show private-vlan

Primary Secondary Type Ports

```
------------------------------------------------------------------------------
2         N/A       promiscuous    eth-0-1
2         N/A       isloate        eth-0-2
2         2         community      eth-0-3   eth-0-4
2         3         community      eth-0-5   eth-0-6
```

6.13 AAA 配置

6.13.1 简介

系统可以使用 AAA 认证的方法去验证访问网络和网络服务的用户。RADIUS 认证是 AAA 认证方法之一。RADIUS 是防止未经授权的访问，确保网络安全的分布式客户机/服务器系统。RADIUS 为网络环境中广泛使用的协议。它通常用于嵌入式网络设备如路由器，调制解调器服务器，交换机等。RADIUS 客户端通常在支持 RADIUS 的路由器和交换机上运行。客户端发送认证请求到 RADIUS 服务器，RADIUS 服务器包含所有的用户认证和网络服务访问信息。

6.13.2 拓扑

下图为 RADIUS 的网络拓扑。

一台 PC 机作为 RADIUS 服务器， 配置网卡 1.1.1.2/24。

设置 Switch 的 Eth-0-23 接口的 IP 地址为 1.1.1.1/24。 配置交换机的管理口 IP 地址为 10.10.29.215,连接交换机管理口的 PC 机 IP 地址为 10.10.29.10。

图6-8 RADIUS 拓扑图

6.13.3 配置

配置 AAA

Switch# configure terminal	进入全局配置模式
Switch(config)# aaa new-model	启用 AAA 协议
Switch(config)# aaa authentication login radius-login radius local	设置 AAA 验证的模式
Switch(config)# radius-server host 1.1.1.2 auth-port 1819 key keyname	配置 RADIUS 服务器参数
Switch(config)# radius-server host 2001:1000::1 auth-port 1819 key keyname	(可选)配置 RADIUS 服务器参数
Switch(config)# interface eth-0-23	进入接口模式
Switch(config-if)# no switchport	设置端口为三层端口
Switch(config-if)# ip address 1.1.1.1/24	配置 IP 地址
Switch(config-if)# quit	退出接口模式
Switch(config)# line vty 0 7	进入 VTY 模式
Switch(config-line)#login authentication radius-login Switch(config-line)#privilege level 4 Switch(config-line)#no line-password	配置验证方式

配置 PC 及 WinRADIUS

步骤 1 配置 IP 地址，详细参见图 6-9 所示。

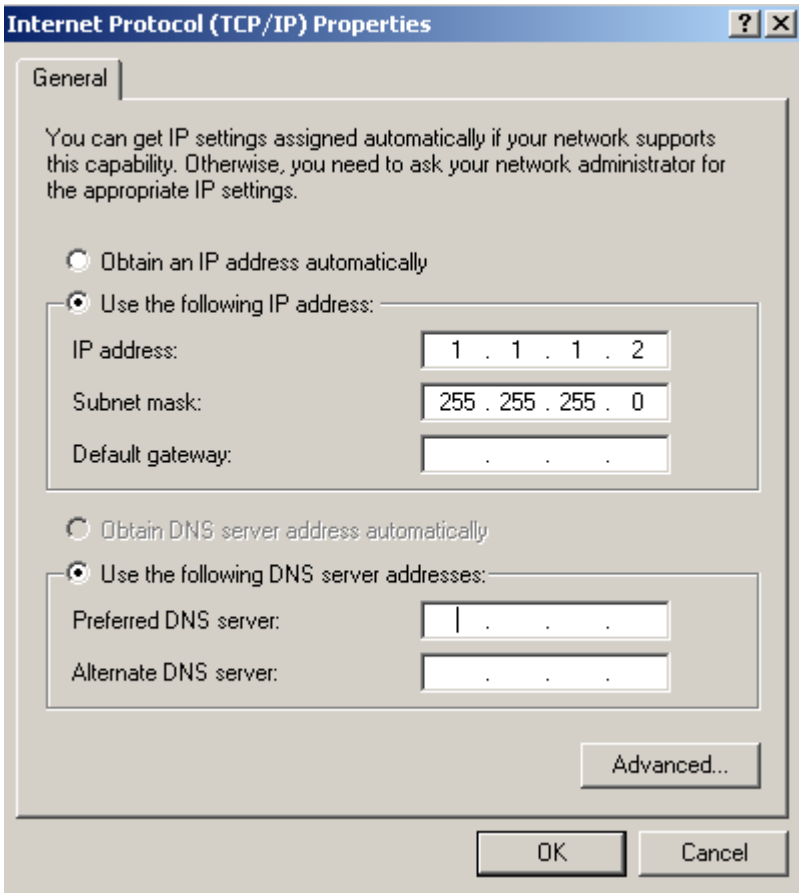


图6-9 配置 IP 地址

步骤 2 测试客户机和服务器之间的连通性，详细如图 6-10 所示。

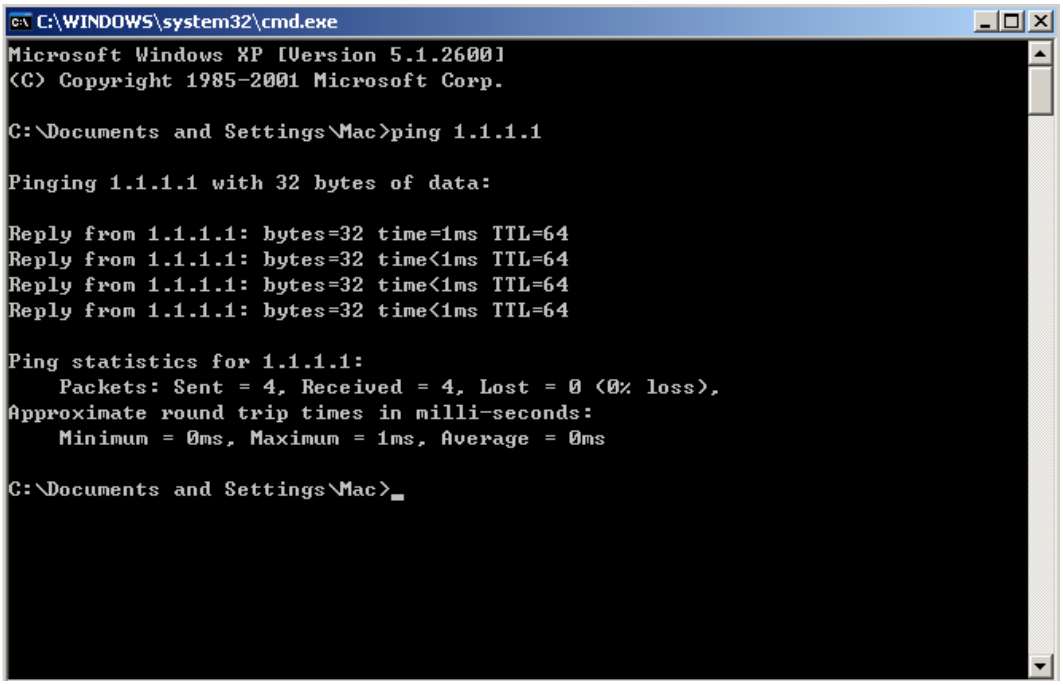


图6-10 连通性检查结果

步骤 3 打开服务器上的软件，详细如图 6-11 所示。

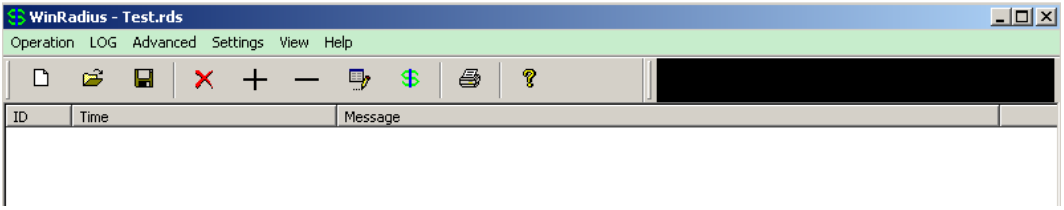


图6-11 打开服务器软件

步骤 4 按照图 6-12 所示的操作，进行系统设置。

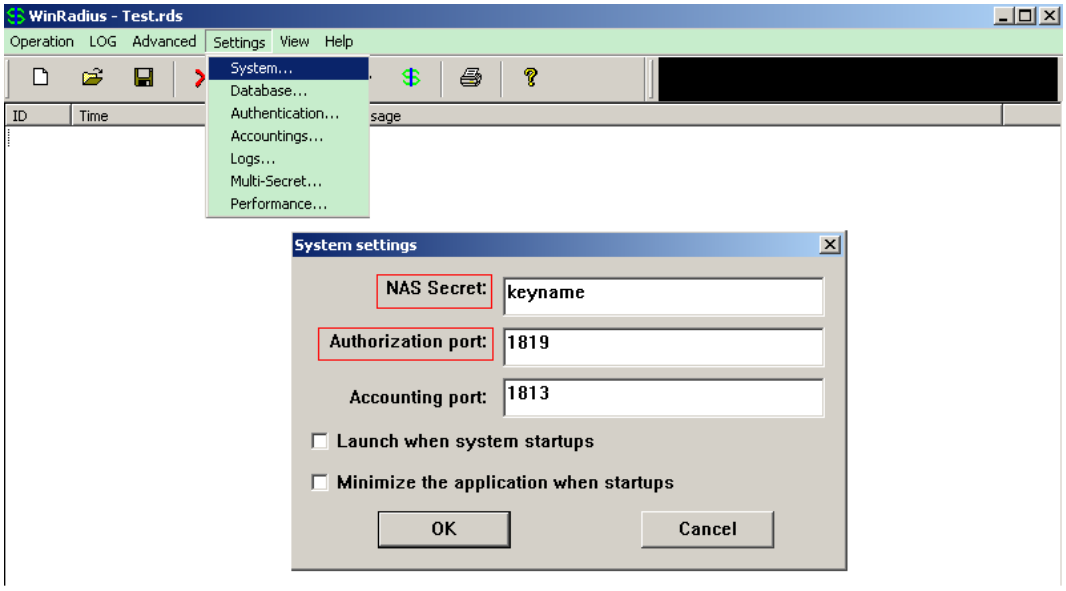


图6-12 系统配置

步骤 5 加入用户名和密码的详细步骤和参数类似图 6-13 所示。

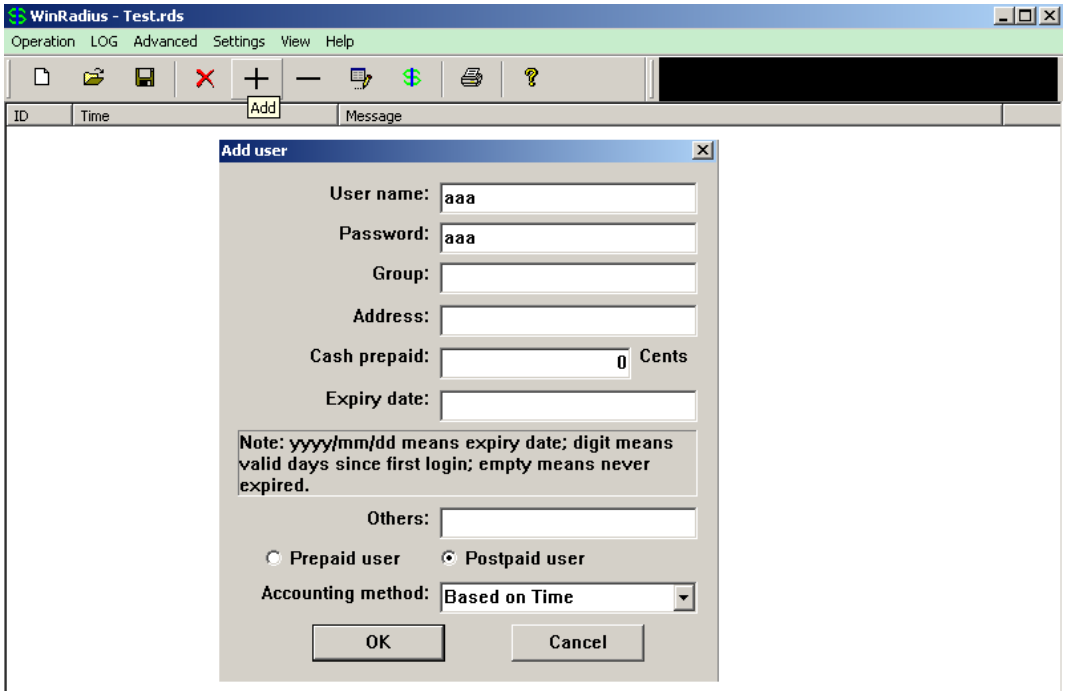


图6-13 加入用户名和密码

步骤 6 使用 **ping** 命令检查连通情况，详细类似图 6-14 所示。

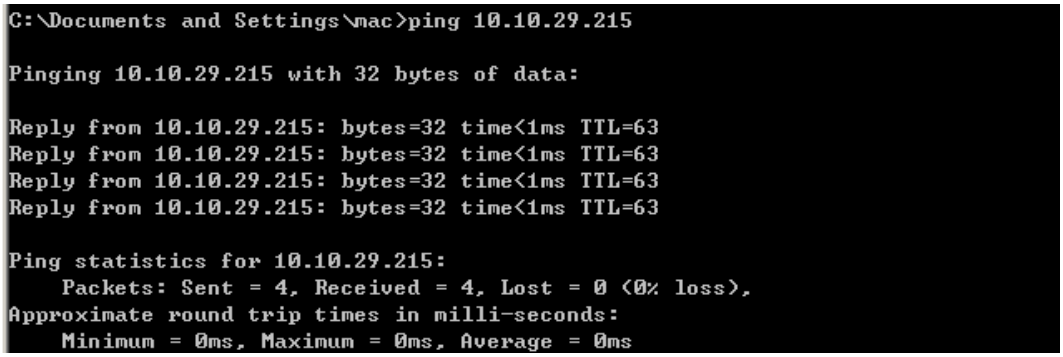


图6-14 Ping 命令检查结果

6.13.4 命令验证

使用 show 命令检查交换机的工作状态。

Switch# show aaa status

```
aaa stats:
     Authentication enable
```

可以使用 show 命令来显示交换机中的关键信息。

Switch# show aaa method-lists authentication

```
authen queue=AAA_ML_AUTHEN_LOGIN
    Name = default  state = ALIVE :   local
Name = radius-login  state = ALIVE :   radius  local
```

6.13.5 显示结果

进行 Telnet 测试，如配置正确，则 Telnet 连接的结果信息类似图 6-15 所示。

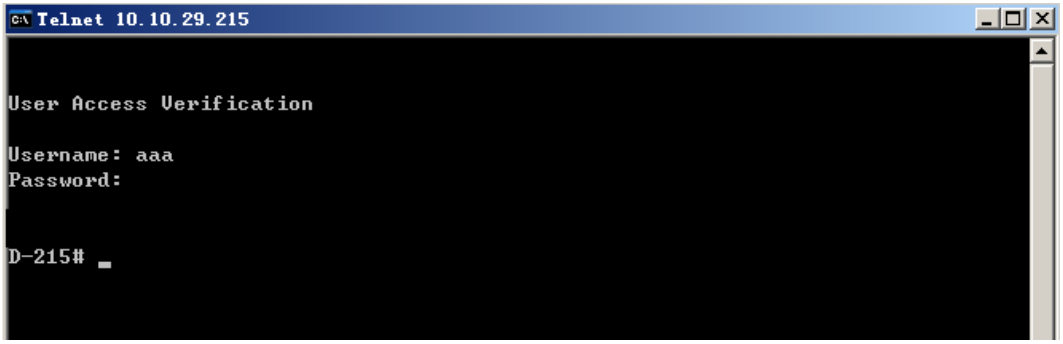


图6-15 Telnet 连接测试

NOTE

不要忘记打开 RADIUS 验证功能。

确认线缆连接的正确性。

若交换机无法进行 RADIUS 认证，可以使用命令检查系统日志信息：

Switch# show logging buffer

6.14 TACACS+配置

6.14.1 简介

系统可以使用 AAA 认证的方法去验证访问网络和网络服务的用户。TACACS+认证是 AAA 认证方法之一。TACACS+是防止未经授权的访问，确保网络安全的分布式客户机/服务器系统。TACACS+为网络环境中广泛使用的协议。它通常用于嵌入式网络设备如路由器，调制解调器服务器，交换机等支持 TACACS+的路由器和交换机上运行的客户。客户端发送认证请求到 TACACS+服务器，TACACS+服务器包含所有的用户认证和网络服务访问信息。

6.14.2 拓扑

下图是 TACACS+的网络拓扑。一台 PC 机作为 TACACS+服务器，配置网卡 1.1.1.2/24。设置 Switch 的 eth-0-23 接口的 IP 地址为 1.1.1.1/24。配置交换机的管理口 IP 地址为 10.10.29.215，连接交换机管理口（仅限带内管理口）的 PC 机 IP 地址为 10.10.29.10。

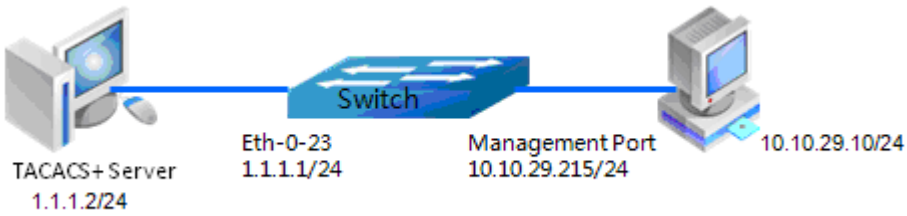


图6-16 TACACS+拓扑图

6.14.3 配置

配置 AAA 和 TACACS+

Switch# configure terminal	进入全局配置模式
Switch(config)# aaa new-model	启用 AAA 协议
Switch(config)# aaa authentication login tac-login tacacs-plus local	设置 AAA 验证的模式
Switch(config)# aaa authorization exec default tacacs-plus	设置 AAA 授权模式
Switch(config)# aaa accounting exec default start-stop tacacs-plus	设置 AAA EXEC 计费

Switch(config)# aaa accounting commands default tacacs-plus	设置 AAA 命令行计费
Switch(config)# tacacs-server host 1.1.1.2 port 123 key keyname	设置 TACACS+服务器的 IP 地址,验证端口和密码
Switch(config)# interface eth-0-23	进入接口模式
Switch(config-if)# no switchport	设置端口为三层端口
Switch(config-if)# ip address 1.1.1.1/24	配置 IP 地址
Switch(config-if)# quit	退出接口模式
Switch(config)# line vty 0 7	进入 VTY 模式
Switch(config-line)#login authentication tac-login Switch(config-line)#privilege level 4 Switch(config-line)#no line-password	配置验证方式

6.14.4 配置 TACACS + 服务器

步骤 1 下载 TACACS+服务器代码， DEVEL.201105261843.tar.bz2。

步骤 2 编译 TACACS+服务器代码。

步骤 3 修改配置文件，增加用户名和密码。

```
#!../obj.linux-2.6.9-89.29.1.elsmp-x86_64/tac_plus
id = spawnd {
        listen = { port = 49 }
        spawn = {
                instances min = 1
                instances max = 10
        }
        background = no
}
        user = aaa {
                password = clear bbb
                member = guest
        }
```

步骤 4 运行 TACACS+服务器程序。

[disciple: ~]$ ./tac_plus ./tac_plus.cfg.in -d 1

步骤 5 使用 Ping 命令检查连通结果。

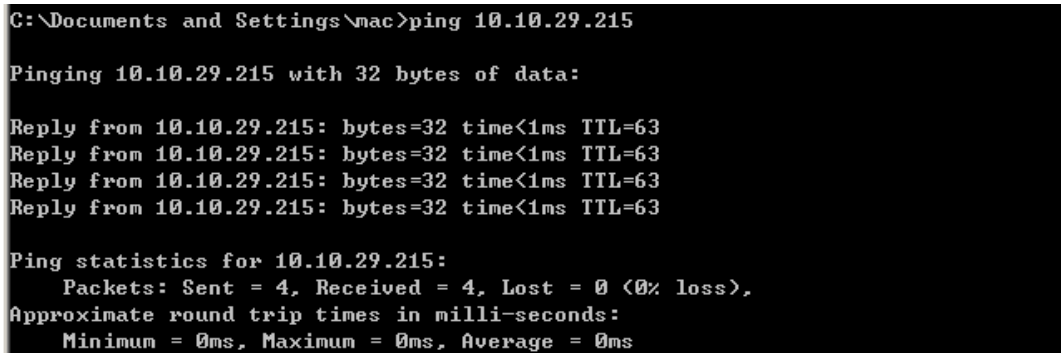


图6-17 连通结果

6.14.5 命令验证

使用 show authentication status 命令检查配置。

Switch# show aaa status

```
aaa stats:
     Authentication enable
```

使用 show aaa method-lists authentication 命令检查 AAA 配置。

```
Switch# show aaa method-lists authentication
authen queue=AAA_ML_AUTHEN_LOGIN
    Name = default  state = ALIVE :   local
Name = tac-login  state = ALIVE :   tacacs-plus local
```

6.14.6 显示结果

进行 Telnet 测试：如配置正确，则 Telnet 连接的结果信息类似图 6-18 所示。

图6-18 Telnet 测试结果

6.15 Port-Isolate 配置

6.15.1 简介

通过 Port-Isolated 端口隔离特性，可以实现不同用户的端口属于同一个 VLAN，但是不同端口之间不能互通。从而增强了网络的安全性，提供了灵活的组网方案，同时节省了大量的 VLAN 资源。

6.15.2 拓扑

图 6-19 显示了基本的端口隔离拓扑。

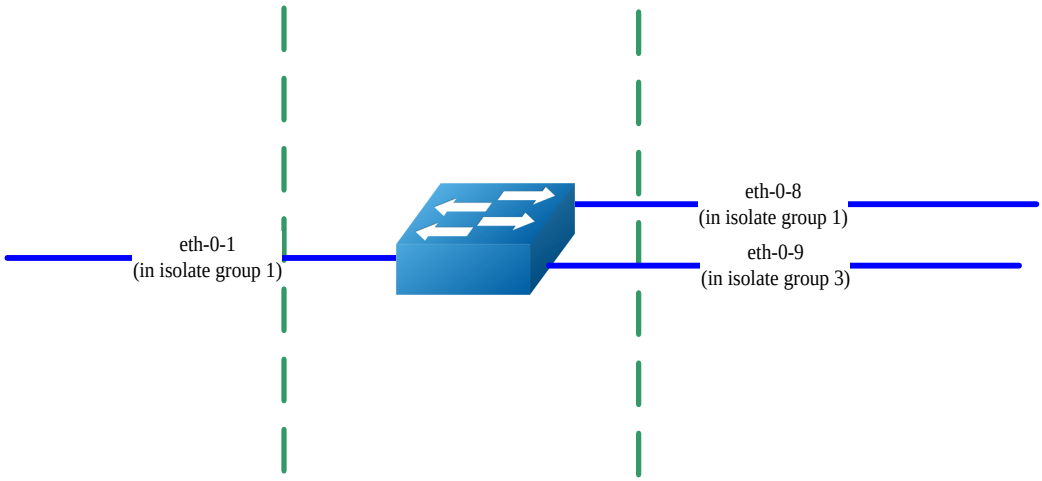


图6-19 Basic topology for port-isolate

端口 1 和端口 8 在同一个隔离组 1,所以端口 1 和端口 8 不能互相通信。

端口 9 在隔离组 3.所以端口 9 能和端口 1，8 互相通信。

6.15.3 配置

交换机配置

Switch# configure terminal	进入配置模式
Switch(config)# port-isolate mode l2	设置端口隔离的模式
Switch(config-if)# interface eth-0-1	进入接口模式
Switch(config-if)# port-isolate group 1	配置端口属于隔离组 1
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-8	进入接口模式
Switch(config-if)# port-isolate group 1	配置端口属于隔离组 1
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# port-isolate group 3	配置端口属于隔离组 3
Switch(config-if)# exit	退出接口模式
Switch(config)# end	退出配置模式
Switch# show port-isolate	显示端口隔离的配置

6.15.4 命令验证

使用如下命令显示端口隔离的配置。

```
switch# show port-isolate
------------------------------------------------------------------
 Port Isolate Groups:
------------------------------------------------------------------
 Groups ID: 1
eth-0-1, eth-0-8
------------------------------------------------------------------
Groups ID: 3
eth-0-9
------------------------------------------------------------------
```

6.16 DDoS 防御配置

6.16.1 简介

DDoS 攻击全名是 distributed denial-of-service attack，分布式拒绝攻击，是由 DoS(denial-of-service)攻击发展而来，攻击原理是利用合理的服务请求来占用过多的服务资源，从而使服务器无法处理合法用户的指令。DDoS 攻击利用处于不同位置的多个攻击者同时向一个或者数个目标发起攻击，或者一个或多个攻击者控制了位于不同位置的多台机器(傀儡机)并利用这些机器对受害者同时实施攻击。 DDoS 攻击将造成网络资源浪费、链路带宽堵塞、服务器资源耗尽而业务中断。

DDoS 防御特性可以保护我们交换机抵挡以下类型的攻击，可以拦截该种类型攻击对应的数据包：

- ICMP 泛洪：该攻击通过向目标 IP 发送大量 ICMP 包，占用带宽，从而导致合法报文无法达到目的地，达到攻击目的。
- Smurf 攻击：攻击者先使用受害主机的地址，向一个广播地址发送 ICMP 回响请求，在此广播网络上，潜在的计算机会做出响应，大量响应将发送到受害主机，此攻击后果同 ICMP 泛洪，但比之更为隐秘。
- SYN 泛洪：蓄意侵入 tcp 三次握手并打开大量的 TCP/IP 连接而进行的攻击，该攻击利用 IP 欺骗，向受害者的系统发送看起来合法的 SYN 请求，而事实上该源地址不存在或当时不在线，因而回应的 ACK 消息无法到达目的，而受害者的系统被大量的这种半开连接充满，资源耗尽，而合法的连接无法被响应。
- UDP 泛洪：该攻击通过向目标 IP 发送大量 UDP 包，占用带宽，消耗资源。
- Fraggle 攻击：该攻击是 smurf 的变种，针对防火墙对 ICMP 包检查比较严格的前提下，不再向广播地址发 ICMP 请求包，而是改为发送 UDP 包。
- Small-packet 攻击：IP 小报文攻击是发送大量的小报文到被攻击系统来消耗系统的资源。
- bad mac intercept ：目的 MAC 地址等于源 MAC 地址的报文攻击。
- bad ip equal：目的 IP 地址等于源 IP 地址的报文攻击。

6.16.2 配置

配置抵御 ICMP 泛洪攻击

Switch# configure terminal	进入全局配置模式
Switch(config)# ip icmp intercept maxcount 100	使能 ICMP 泛洪检测，设置每秒接收 ICMP 报文个数最大为 100
Switch(config)# end	退出全局配置模式
Switch# show ip-intercept config	显示当前 DDoS 防御配置

配置抵御 UDP 泛洪攻击

Switch# configure terminal	进入全局配置模式
Switch(config)# ip udp intercept maxcount 100	使能 UDP 泛洪检测，设置每秒接收 UDP 报文个数最大为 100
Switch(config)# end	退出全局配置模式
Switch# show ip-intercept config	显示当前 DDoS 防御配置

配置抵御 Smurf 攻击

Switch# configure terminal	进入全局配置模式
Switch(config)# ip smurf intercept	使能 smurf 攻击检测
Switch(config)# end	退出全局配置模式
Switch# show ip-intercept config	显示当前 DDoS 防御配置

配置抵御 SYN 泛洪攻击

Switch# configure terminal	进入全局配置模式
Switch(config)# ip tcp intercept maxcount 100	使能 SYN 泛洪检测，设置每秒接收 TCP 的 SYN 报文个数最大为 100
Switch(config)# end	退出全局配置模式
Switch# show ip-intercept config	显示当前 DDoS 防御配置

配置抵御 Fraggle 攻击

Switch# configure terminal	进入全局配置模式
Switch(config)# ip fraggle intercept	使能 Fraggle 攻击
Switch(config)# end	退出全局配置模式
Switch# show ip-intercept config	显示当前 DDoS 防御配置

配置抵御 Small-packet 攻击

Switch# configure terminal	进入全局配置模式
Switch(config)# ip small-packet intercept maxlength 32	使能 Small-packet 攻击检测，设置接收 IP 报文长度最小为 32 字节。
Switch(config)# end	退出全局配置模式
Switch# show ip-intercept config	显示当前 DDoS 防御配置

配置过滤相同 IP 报文

Switch# configure terminal	进入全局配置模式
Switch(config)# ip ipeq intercept	使能检测源 IP 地址等于目的 IP 地址的报文攻击
Switch(config)# end	退出全局配置模式
Switch# show ip-intercept config	显示当前 DDoS 防御配置

配置过滤相同 MAC 报文

Switch# configure terminal	进入全局配置模式
Switch(config)# ip maceq intercept	使能检测源 MAC 地址等于目的 MAC 地址的报文攻击
Switch(config)# end	退出全局配置模式
Switch# show ip-intercept config	显示当前 DDoS 防御配置

6.16.3 命令验证

Switch# show ip-intercept config

```
Current DDoS Prevent configuration:
============================================================
ICMP Flood Intercept             :Enable  Maxcount:100
UDP Flood Intercept              :Enable  Maxcount:100
SYN Flood Intercept              :Enable  Maxcount:100
Small-packet Attack Intercept    :Enable  Packet Length:32
Sumrf Attack Intercept           :Enable
Fraggle Attack Intercept         :Disable
MAC Equal Intercept              :Enable
```

```
IP Equal Intercept               :Enable
Switch# show ip-intercept statistics
Current DDoS Prevent statistics:
============================================================
Resist Small-packet Attack packets number    :  65
Resist ICMP Flood packets number             :  0
Resist Smurf Attack packets number           :  0
Resist SYN Flood packets number              :  0
Resist UDP Flood packets number              :  0
```

6.17 Key Chain 配置

6.17.1 简介

密钥链是一种通用的认证方法，适用于需要共享密钥的实体在建立相互信任之前交换密钥完成认证。这种认证方法通常被用在路由协议和网络应用中，可以增强对等体之间通信的安全性。

密钥链提供了一种包含密钥控制和基于生命周期的转滚法的安全机制，它将一连串的密钥通过生命周期联系在一起，并将它们按照序号挂在密钥链里。密钥链在使用时会依次比对链中的各个密钥，找到密钥则通过验证。

为了发挥生命周期的作用，在使用密钥链之前，必须要定义密钥的有效时间，并且为了保持稳定性，最好能同时使用一个以上的有效密钥。

6.17.2 配置

配置密钥链

Switch# configure terminal	进入全局配置模式
Switch(config)# key chain test	创建名为 test 的密钥链，并且进入密钥链配置模式
Switch(config-keychain)# key 1	创建 ID 为 1 的密钥，并且进入密钥配置模式
Switch(config-keychain-key)# key-string ##test_keystring_1##	配置密钥字符串
Switch(config-keychain-key)# accept-lifetime 0:0:1 1 jan 2012 infinite	配置密钥的合法接收时间
Switch(config-keychain)# key 2	创建 ID 为 2 的密钥，并且进入密钥配置模式
Switch(config-keychain-key)# key-string ##test_keystring_2##	配置密钥字符串

Switch(config-keychain-key)# send-lifetime 0:0:1 2 jan 2012 infinite	配置密钥的合法发送时间

6.17.3 命令验证

在特权模式下使用命令 **show key chain**，显示 key chain 的配置。

Switch # show key chain

```
key chain test:
    key 1 -- text "key-string ##test_keystring_1##"
      accept-lifetime <00:00:01 Jan 01 2012> - <infinite>
      send-lifetime <always valid> - <always valid> [valid now]
    key 2 -- text "key-string ##test_keystring_2##"
      accept-lifetime <always valid> - <always valid> [valid now]
      send-lifetime <00:00:01 Jan 02 2012> - <infinite>
```

6.18 Port-Block 配置

6.18.1 简介

默认情况下，端口泛洪报文都是没有目的 MAC 地址的。如果这些报文被送到保护端口上，将有可能出现安全问题。为了避免目的 MAC 地址未知或者已知的单播或组播传输到其他端口，可以阻塞该端口以避免发送单播或者组播出去。

6.18.2 配置

配置密钥链

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入端口配置模式
Switch(config-if)# port-block unknown-unicast	对 MAC 地址未知的单播阻塞
Switch(config-if)# end	退出全局配置模式
Switch# show port-block interface eth-0-1	显示端口的 port-block 配置信息

6.18.3 命令验证

在端口下配置 port-block，并 show 出配置信息：

```
Switch # show port-block interface eth-0-1
```

```
Known unicast blocked: Enabled
Known multicast blocked: Disabled
Unknown unicast blocked: Disabled
Unknown multicast blocked: Disabled
Broadcast blocked: Disabled
```

7 IP 业务配置指导

7.1 ARP 配置

7.1.1 简介

ARP（Address Resolution Protocol，地址解析协议）用于将网络层的 IP 地址解析为数据链路层的物理地址（MAC 地址）。

ARP 缓存 IP 和 MAC 地址的映射。当一个接口请求的地址映射不在缓存中，则设备将会缓存接收到的报文并在相应的子网内广播一个地址请求，如果获得响应，则生成新的地址映射并且发送缓存的报文。ARP 在等待地址映射回应消息的时候最多缓存一个报文，而且只有最近传输的报文才会被保存。如果目的主机在 3 次请求后都无法响应，则主机被认为故障，同时相应的错误消息将被返回。如果目的主机在一段时间内（通常为一小时）不发送消息，主机被认为可能出现问题，在删除 ARP 表项之前几个请求（一般为 6 个，3 个是单播和 3 个是广播）将被发送到主机上。

ARP 表项可以通过手工添加、删除、修改。手工添加的表项是永久的。

7.1.2 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入端口配置模式
Switch(config-if)# no switchport	配置端口到三层接口
Switch(config-if)# ip address 11.11.11.1/24	配置端口的 IP 地址
Switch(config-if)# arp timeout 1200	设置老化时间
Switch(config-if)# arp retry-interval 2	设置请求重发延迟时间
Switch(config)# arp 11.11.11.2 1a.a011.eca2	添加静态 ARP 条目

7.1.3 命令验证

验证 ARP 条目

Switch# show ip arp

```
 Protocol    Address          Age (min)  Hardware Addr   Interface
 Internet    11.11.11.2             -   001a.a011.eca2     eth-0-1
```

Switch# show ip arp summary

```
 1 IP ARP entries, with 0 of them incomplete
(Static:0,  Dyamic:0, Interface:1)
 ARP Pkt Received is: 0
 ARP Pkt Send number is: 0
 ARP Pkt Dicard number is: 0
```

验证 ARP 请求重发延迟和老化时间

Switch# show interface eth-0-1

```
Interface eth-0-1
  Interface current state: Administratively DOWN
  Hardware is Ethernet, address is 6c02.530c.2300 (bia 6c02.530c.2300)
  Bandwidth 1000000 kbits
  Index 1 , Metric 1 , Encapsulation ARPA
  Speed - Auto , Duplex - Auto  , Media type is 1000BASE_T
  Link speed type is autonegotiation, Link duplex type is autonegotiation
  Input flow-control is off, output flow-control is off
  The Maximum Frame Size is 1534 bytes
  VRF binding: not bound
  Label switching is disabled
  No virtual circuit configured
  VRRP master of :  VRRP is not configured on this interface
  ARP timeout 00:20:00,  ARP retry interval 2s
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes
    Received 0 unicast, 0 broadcast, 0 multicast
    0 runts, 0 giants, 0 input errors, 0 CRC
    0 frame, 0 overrun, 0 pause input
    0 input packets with dribble condition detected
    0 packets output, 0 bytes
    Transmitted 0 unicast, 0 broadcast, 0 multicast
0 underruns, 0 output errors, 0 pause output
```

7.2 ARP 代理配置

7.2.1 简介

代理 ARP 是 ARP 协议的一个变种。 对于没有配置缺省网关的计算机要和其他网络中的计算机实现通信，网关收到源计算机的 ARP 请求会使用自己的 MAC 地址与目标计算机的 IP 地址对源计算机进行应答。代理 ARP 就是将一个主机作为对另一个主机 ARP 进行应答。它能使得在不影响路由表的情况下添加一个新的 Router，使得子网对该主机来说变得更透明化。同时也会带来巨大的风险，除了 ARP 欺骗，和某个网段内的 ARP 增加，最重要的就是无法对网络拓扑进行网络概括。proxy ARP 的最主要的一个优点在于能够在不影响其他 router 的路由表的情况下在网络上添加一个新的 router,这样使得子网的变化对主机是透明的 。代理 ARP 的使用一般是使用在没有配置默认网关和路由策略的网络上的。

代理 ARP 又分为普通的 ARP 代理和本地 ARP 代理。

同一网段内连接到设备的不同 VLAN 接口的主机，可以利用设备的代理 ARP 功能，通过三层转发实现互通。

为了实现三层互通，如果以太网交换机或其下挂的交换机开启了二层端口隔离功能，则需要开启本地代理 ARP 功能。注意：本地 ARP 代理功能开启后，ICMP 重定向功能将自动关闭。

7.2.2 配置普通 ARP 代理

I. 拓扑

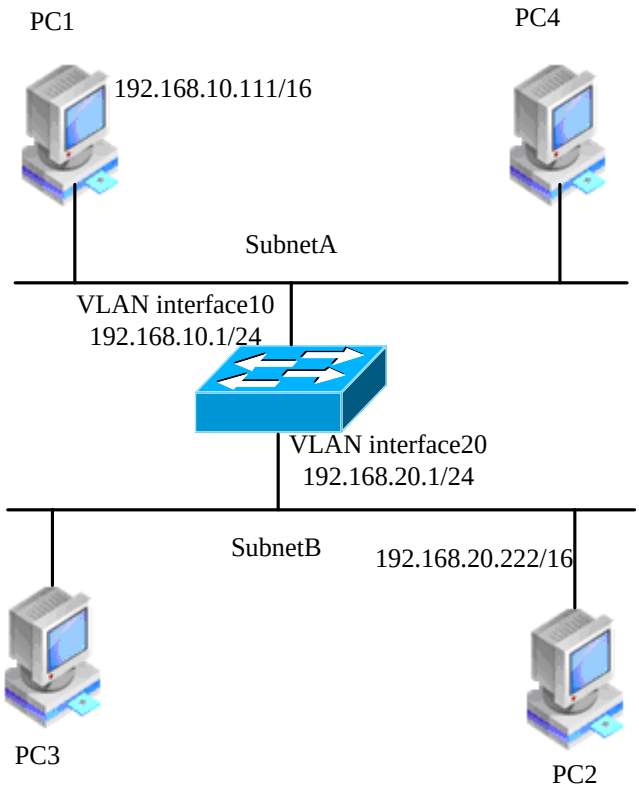


图7-1 ARP 代理拓扑

II. 配置

如上图所示，PC1 属于 VLAN10，PC2 属于 VLAN20，在 VLAN interface10 和 VLAN interface 20 上各自配置 ARP 代理以实现 PC1 和 PC2 之间的互通。

按照下面的配置步骤在 VLAN10 和 VLAN 20 上使能 ARP 代理功能。

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	进入 VLAN database
Switch(config-vlan)# vlan 10,20	创建 VLAN 10, VLAN 20
Switch(config-vlan)# exit	退出 VLAN database
Switch(config)# interface eth-0-22	进入接口模式
Switch(config-if)# switchport access vlan 10	将接口加到 vlan 10 中
Switch(config-if)# no shutdown	使能接口

Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-23	进入接口模式
Switch(config-if)# switchport access vlan 20	将接口加到 vlan 20 中
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出接口模式
Switch(config)# interface vlan 10	创建 3 层 vlan 接口 10，进入接口模式
Switch(config-if)# ip address 192.168.10.1/24	配置接口地址
Switch(config-if)# proxy-arp enable	使能 ARP 代理功能
Switch(config-if)# exit	退出接口模式
Switch(config)# interface vlan 20	创建 3 层 vlan 接口 20，进入接口模式
Switch(config-if)# ip address 192.168.20.1/24	配置接口地址
Switch(config-if)# proxy-arp enable	使能 ARP 代理功能
Switch(config-if)# exit	退出接口模式

III. 命令验证

交换机上的输出结果

Switch# show ip interface vlan 10

```
Interface vlan10
  Interface current state: UP
  Internet address(es):
    192.168.10.1/24 broadcast 192.168.10.255
  Joined group address(es):
    224.0.0.1
  The maximum transmit unit is 1500 bytes
  ICMP error messages limited to one every 1000 milliseconds
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are always sent
  ARP timeout 01:00:00,  ARP retry interval 1s
  ARP Proxy is enabled, Local ARP Proxy is disabled
  VRRP master of :  VRRP is not configured on this interface
```

Switch# show ip interface vlan 20

```
Interface vlan20
  Interface current state: UP
```

```
  Internet address(es):
    192.168.20.1/24 broadcast 192.168.20.255
  Joined group address(es):
    224.0.0.1
  The maximum transmit unit is 1500 bytes
  ICMP error messages limited to one every 1000 milliseconds
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are always sent
  ARP timeout 01:00:00,  ARP retry interval 1s
  ARP Proxy is enabled, Local ARP Proxy is disabled
  VRRP master of :  VRRP is not configured on this interface
```

Switch# show ip arp

```
 Protocol    Address          Age (min)  Hardware Addr   Interface
 Internet    192.168.10.1            -   7cc3.11f1.aa00  vlan10
 Internet    192.168.10.111          5   0cf9.11b6.6e2e  vlan10
 Internet    192.168.20.1            -   7cc3.11f1.aa00  vlan20
 Internet    192.168.20.222          6   5a94.031f.2357  vlan20
```

主机 PC1 上的输出结果

[Host:~]$ ifconfig eth0

```
eth0      Link encap:Ethernet  HWaddr 0C:F9:11:B6:6E:2E
          inet addr:192.168.10.111  Bcast:192.168.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1600  Metric:1
          RX packets:11 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:588 (588.0 b)  TX bytes:700 (700.0 b)
          Interrupt:5
```

[Host:~]$ arp –a

```
? (192.168.20.222) at 7c:c3:11:f1:aa:00 [ether] on eth0
```

[Host: ~]$ route -v

```
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.0.0     *               255.255.0.0     U     0      0        0 eth0
```

[Host:~]$ ping 192.168.20.222

```
PING 192.168.20.222 (192.168.20.222) 56(84) bytes of data.
64 bytes from 192.168.20.222: icmp_seq=0 ttl=63 time=189 ms
64 bytes from 192.168.20.222: icmp_seq=1 ttl=63 time=65.2 ms
--- 192.168.20.222 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 65.209/127.226/189.244/62.018 ms, pipe 2
```

主机 PC2 上的输出结果

[Host:~]$ ifconfig eth0

```
eth0      Link encap:Ethernet  HWaddr 5A:94:03:1F:23:57
          inet addr:192.168.20.222  Bcast:192.168.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1600  Metric:1
          RX packets:14 errors:0 dropped:0 overruns:0 frame:0
          TX packets:17 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:784 (784.0 b)  TX bytes:1174 (1.1 KiB)
          Interrupt:5
```

[Host:~]$ arp -a

```
? (192.168.10.111) at 7c:c3:11:f1:aa:00 [ether] on eth0
[Host: ~]$ route -v
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.0.0     *               255.255.0.0     U     0      0        0 eth0
```

[Host: ~]$ ping 192.168.10.111

```
PING 192.168.10.111 (192.168.10.111) 56(84) bytes of data.
64 bytes from 192.168.10.111: icmp_seq=0 ttl=63 time=53.8 ms
64 bytes from 192.168.10.111: icmp_seq=1 ttl=63 time=65.8 ms
--- 192.168.10.111 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1007ms
rtt min/avg/max/mdev = 53.832/59.842/65.852/6.010 ms, pipe 2
```

7.2.3 配置本地 ARP 代理

I. 拓扑

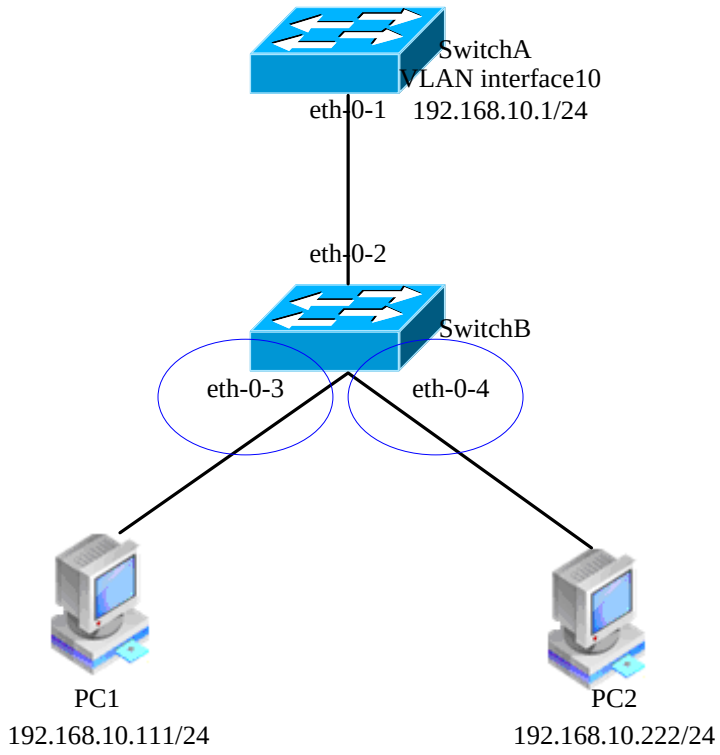


图7-2 本地 ARP 代理拓扑

II. 配置

如图 7-2 所示，Switch B 上的 3 个 2 层端口 eth2，eth3 和 eth4 都属于 VLAN10，其中端口 3 和端口 4 在同一个隔离组 1，所以端口 3 和端口 4 不能互相通信。端口 2 在隔离组 3，所以端口 2 能和端口 3，4 互相通信。PC1 和 PC2 分别连接到 Switch B 的 eth3 和 eth4 口，它们都属于 VLAN10。

通过如下配置步骤，使 PC1 和 PC2 之间实现 3 层互通：

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN database
Switch(config-vlan)# vlan 10	创建 vlan 10
Switch(config-vlan)# exit	退出 VLAN database
Switch(config)# interface eth-0-3	进入接口模式

Switch(config-if)# switchport access vlan 10	将接口加到 vlan 10
Switch (config-if)# no shutdown	使能接口
Switch (config-if)# exit	退出接口模式
Switch(config)# interface eth-0-4	进入接口模式
Switch(config-if)# switchport access vlan 10	将接口加到 vlan 10
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# switchport access vlan 10	将接口加到 vlan 10
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出接口模式
Switch(config)# port-isolate mode l2	设置端口隔离的模式
Switch(config-if)# interface eth-0-3	进入接口模式
Switch(config-if)# port-isolate group 1	配置端口属于隔离组 1
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-4	进入接口模式
Switch(config-if)# port-isolate group 1	配置端口属于隔离组 1
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# port-isolate group 3	配置端口属于隔离组 3
Switch(config-if)# exit	退出接口模式

SwitchA

Switch# configure terminal	进入配置模式
Switch(config)# vlan database	进入 VLAN database
Switch(config-vlan)# vlan 10	创建 vlan 10

Switch(config-vlan)# exit	退出 VLAN database
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# switchport access vlan 10	将接口加到 vlan 10
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出接口模式
Switch(config)# interface vlan 10	创建 3 层接口 vlan10，进入接口模式
Switch(config-if)# ip address 192.168.10.1/24	配置接口的 3 层地址
Switch(config-if)# local-proxy-arp enable	使能本地 ARP 代理
Switch(config-if)# exit	退出接口模式

I. 命令验证

交换机 SwitchA 上的输出结果

Switch# show ip arp

```
 Protocol    Address          Age (min)  Hardware Addr   Interface
 Internet    192.168.10.1            -   eeb4.2a8d.6c00  vlan10
 Internet    192.168.10.111          0   34b0.b279.5f67  vlan10
 Internet    192.168.10.222          0   2a65.9618.57fa  vlan10
```

Switch# show ip interface vlan 10

```
Interface vlan10
  Interface current state: UP
  Internet address(es):
    192.168.10.1/24 broadcast 192.168.10.255
  Joined group address(es):
    224.0.0.1
  The maximum transmit unit is 1500 bytes
  ICMP error messages limited to one every 1000 milliseconds
  ICMP redirects are never sent
  ICMP unreachables are always sent
  ICMP mask replies are always sent
  ARP timeout 01:00:00,  ARP retry interval 1s
  ARP Proxy is disabled, Local ARP Proxy is enabled
  VRRP master of :  VRRP is not configured on this interface
```

主机 PC1 上的输出结果

[Host: ~]$ ifconfig eth0

```
eth0      Link encap:Ethernet  HWaddr 34:B0:B2:79:5F:67
```

```
          inet addr:192.168.10.111  Bcast:192.168.10.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1600  Metric:1
          RX packets:22 errors:0 dropped:0 overruns:0 frame:0
          TX packets:28 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1344 (1.3 KiB)  TX bytes:2240 (2.1 KiB)
          Interrupt:5
```

[Host: ~]$ arp -a

```
? (192.168.10.222) at ee:b4:2a:8d:6c:00 [ether] on eth0
```

[Host: ~]$ ping 192.168.10.222

```
PING 192.168.10.222 (192.168.10.222) 56(84) bytes of data.
64 bytes from 192.168.10.222: icmp_seq=0 ttl=63 time=131 ms
64 bytes from 192.168.10.222: icmp_seq=1 ttl=63 time=159 ms
--- 192.168.10.222 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1003ms
rtt min/avg/max/mdev = 131.078/145.266/159.454/14.188 ms, pipe 2
```

主机 PC2 上的输出结果

[Host:~]$ ifconfig eth0

```
eth0      Link encap:Ethernet  HWaddr 2A:65:96:18:57:FA
          inet addr:192.168.10.222  Bcast:192.168.10.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1600  Metric:1
          RX packets:19 errors:0 dropped:0 overruns:0 frame:0
          TX packets:20 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1148 (1.1 KiB)  TX bytes:1524 (1.4 KiB)
          Interrupt:5
```

[Host:~]$ arp -a

```
? (192.168.10.111) at ee:b4:2a:8d:6c:00 [ether] on eth0
```

[Host: ~]$ ping 192.168.10.111

```
PING 192.168.10.111 (192.168.10.111) 56(84) bytes of data.
64 bytes from 192.168.10.111: icmp_seq=0 ttl=63 time=198 ms
64 bytes from 192.168.10.111: icmp_seq=1 ttl=63 time=140 ms
64 bytes from 192.168.10.111: icmp_seq=2 ttl=63 time=146 ms
--- 192.168.10.111 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2008ms
rtt min/avg/max/mdev = 140.196/161.959/198.912/26.267 ms, pipe 2
```

7.3 DHCP Client 配置

7.3.1 简介

DHCP（Dynamic Host Configuration Protocol） client 通过 DHCP 协议从 DHCP server 动态获得 ip 地址和配置参数。若客户端和服务器都在一个子网内，则客户端和服务器之间可以直接进行 DHCP 协议的交互，否则需要有 DHCP relay agent 转发 DHCP 消息。

DHCP client 通过 DHCP 广播报文向 DHCP server 请求 ip 地址，在获得 ip 地址和相应的租期后，配置地址并设置租期的时间。在租期过半的时候开始发送 DHCP 报文请求继续使用当前的 ip 地址，并期望获得新的租期。在成功续租后，DHCP client 更新租期的时间。

DHCP client 可以向 server 请求的选项包括：router，static-route，classless-static-route，classless-static-route-ms，tftp-server-address，dns-nameserver ，domain-name，netbios-nameserver，vendor-specific。选项 router，static-route，classless-static-route，classless-static-route-ms，tftp-server-address 默认是被请求的，可以通过命令取消这些请求。

7.3.2 配置

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# no switchport	将接口设置三层接口
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# no dhcp client request static-route	取消 static-route 选项请求
Switch(config-if)# ip address dhcp	启用 DHCP client
Switch(config-if)# end	退出到特权模式

7.3.3 命令验证

检查接口配置

Switch# show running-config interface eth-0-1

```
Building configuration...
!
interface eth-0-1
 no switchport
 ip address dhcp
 no dhcp client request static-route
!
```

检查 DHCP client 工作状态

Switch# show dhcp client verbose

```
DHCP client informations:
============================================================
eth-0-1 DHCP client information:
  Current state: BOUND
  Allocated IP: 4.4.4.199 255.255.255.0
  Lease/renewal/rebinding: 1187/517/1037 seconds
  Lease from 2011-11-18 05:59:59 to 2011-11-18 06:19:59
  Will Renewal in 0 days 0 hours 8 minutes 37 seconds
  DHCP server: 4.4.4.1
  Transaction ID: 0x68857f54
  Client ID: switch-7e39.3457.b700-eth-0-1
```

显示 DHCPclient 统计

Switch# show dhcp client statistics

```
DHCP client packet statistics:
============================================================
DHCP OFFERS    received: 1
DHCP ACKs      received: 2
DHCP NAKs      received: 0
DHCP Others    received: 0
DHCP DISCOVER      sent: 1
DHCP DECLINE       sent: 0
DHCP RELEASE       sent: 0
DHCP REQUEST       sent: 2
DHCP packet send failed: 0
```

7.4 DHCP Relay 配置

7.4.1 简介

DHCP 服务器和客户端都在一个子网内，则客户端和服务器之间可以直接进行 DHCP 协议的交互，这时不需要启动 DHCP Relay 功能。如果 DHCP 服务器和客户端不在一个子网内，则需要启动 DHCP Relay 功能将 DHCP 报文转发到外部的 DHCP 服务器。

DHCP Relay 转发同正常的 IP 路由转发不同，IP 路由转发的 IP 数据包在网络之间透明交换，而 DHCP Relay 代理接收 DHCP 消息同时产生一个新的 DHCP 消息发送到另一个接口。DHCP Relay 代理在报文中设置网关地址，添加中继代理信息（option82），转发到 DHCP 服务器端。通过 DHCP Relay 代理，在收到服务器响应的消息时，会移除消息中 option82 内容后，转发给客户端。

7.4.2 拓扑图

下图为测试 DHCP 中继代理功能的网络拓扑，需要两台 PC 机和一台交换机构建测试环境。

- 计算机 A 作为 DHCP 服务器
- 计算机 B 作为 DHCP 客户端
- 交换机作为 DHCP 中继代理

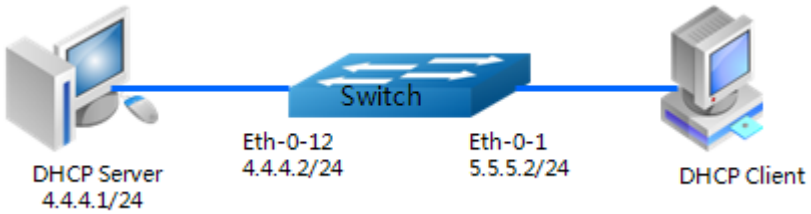


图7-3 DHCP 中继拓扑图

7.4.3 配置

配置接口 eth-0-12

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-12	进入接口配置模式
Switch(config-if)# no switchport	将接口设置三层接口
Switch(config-if)# ip address 4.4.4.2/24	设置 IP 地址
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出接口配置模式

配置 DHCP 服务器组

Switch(config)# dhcp-server 1 4.4.4.1	创建 DHCP 服务器组

配置接口 eth-0-1

Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# no switchport	将接口设置三层接口.

Switch(config-if)# ip address 5.5.5.2/24	将接口设置三层接口
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# dhcp relay information trusted	设置接口启用 DHCP 中继 option82 功能
Switch(config-if)# dhcp-server 1	设置 DHCP 服务器
Switch(config-if)# exit	退出接口配置模式

使能 DHCP 中继全局服务器

Switch(config)# service dhcp enable	使能 DHCP 服务器
Switch(config)# dhcp relay	使能 DHCP Relay 功能

7.4.4 命令验证

步骤 1 检查接口配置。

Switch# show running-config interface eth-0-12

```
!
interface eth-0-12
 no switchport
 ip address 4.4.4.2/24
!
Switch# show running-config interface eth-0-1
!
interface eth-0-1
 no switchport
 dhcp relay information trusted
 dhcp-server 1
 ip address 5.5.5.2/24
!
```

步骤 2 检查 DHCP 服务器状态。

Switch# show services

```
Networking services configuration:
Service Name         Status
============================================================
dhcp                   enable
```

步骤 3 检查 DHCP 服务器组配置。

Switch# show dhcp-server

```
DHCP server group information:
============================================================
group 1 ip address list:
[1] 4.4.4.1
```

步骤 4 显示 DHCP 中继统计检查 DHCP 中继统计。

Switch# show dhcp relay statistics

```
DHCP relay packet statistics:
============================================================
Client relayed packets:  20
Server relayed packets:  20
Client error packets:     20
Server error packets:     0
Bogus GIADDR drops:       0
Bad circuit ID packets:   0
Corrupted agent options:  0
Missing agent options:    0
Missing circuit IDs:      0
```

步骤 5 检查计算机从 DHCP 服务器获取的 IP 地址。

Ipconfig /all

```
Dhcp Enabled. . . . . . . . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IP Address. . . . . . . . . . . . : 5.5.5.1
Subnet Mask . . . . . . . . . . . : 255.255.255.0
Default Gateway . . . . . . . . . : 5.5.5.2
DHCP Server . . . . . . . . . . . : 4.4.4.1
DNS Servers . . . . . . . . . . . : 4.4.4.1
```

7.5 DHCP server 配置

7.5.1 简介

DHCP server 通过 DHCP 协议为 client 提供 ip 地址和网络配置参数。为了能够给客户端提供 DHCP 服务，DHCP server 需要完成一些基本的配置，例如，地址池的分配，默认网关的设置，网络参数的设置。在实际工作的时候，DHCP server 会从设置的地址池内找到可用的地址分配给请求地址的 DHCP client，同时，将 client 请求的网络配置参数发送给 client。这些分配的地址和参数都有一个有效期限（租约），client 需要在到期之前向 server 发出续约请求，保留自己的 ip 地址，同时更新租约。

在实际环境中，若 DHCP server 和 DHCP client 在同一子网内，则 DHCP server 在直接相连后就可以正常工作。若它们不在同一网段内，则 DHCP server 需要 DHCP relay 协助转发 DHCP 消息，才能为 client 提供 DHCP 服务。

DHCP server 支持的主要 option 包括：bootfile-name，dns-server，domain-name，gateway，netbios-name-server，netbios-node-type，tftp-server-address。同时，支持部分 raw option。

7.5.2 拓扑

图 5-1：DHCP server 拓扑图

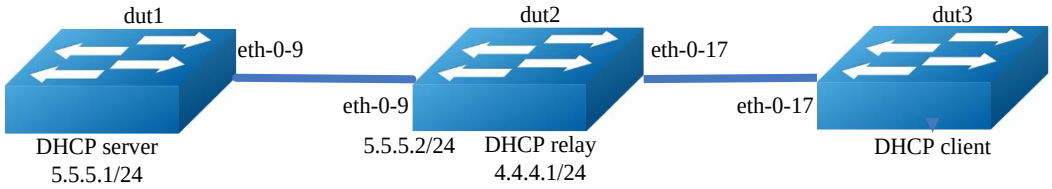


图 5-2：DHCP relay 参与的拓扑图

上图为测试 DHCP server 的网络拓扑。

7.5.3 配置

拓扑 1：

□□ DHCP Server(dut1)

Switch#configure terminal	进入全局配置模式
Switch(config)#service dhcp enable	全局启用 DHCP 服务
Switch(config)#dhcp server	全局启用 DHCP server
Switch(config)#dhcp pool pool5	添加 dhcp pool，进入 DHCP 配置模式
Switch(dhcp-config)#network 5.5.5.0/24	配置地址池
Switch(dhcp-config)#gateway 5.5.5.1	配置 option：默认网关
Switch(dhcp-config)#exit	退出 DHCP 配置模式
Switch(config)#interface eth-0-9	进入接口配置模式
Switch (config-if)#no switchport	将接口设置三层接口
Switch (config-if)# no shutdown	使能接口
Switch (config-if)# ip address 5.5.5.1/24	配置 IP 地址
Switch (config-if)# dhcp server enable	启用 DHCP server
Switch (config-if)#exit	退出接口配置模式

□□DHCP Client(dut2)

Switch#configure terminal	进入全局配置模式
Switch(config)#interface eth-0-9	进入接口配置模式
Switch (config-if)#no switchport	将接口设置三层接口
Switch (config-if)# no shutdown	使能接口
Switch (config-if)# ip address dhcp	启用 DHCP client
Switch (config-if)#exit	退出接口配置模式

拓扑 2：

□□DHCP Server(dut1)

Switch#configure terminal	进入全局配置模式
Switch(config)#service dhcp enable	全局启用 DHCP 服务
Switch(config)#dhcp server	全局启用 DHCP server
Switch(dhcp-config)#dhcp pool pool4	添加 dhcp pool，进入 DHCP 配置模式
Switch(dhcp-config)#network 4.4.4.0/24	配置地址池
Switch(dhcp-config)#gateway 4.4.4.1	配置 option：默认网关
Switch(dhcp-config)#exit	退出 DHCP 配置模式
Switch(config)#ip route 4.4.4.0/24 5.5.5.2	添加路由
Switch(config)#interface eth-0-9	进入接口配置模式
Switch (config-if)#no switchport	将接口设置三层接口
Switch (config-if)# no shutdown	使能接口
Switch (config-if)# ip address 5.5.5.1/24	配置 IP 地址
Switch (config-if)# dhcp server enable	启用 DHCP server
Switch (config-if)#exit	退出接口配置模式

□□DHCP Relay(dut2)

Switch#configure terminal	进入全局配置模式
Switch(config)#service dhcp enable	全局启用 DHCP 服务
Switch(config)#dhcp relay	全局启用 DHCP relay

Switch(config)#dhcp-server 1 5.5.5.1	添加 DHCP server 组
Switch(config)#interface eth-0-17	进入接口配置模式
Switch (config-if)#no switchport	将接口设置成三层口
Switch (config-if)# no shutdown	使能接口
Switch (config-if)# ip address 4.4.4.1/24	配置 IP 地址
Switch (config-if)# dhcp-server 1	选择 DHCP server 组
Switch (config-if)#interface eth-0-9	进入接口配置模式
Switch (config-if)#no switchport	将接口设置成三层口
Switch (config-if)# no shutdown	使能接口
Switch (config-if)# ip address 5.5.5.2/24	配置 IP 地址
Switch (config-if)#exit	退出接口配置模式

□□DHCP Client(dut3)

Switch#configure terminal	进入全局配置模式
Switch(config)#interface eth-0-17	进入接口配置模式
Switch (config-if)#no switchport	将接口设置三层接口
Switch (config-if)# no shutdown	使能接口
Switch (config-if)# ip address dhcp	启用 DHCP client
Switch (config-if)#exit	退出接口配置模式

7.5.4 命令验证

拓扑 1：

查看 DHCP Server（dut1）□□□

Switch# show running-config

```
!
service dhcp enable
!
interface eth-0-9
 no switchport
 dhcp server enable
 ip address 5.5.5.1/24!
!
dhcp server
```

```
dhcp pool pool5
 network 5.5.5.0/24
 gateway 5.5.5.1
```

DHCP Server（dut1）上查看 DHCP client□□□

Switch# show dhcp client verbose

```
DHCP client informations:
============================================================
eth-0-9 DHCP client information:
  Current state: BOUND
  Allocated IP: 5.5.5.2 255.255.255.0
  Lease/renewal/rebinding: 1194/546/1044 seconds
  Lease from 2012-02-04 07:40:12 to 2012-02-04 08:00:12
  Will Renewal in 0 days 0 hours 9 minutes 6 seconds
  DHCP server: 5.5.5.1
  Transaction ID: 0x45b0b27b
  Default router: 5.5.5.1
  Classless static route:
    Destination: 5.5.4.0, mask: 255.255.255.0, Nexthop: 5.5.5.1
  TFTP server addresses: 5.5.5.3
Client ID: switch-6e6e.361f.8400-eth-0-9
```

DHCP Server（dut1）上查看 DHCP server 统计：

Switch# show dhcp server statistics

```
DHCP server packet statistics:
============================================================
Message Received:
BOOTREQUEST:  0
DHCPDISCOVER:  1
DHCPREQUEST:  1
DHCPDECLINE:  0
DHCPRELEASE:  0
DHCPINFORM:  0
Message Sent:
BOOTREPLY:  0
DHCPOFFER:  1
DHCPACK:  1
DHCPNAK:  0
```

DHCP Server（dut1）上查看 DHCP server 地址分配及接口信息：

Switch# show dhcp server binding all

```
 IP address      Client-ID/            Lease expiration            Type
                 Hardware address
 5.5.5.2         6e:6e:36:1f:84:00     Sat 2012.02.04 08:00:12     Dynamic
```

Switch# show dhcp server interfaces

```
List of DHCP server enabled interface(s):
DHCP server service status: enabled
Interface Name
```

```
============================================================
eth-0-9
```

拓扑 2：

查看 DHCP Server（dut1）□□□

Switch# show running-config

```
!
service dhcp enable
!
interface eth-0-9
 no switchport
 dhcp server enable
 ip address 5.5.5.1/24!
!
ip route 4.4.4.0/24 5.5.5.2
!
dhcp server
dhcp pool pool4
 network 4.4.4.0/24
 gateway 4.4.4.1
```

DHCP Server（dut1）上查看 DHCP client□□□

Switch# show dhcp client verbose

```
DHCP client informations:
============================================================
eth-0-17 DHCP client information:
  Current state: BOUND
  Allocated IP: 4.4.4.5 255.255.255.0
  Lease/renewal/rebinding: 1199/517/1049 seconds
  Lease from 2012-02-06 05:23:09 to 2012-02-06 05:43:09
  Will Renewal in 0 days 0 hours 8 minutes 37 seconds
  DHCP server: 5.5.5.1
  Transaction ID: 0x192a4f7d
  Default router: 4.4.4.1
  Classless static route:
    Destination: 5.5.4.0, mask: 255.255.255.0, Nexthop: 4.4.4.1
  TFTP server addresses: 5.5.5.3
  Client ID: switch-3c9a.b29a.ba00-eth-0-17
```

DHCP Server（dut1）上查看 DHCP server 统计：

Switch# show dhcp server statistics

```
DHCP server packet statistics:
============================================================
Message Received:
BOOTREQUEST:  0
DHCPDISCOVER:  1
DHCPREQUEST:  1
DHCPDECLINE:  0
```

```
DHCPRELEASE:  0
DHCPINFORM:  0
Message Sent:
BOOTREPLY:  0
DHCPOFFER:  1
DHCPACK:  1
DHCPNAK:  0
```

DHCP Server（dut1）上查看 DHCP server 地址分配及接口信息：

Switch# show dhcp server binding all

```
IP address      Client-ID/            Lease expiration            Type
                 Hardware address
 4.4.4.5         3c:9a:b2:9a:ba:00     Mon 2012.02.06 05:43:09      Dynamic
```

Switch# show dhcp server interfaces

```
List of DHCP server enabled interface(s):
DHCP server service status: enabled
Interface Name
============================================================
eth-0-9
```

7.6 DNS 配置

7.6.1 简介

DNS 是域名系统(Domain Name System)的缩写，通过这个分布式数据库，你可以将主机名称映射到 IP 地址。当你在交换机上配置 DNS 时，你可以在所有与 IP 相关的命令，如 ping、telnet、connect 以及 telnet 支持的其他相关操作中用主机名代替 IP 地址。

IP 被定义为一个有层次的名称摘要。这些域名使用点(.)分隔。

要解析域名,必须要定义一个域名服务器，该服务器保存了将域名解析为 IP 地址的域名缓存(或数据库)。为了能够将域名解析为 ip 地址，用户必须指定本网络中有效的服务器，然后再启用 DNS。

7.6.2 拓扑

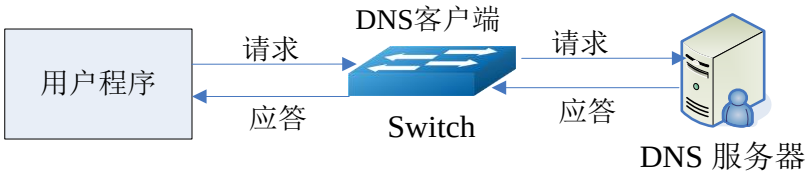


图7-4 典型的 DNS 拓扑

7.6.3 配置

Switch#configure terminal	进入全局配置模式
Switch(config)#dns domain server1	定义一个默认域名系统的的名称，该程序将主机域名（非点分十进制的字符串）映射到 IP 地址.
Switch(config)#dns server 202.100.10.20	配置域名系统(DNS)中的域名服务器的 Ipv4 地址,域名系统用来解决内部的域名查询.
Switch(config)# ip host www.example1.com 192.0.2.141	设置静态域名解析表中主机名及其对应的主机 IPv4 地址

命令验证

Switch# show dns server

```
Current DNS name server configuration:
     Server                 IP Address
----------------------------------------------------------------
1    nameserver             202.100.10.20
```

8 IP 路由配置指导

8.1 IP Unicast-Routing 配置

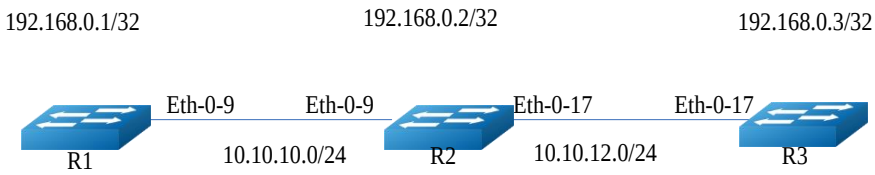
8.1.1 简介

静态路由是一种特殊的路由，由管理员手工配置。当网络结构比较简单时，只需配置静态路由就可以使网络正常工作。合理设置和使用静态路由可以改进网络性能，并可为重要的网络应用保证带宽。静态路由的缺点在于：当网络发生故障或者拓扑发生变化后，可能会出现路由不可达，从而导致网络中断。此时必须由网络管理员手工修改静态路由的配置。

这个例子说明在一个简单的网络拓扑结构下如何使能静态路。静态路由在小型网络中非常有用。静态路由可提供使几个目的地可达的简单解决方案。大型网络使用动态路由协议。静态路由是由网络前缀（主机地址）和下一跳（网关）组成。

路由器 R1 配置三个静态路由，一个是远程网络 10.10.12.0/24，另外两个是到路由器 R2 和 R3 的环回地址（主机地址）。路由器 R3 配置了一条默认静态路由，相当于单独的静态路由配置使用相同的网关或下一跳地址。路由器 R2 有两条路由，每一条路由的目的地都是远端路由器的环回口地址。

8.1.2 拓扑



8.1.3 配置

R1

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-9	进入接口配置模式

Switch(config-if)# no shutdown	端口 UP
Switch(config-if)# no switchport	设置为三层接口
Switch(config-if)# ip address 10.10.10.1/24	配置 IP 地址
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface loopback 0	指定想要配置的环回接口
Switch(config-if)# ip address 192.168.0.1/32	配置 IP 地址和 32bit 掩码，作为主机地址
Switch(config-if)# exit	退出接口配置模式
Switch(config)# ip route 10.10.12.0/24 10.10.10.2 Switch(config)# ip route 192.168.0.2/32 10.10.10.2 Switch(config)# ip route 192.168.0.3/32 10.10.10.2	指定目的前缀和掩码网关所需网络，例如，10.10.12.0/24，为他们每个添加网关（对此所有情况下位 10.10.10.2）。由于 R2 是唯一可用的下一跳，可以配置默认路由而不是配置为单独的地址，见 R3 配置

R2

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-9	进入接口配置模式
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# no switchport	设置为三层接口
Switch(config-if)# ip address 10.10.10.2/24	配置 IP 地址
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface eth-0-17	进入接口配置模式
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# no switchport	设置为三层接口
Switch(config-if)# ip address 10.10.12.2/24	设置 IP 地址
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface loopback 0	指定想要配置的环回接口
Switch(config-if)# ip address 192.168.0.2/32	配置 IP 地址和 32bit 掩码，作为主机地址

Switch(config-if)# exit	退出接口配置模式
Switch(config)# ip route 192.168.0.1/32 10.10.10.1 Switch(config)# ip route 192.168.0.3/32 10.10.12.3	指定目的和掩码，添加网关

R3

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-17	进入接口配置模式
Switch(config-if)# no shutdown	端口 up
Switch(config-if)# no switchport	设置为三层接口
Switch(config-if)# ip address 10.10.12.3/24	设置 IP 地址
Switch(config-if)# exit	退出接口配置模式
Switch(config)# interface loopback 0	指定想要配置的环回接口
Switch(config-if)# ip add 192.168.0.3/32	配置 IP 地址和 32bit 掩码，作为主机地址
Switch(config-if)# exit	退出接口配置模式
Switch(config)# ip route 0.0.0.0/0 10.10.12.2	指定 10.10.12.2 作为到达任意网络的默认网关，因为 10.10.12.2 是唯一的一条可以指定默认网关，而不是单个网络或主机的网关指定。

8.1.4 验证命令

R 1

```
R1# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C       10.10.10.0/24 is directly connected, eth-0-9
C       10.10.10.1/32 is in local loopback, eth-0-9
S       10.10.12.0/24 [1/0] via 10.10.10.2, eth-0-9
```

```
C       192.168.0.1/32 is directly connected, loopback0
S       192.168.0.2/32 [1/0] via 10.10.10.2, eth-0-9
S       192.168.0.3/32 [1/0] via 10.10.10.2, eth-0-9
```

R 2

```
R2# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C       10.10.10.0/24 is directly connected, eth-0-9
C       10.10.10.2/32 is in local loopback, eth-0-9
C       10.10.12.0/24 is directly connected, eth-0-17
C       10.10.12.2/32 is in local loopback, eth-0-17S       192.168.0.1/32 [1/0]
via 10.10.10.1, eth-0-9
C       192.168.0.2/32 is directly connected, loopback0
S       192.168.0.3/32 [1/0] via 10.10.12.3, eth-0-17
```

R 3

```
R3# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
Gateway of last resort is 10.10.12.2 to network 0.0.0.0
S*      0.0.0.0/0 [1/0] via 10.10.12.2, eth-0-17
C       10.10.12.0/24 is directly connected, eth-0-17
C       10.10.12.3/32 is in local loopback, eth-0-17
C       192.168.0.3/32 is directly connected, loopback0
```

8.2 RIP 配置

8.2.1 简介

RIP（Routing Information Protocol，路由信息协议）是一种较为简单的内部网关协议（Interior Gateway Protocol，IGP），主要用于规模较小的网络中。

RIP 是一种基于距离矢量（Distance-Vector）算法的协议，它通过 UDP 报文进行路由信息的交换。RIP 使用跳数（Hop Count）来衡量到达目的地址的距离，称为路由权（RoutingCost）。在 RIP 中，路由器到与它直接相连网络的跳数为 0，通过一个路由器可达的网络的跳数为 1，其余依此类推。为限制收敛时间，RIP 规定 cost 的取值为 0～

15 之间的整数，cost 取值大于或等于 16 的跳数被定义为无穷大，即目的网络或主机不可达。

为提高性能，防止产生路由环，RIP 支持水平分割（Split Horizon）。RIP 还可引入其它路由协议所得到的路由。

8.2.2 配置启用 RIP

在两个交换机上启用 RIP 路由协议的配置步骤如图 8-1 所示。

I. 拓扑

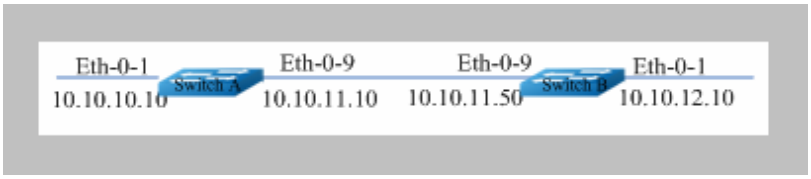


图8-1 RIP Topology

II. 配置

Switch A

SwitchA# configure terminal	进入配置模式
SwitchA(config)# interface eth-0-1	进入接口模式
SwitchA(config-if)# no switchport	设置接口为三层接口
SwitchA(config-if)# ip address 10.10.10.10/24	配置 IP 地址
SwitchA(config-if)# exit	退出接口模式
SwitchA(config)# interface eth-0-9	进入接口模式
SwitchA(config-if)# no switchport	设置接口为三层接口
SwitchA(config-if)# ip address 10.10.11.10/24	配置 IP 地址
SwitchA(config-if)# exit	退出接口模式
SwitchA(config)# router rip	启用 RIP 路由协议
SwitchA(config-router)#network 10.10.10.0/24	发布 10.10.10.0 网段到 RIP 路由协议中
SwitchA(config-router)#network 10.10.11.0/24	发布 10.10.11.0 网段到 RIP 路由协议中

Switch B

SwitchB# configure terminal	进入配置模式
SwitchB(config)# interface eth-0-1	进入接口模式
SwitchB(config-if)# no switchport	设置接口为三层接口
SwitchB(config-if)# ip address 10.10.12.10/24	配置 IP 地址
SwitchB(config-if)# exit	退出接口模式
SwitchB(config)# interface eth-0-9	进入接口模式
SwitchB(config-if)# no switchport	设置接口为三层接口
SwitchB(config-if)# ip address 10.10.11.50/24	设置 IP 地址
SwitchB(config)# router rip	启用 RIP 路由协议
SwitchB(config-router)#network 10.10.11.0/24	发布 10.10.11.0 网段到 RIP 路由协议中
SwitchB(config-router)#network 10.10.12.0/24	发布 10.10.12.0 网段到 RIP 路由协议中

III. 命令验证

使用如下命令，验证上述配置：

show ip rip database，show ip protocols rip，show ip rip interface 和 show ip route

Switch A output

```
SwitchA# show ip rip database
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network            Next Hop        Metric From            If       Time
Rc 10.10.10.0/24                          1                    eth-0-1
Rc 10.10.11.0/24                          1                    eth-0-9
R  10.10.12.0/24      10.10.11.50       2 10.10.11.50     eth-0-9  00:02:52
SwitchA# show ip protocols rip
Routing protocol is "rip"
  Sending updates every 30 seconds with +/-5 seconds, next due in 17 seconds
  Timeout after 180 seconds, Garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Default redistribution metric is 1
  Redistributing:
  Default version control: send version 2, receive version 2
    Interface        Send          Recv   Key-chain
    eth-0-1          2             2
```

```
    eth-0-9          2             2
  Routing for Networks:
    10.10.10.0/24
    10.10.11.0/24
  Routing Information Sources:
    Gateway          Distance  Last Update  Bad Packets  Bad Routes
    10.10.11.50           120  00:00:22               0           0
  Number of routes (including connected): 3
  Distance: (default is 120)
SwitchA# show ip rip interface
eth-0-1 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIP packets
    Send RIP packets
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.10.10.10/24
eth-0-9 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIP packets
    Send RIP packets
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.10.11.10/24
SwitchA# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C       10.10.10.0/24 is directly connected, eth-0-1
C       10.10.10.10/32 is in local loopback, eth-0-1
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.10/32 is in local loopback, eth-0-9
R       10.10.12.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:25:50
```

8.2.3 配置 RIP 的版本

配置路由接口发送接收的 RIP 版本。在下面例子中交换机 B 在 Eth-0-9 和 Eth-0-20 上面发送和接收的 RIP 版本是 v1 和 v2。

I. 拓扑

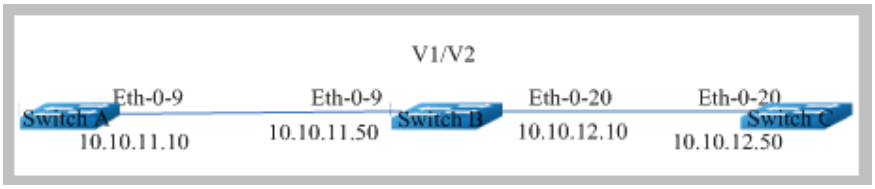


图8-2 RIP Topology II

II. 配置

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# router rip	启用 RIP 路由协议.
Switch(config-router)# exit	退出路由模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# ip rip send version 1 2	设置接口发送的 RIP 版本信息
Switch(config-if)# ip rip receive version 1 2	设置接口接收的 RIP 版本信息
Switch(config-if)# quit	退出接口模式
Switch(config)# interface eth-0-20	进入接口模式
Switch(config-if)# ip rip send version 1 2	设置接口发送的 RIP 版本信息
Switch(config-if)# ip rip receive version 1 2	设置接口接收的 RIP 版本信息

III. 命令验证

使用如下命令，验证上述配置：

show ip rip database，Show running-config，show ip protocols rip，show ip rip interface 和 show ip route

Switch B output

```
Switch# show ip rip database
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, K - Kernel,
       C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network            Next Hop        Metric From            If       Time
R  10.0.0.0/8                             1                    eth-0-9
Rc 10.10.11.0/24                          1                   eth-0-9
Rc 10.10.12.0/24                          1                   eth-0-20
Switch# show ip protocols rip
```

```
Routing protocol is "rip"
  Sending updates every 30 seconds with +/-5 seconds, next due in 1 seconds
  Timeout after 180 seconds, Garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Default redistribution metric is 1
  Redistributing:
  Default version control: send version 2, receive version 2
    Interface        Send          Recv   Key-chain
    eth-0-9          1 2           1 2
    eth-0-20         1 2           1 2
  Routing for Networks:
    10.10.11.0/24
    10.10.12.0/24
  Routing Information Sources:
    Gateway          Distance  Last Update  Bad Packets  Bad Routes
    10.10.11.10           120  00:00:22               0           0
    10.10.12.50           120  00:00:27               0           0
  Number of routes (including connected): 3
  Distance: (default is 120)
Switch# show ip rip inter
eth-0-9 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv1 and RIPv2 packets
    Send RIPv1 and RIPv2 packets
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.10.11.50/24
eth-0-20 is up, line protocol is up
  Routing Protocol: RIP
    Receive RIPv1 and RIPv2 packets
    Send RIPv1 and RIPv2 packets
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IP interface address:
      10.10.12.10/24
Switch# show run
interface eth-0-9
 no switchport
 ip address 10.10.11.50/24
 ip rip send version 1 2
 ip rip receive version 1 2
!
interface eth-0-20
 no switchport
 ip address 10.10.12.10/24
 ip rip send version 1 2
 ip rip receive version 1 2
!
router rip
 network 10.10.11.0/24
 network 10.10.12.0/24
```

Switch A output

```
Switch# show running-config
interface eth-0-9
 no switchport
 ip address 10.10.11.10/24
!
router rip
 network 10.10.11.0/24
```

Switch C output

```
Switch# show running-config
interface eth-0-20
 no switchport
 ip address 10.10.12.50/24
!
router rip
 network 10.10.12.0/24
```

8.2.4 配置 Metric 参数

附加度量值是附加在 RIP 路由上的输入输出度量值，包括发送附加度量值和接收附加度量值。发送附加度量值不会改变路由表中的路由度量值，仅当接口发送 RIP 路由信息时才会添加到发送路由上；接收附加度量值会影响接收到的路由度量值，接口接收到一条合法的 RIP 路由时，在将其加入路由表前会把度量值附加到该路由上。附加度量值一般包括如下的参数：

- 指定增加路由 Metric 的 ACL 参数说明如下。
  - **In**：应用在从邻居路由器学习到的 RIP 的路由上
  - **Out**：应用在发布给邻居路由器 RIP 通告上
- 匹配 ACL 路由的偏移值 Metric
- 应用偏移列表的接口

  如果有一个路由匹配全局偏移表（不指定接口）和一个基于接口的偏移列表，此时基于接口的偏移列表优先。在这种情况下，基于接口的偏移列表的度量值是被加到路由上。

下面例子讲述如何在 SwitchA 上将 1.1.1.0 在 Eth-0-13 接口上增加 metric 3。

I. 拓扑

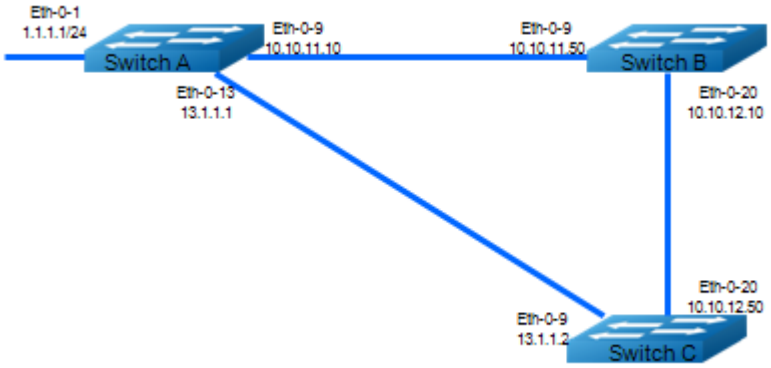


图8-3 RIP Topology III

II. 配置

Switch A configuration

```
interface eth-0-1
no switchport
ip address 1.1.1.1/24
!
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
interface eth-0-13
no switchport
ip address 13.1.1.1/24
!
router rip
network 1.1.1.0/24
network 10.10.11.0/24
network 13.1.1.0/24
```

Switch B configuration

```
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
interface eth-0-20
no switchport
ip address 10.10.12.10/24
!
router rip
network 10.10.11.0/24
network 10.10.12.0/24
```

Switch C configuration

```
interface eth-0-13
no switchport
ip address 13.1.1.2/24
!
interface eth-0-20
no switchport
ip address 10.10.12.50/24
!
router rip
network 10.10.12.0/24
network 13.1.1.0/24
```

Validation route table on Switch C

```
Switch# show ip route rip
R       1.1.1.0/24 [120/2] via 13.1.1.1, eth-0-13, 00:07:46
R       10.10.11.0/24 [120/2] via 13.1.1.1, eth-0-13, 00:07:39
                    [120/2] via 10.10.12.10, eth-0-20, 00:07:39
Change router 1.1.1.0/24 via 10.10.12.10
```

Switch A

Switch# configure terminal	进入配置模式
Switch(config)#ip access-list ripoffset	创建 ACL.
Switch(config-ip-acl)#permit any 1.1.1.0 0.0.0.255 any	匹配相应的网段
Switch(config-ip-acl)# router rip	启用 RIP 路由协议
Switch(config-router)# offset-list ripoffset out 3 eth-0-13	设置偏移列表的 Metric 值

III. 命令验证

Switch C output

Switch# show ip route rip

```
R       1.1.1.0/24 [120/3] via 10.10.12.10, eth-0-20, 00:00:02
R       10.10.11.0/24 [120/2] via 13.1.1.1, eth-0-13, 00:11:40
                    [120/2] via 10.10.12.10, eth-0-20, 00:11:40
```

8.2.5 配置管理距离

默认情况下，RIP 的管理距离是 120。比较路由时，管理距离越低，路由越容易被选中。

下面例子讲述了如何修改 RIP 的管理距离。

I. 拓扑

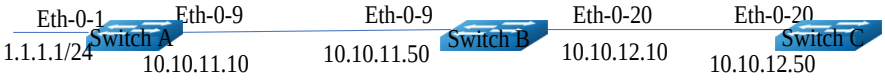


图8-4 RIP Topology Ⅳ

II. 配置

Switch A configuration

```
interface eth-0-1
no switchport
ip address 1.1.1.1/24
!
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
router ospf
network 1.1.1.0/24 area 0
network 10.10.11.0/24 area 0
!
router rip
network 1.1.1.0/24
network 10.10.11.0/24
```

Switch B configuration

```
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
interface eth-0-20
no switchport
ip address 10.10.12.10/24
!
router ospf
network 10.10.11.0/24 area 0
network 10.10.12.0/24 area 0
!
router rip
network 10.10.11.0/24
network 10.10.12.0/24
```

Switch C configuration

```
interface eth-0-20
no switchport
ip address 10.10.12.50/24
!
router ospf
network 10.10.12.0/24 area 0
!
router rip
network 10.10.12.0/24
```

Switch C output

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
O       1.1.1.0/24 [110/3] via 10.10.12.10, eth-0-20, 01:05:49
O       10.10.11.0/24 [110/2] via 10.10.12.10, eth-0-20, 01:05:49
C       10.10.12.0/24 is directly connected, eth-0-20
C       10.10.12.50/32 is in local loopback, eth-0-20
```

通过以下步骤改变交换机 C 上的 1.1.1.0 网段的 RIP 管理距离。

Switch# configure terminal	进入配置模式
Switch(config)#ip access-list ripdistancelist	创建 ACL
Switch(config-ip-acl)#permit any 1.1.1.0 0.0.0.255 any	匹配相应的网段
Switch(config-ip-acl)# router rip	启用 RIP 路由协议
Switch(config-router)# distance 100 0.0.0.0/0 ripdistancelist	设置 RIP 路由的管理距离为 100 0.0.0.0/0 是源 IP 前缀，所有匹配这个网段的路由其管理距离将被设为 100

III. 命令验证

Switch C output

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
```

```
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
R       1.1.1.0/24 [100/3] via 10.10.12.10, eth-0-20, 00:00:02
O       10.10.11.0/24 [110/2] via 10.10.12.10, eth-0-20, 01:10:42
C       10.10.12.0/24 is directly connected, eth-0-20
C       10.10.12.50/32 is in local loopback, eth-0-20
```

8.2.6 配置重分布

你可以将静态路由，直连路由以及其他路由协议比如 OSPF 的路由重分布到 RIP 中并被 RIP 发送给它的邻居。

默认 RIP 的重发布 Metric 为 1，最大 16。

将特定的路由重发布到 RIP 上，其度量值可以是默认的，也可以是修改后的。

下面例子讲述如何重分布其他的路由信息到 RIP。

I. 拓扑

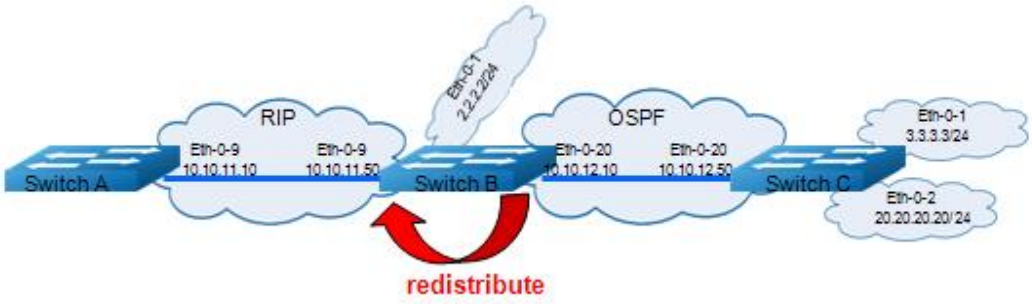


图8-5 RIP Topology V

II. 配置

Switch A configuration

```
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
router rip
network 10.10.11.0/24
```

Switch B configuration

```
interface eth-0-1
no switchport
ip address 2.2.2.2/24
!
interface eth-0-9
```

```
no switchport
ip address 10.10.11.50/24
!
interface eth-0-20
no switchport
ip address 10.10.12.10/24
!
router ospf
network 10.10.12.0/24 area 0
!
router rip
network 10.10.11.0/24
!
ip route 20.20.20.0/24 10.10.12.50
```

Switch C configuration

```
interface eth-0-1
no switchport
ip address 3.3.3.3/24
!
interface eth-0-2
no switchport
ip address 20.20.20.20/24
!
interface eth-0-20
no switchport
ip address 10.10.12.50/24
!
router ospf
network 3.3.3.0/24 area 0
network 10.10.12.0/24 area 0
```

Switch A output

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.10/32 is in local loopback, eth-0-9
```

Switch B output

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
```

```
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C       2.2.2.0/24 is directly connected, eth-0-1
C       2.2.2.02/32 is in local loopback, eth-0-1
O       3.3.3.0/24 [110/2] via 10.10.12.50, eth-0-20, 01:05:41
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.50/32 is in local loopback, eth-0-9
C       10.10.12.0/24 is directly connected, eth-0-20
C       10.10.12.10/24 is in local loopback, eth-0-20
S       20.20.20.0/24 [1/0] via 10.10.12.50, eth-0-20
```

Switch B Configure Redistribute

Switch# configure terminal	进入配置模式
Switch(config)# router rip	启用 RIP 路由协议
Switch(config-router)#default-metric 2	指定默认的 Metric
Switch(config-router)# redistribute static	重分布静态路由
Switch(config-router)# redistribute connected	重分布直连路由
Switch(config-router)#redistribute ospf metric 5	重分布 OSPF 路由到 RIP 中
Switch(config)# router ospf	启用 OSPF 路由协议
Switch(config-router)# redistribute connected	重分布直连路由

III. 命令验证

Switch A output

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
R       2.2.2.0/24 [120/3] via 10.10.11.50, eth-0-9, 00:02:36
R       3.3.3.0/24 [120/6] via 10.10.11.50, eth-0-9, 00:02:26
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.10/32 is in local loopback eth-0-9
R       10.10.12.0/24 [120/3] via 10.10.11.50, eth-0-9, 00:02:36
R       20.20.20.0/24 [120/3] via 10.10.11.50, eth-0-9, 00:02:41
```

8.2.7 配置水平分割参数

通常情况下，连接到广播网络并且使用距离矢量路由协议的路由器，使用水平分割机制来避免环路。配置水平分割可以使得从一个接口学到的路由不能通过此接口向外发布，这通常优化了多个路由器之间的通信，尤其在链路中断时。配置毒性逆转可以使得从一个接口学到的路由还可以从这个接口向外发布，但这些路由的度量值已设置为16，即不可达。

I. 拓扑

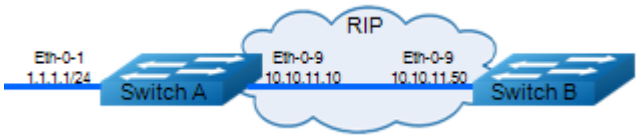


图8-6 RIP Topology VI

II. 配置

Switch A Configuration

```
interface eth-0-1
no switchport
ip address 1.1.1.1/24
!
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
router rip
network 10.10.11.0/24
redistribute connected
```

Switch B Configuration

```
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
router rip
network 10.10.11.0/24
```

Switch B debug Configuration

```
Switch# debug rip packet send detail
Switch# terminal monitor
```

Disable Split-horizon on Switch B

Switch# configure terminal	进入配置模式

Switch(config)#interface eth-0-9	配置接口 eth-0-9
Switch(config-if)# no ip rip split-horizon	禁用水平分割

```
Apr  8 06:24:25 Switch RIP4-7: SEND[eth-0-9]: Send to 224.0.0.9:520
Apr  8 06:24:25 Switch RIP4-7: SEND[eth-0-9]: RESPONSE version 2 packet size 44
Apr  8 06:24:25 Switch RIP4-7: 1.1.1.0/24 -> 0.0.0.0 family 2 tag 0 metric 2
Apr  8 06:24:25 Switch RIP4-7: 10.10.11.0/24 -> 0.0.0.0 family 2 tag 0 metric 1
```

Enable Split-horizon on Switch B

Switch(config-if)# ip rip split-horizon	启用水平分割
Switch(config-if)# ip rip split-horizon poisoned	启用毒性逆转

```
Apr  8 06:38:35 Switch RIP4-7: SEND[eth-0-9]: Send to 224.0.0.9:520
Apr  8 06:38:35 Switch RIP4-7: SEND[eth-0-9]: RESPONSE version 2 packet size 44
Apr  8 06:38:35 Switch RIP4-7: 1.1.1.0/24 -> 0.0.0.0 family 2 tag 0 metric 16
Apr  8 06:38:35 Switch RIP4-7: 10.10.11.0/24 -> 0.0.0.0 family 2 tag 0 metric 16
```

III. 命令验证

使用如下命令，验证上述配置：

show running-config 和 show ip rip interface

8.2.8 配置 Timers

RIP 受多个定时器的控制，比如路由更新的频率，路由失效的时间等等。您可以调整这些计时器以调整 RIP 的性能，以更好地满足您的互联网工作的需要。如下参数可供调整：

- Update 定时器，定义了发送更新报文的时间间隔。
- Timeout 定时器，定义了路由老化时间。如果在老化时间内没有收到关于某条路由的更新报文，则该条路由在路由表中的度量值将会被设置为 16。
- Garbage-Collect 定时器，定义了一条路由从度量值变为 16 开始，直到它从路由表里被删除所经过的时间。

I. 配置

使用如下表所示的命令配置 Timer。

Switch# configure terminal	进入配置模式
Switch(config)# router rip	启用 RIP 路由协议

Switch(config-router)# timers basic 10 180 120	指定路由表 update timer 10 秒，指定路由信息超时 180 秒，垃圾信息收集时间 120 秒

II. 命令验证

使用如下命令，验证上述配置：

show running-config 和 show ip protocols rip

Switch# show ip protocols rip

```
Routing protocol is "rip"
  Sending updates every 10 seconds with +/-5 seconds, next due in 2 seconds
  Timeout after 180 seconds, Garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Default redistribution metric is 1
  Redistributing:
  Default version control: send version 2, receive version 2
    Interface        Send          Recv   Key-chain
    eth-0-9           2             2
  Routing for Networks:
    10.10.11.0/24
  Routing Information Sources:
    Gateway          Distance  Last Update  Bad Packets  Bad Routes
    10.10.11.50           120  00:00:02               0           0
  Number of routes (including connected): 5
  Distance: (default is 120)
```

8.2.9 配置 RIP 路由过滤列表

路由器提供路由信息过滤功能，通过指定访问控制列表和地址前缀列表，可以配置入口或出口过滤策略，对接收或发布的路由进行过滤。一个路由过滤列表通常包括如下参数：

- 一个被用作过滤器的 ACL 或 prefix list。
- **In 方向**：过滤器被应用在学习到的路由上；**Out 方向**：过滤器被应用在发布的路由上。
- 应用过滤器的接口（可选）。

I. 拓扑

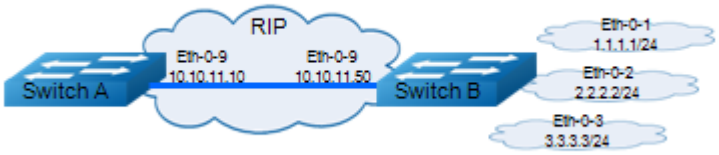


图8-7 RIP Topology VII

II. 配置

Switch A configuration

```
interface eth-0-9
no switchport
ip address 10.10.11.10/24
!
router rip
network 10.10.11.0/24
```

Switch B configuration

```
interface eth-0-1
no switchport
ip address 1.1.1.1/24
!
interface eth-0-2
no switchport
ip address 2.2.2.2/24
!
interface eth-0-3
no switchport
ip address 3.3.3.3/24
!
interface eth-0-9
no switchport
ip address 10.10.11.50/24
!
router rip
network 1.1.1.0/24
network 2.2.2.0/24
network 3.3.3.0/24
network 10.10.11.0/24
```

Switch A output

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
```

```
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
R       1.1.1.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:01:50
R       2.2.2.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:01:50
R       3.3.3.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:01:50
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.10/32 is in local loopback, eth-0-9
```

参照如下表中的命令，配置交换机 B。

Switch# configure terminal	进入配置模式
Switch(config)# ip prefix-list 1 deny 1.1.1.0/24 Switch(config)# ip prefix-list 1 permit any	建立列表
Switch(config)# router rip	启用 RIP 路由协议
Switch(config-router)# distribute-list prefix 1 out	应用策略

III. 命令验证

Switch A output

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
R       2.2.2.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:00:08
R       3.3.3.0/24 [120/2] via 10.10.11.50, eth-0-9, 00:00:08
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.10/32 is in local loopback, eth-0-9
```

8.2.10 配置 RIPv2 验证(single key)

RIP-2 支持两种认证方式：明文认证和 MD5 密文认证。这个例子说明如何使用明文进行认证。Switch A 和 B 是在运行 RIP 路由协议，如果要在交换机上配置明文认证，需要执行如下步骤：

步骤 1 指定一个接口，然后定义该接口的密码。

步骤 2 指定认证模式为明文。

任何从这个指定接口接收的 RIP 数据包应该有相同的字符串作为密码。同样的，Switch B 上也要定义相同的密码和身份验证模式。

II. 拓扑

图8-8 RIPv2

III. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	配置接口 eth-0-1
Switch(config-if)# ip address 1.1.1.1/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config-if)# interface eth-0-9	配置接口 eth-0-9
Switch(config-if)# ip address 10.10.11.10/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router rip	启用 RIP 路由协议
Switch(config-router)# network 10.10.11.0/24	发布网段到 RIP 路由中
Switch(config-router)# redistribute connected	重分布直连路由
Switch(config-router)# exit	退出路由模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# ip rip authentication string Auth1	指定验证的字符串
Switch(config-if)# ip rip authentication mode text	指定验证的模式

Switch B

Switch # configure terminal	进入配置模式
Switch(config)# interface eth-0-1	配置接口 eth-0-1
Switch(config-if)# ip address 2.2.2.2/24	配置 IP 地址

Switch(config-if)# exit	退出接口模式
Switch(config-if)# interface eth-0-9	配置接口 eth-0-9
Switch(config-if)# ip address 10.10.11.50/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router rip	启用 RIP 路由协议
Switch(config-router)# network 10.10.11.0/24	发布网段到 RIP 路由中
Switch(config-router)# redistribute connected	重分布直连路由
Switch(config-router)# exit	退出路由模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# ip rip authentication string Auth1	指定验证的字符串
Switch(config-if)# ip rip authentication mode text	指定验证的模式

IV. 命令验证

使用如下命令，验证上述配置：

show running-config，show ip rip database，show ip protocols rip，show ip rip interface 和 show ip route

8.2.11 配置 RIPv2 MD5 验证 (multiple keys)

这个例子说明了如何使用 MD5 进行 RIP 路由信息交换过程中的验证。对于需要使用 MD5 认证的 Switch A 和 B 来说，首先定义一个钥匙链，然后指定 key 并且配置认证的字符串或密码，然后通过指定接收或者发送的时间来定义 key 生效的时间。最后将该钥匙链应用到接口上并且指定接口的认证模式为 MD5。Switch A 和 B 的密钥配置必须是一样的才能保证 RIP 路由更新信息交换成功。在 MD5 认证中，key ID 和 key 字符串需要同时匹配。在下面的例子中，我们还配置了 key 生效的时间，这样，每隔 5 天，key 就会更新一次。

I. 拓扑

图8-9 RIPv2 MD5 authentication

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	配置接口 eth-0-1
Switch(config-if)# ip address 1.1.1.1/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config-if)# interface eth-0-9	配置接口 eth-0-9
Switch(config-if)# ip address 10.10.11.10/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router rip	启用 RIP 路由协议
Switch(config-router)# network 10.10.11.0/24	发布网段到 RIP 路由中
Switch(config-router)# redistribute connected	重分布直连路由
Switch(config-router)# exit	退出路由模式
Switch(config)# key chain SUN	定义 KEY 链
Switch(config-keychain)# key 1	创建 key id 1
Switch(config-keychain-key)# key-string key1	设置密码
Switch(config-keychain-key)# accept-lifetime 12:00:00 Mar 2 2012 14:00:00 Mar 7 2012	设置应用时间范围
Switch(config-keychain-key)# send-lifetime 12:00:00 Mar 2 2012 12:00:00 Mar 7 2012	设置应用时间范围
Switch(config-keychain-key)# exit	退出
Switch(config-keychain)# key 2	创建 key id 2

Switch(config-keychain-key)# key-string Earth	设置密码
Switch(config-keychain-key)# accept-lifetime 12:00:00 Mar 7 2012 14:00:00 Mar 12 2012	设置应用时间范围
Switch(config-keychain-key)# send-lifetime 12:00:00 Mar 7 2012 12:00:00 Mar 12 2012	设置应用时间范围
Switch(config-keychain-key)# end	退出
Switch # configure terminal	进入配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# ip rip authentication key-chain SUN	定义接口上用验证名字
Switch(config-if)# ip rip authentication mode md5	定义接口上的验证方式

Switch B

Switch # configure terminal	进入配置模式
Switch(config)# interface eth-0-1	配置接口 eth-0-1
Switch(config-if)# ip address 2.2.2.2/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config-if)# interface eth-0-9	配置接口 eth-0-9
Switch(config-if)# ip address 10.10.11.50/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router rip	启用 RIP 路由协议
Switch(config-router)# network 10.10.11.0/24	发布网段到 RIP 路由中
Switch(config-router)# redistribute connected	重分布直连路由
Switch(config-router)# exit	退出路由模式
Switch(config)# key chain SUN	定义 KEY 链
Switch(config-keychain)# key 1	创建 key id 1
Switch(config-keychain-key)# key-string key1	设置密码

Switch(config-keychain-key)# accept-lifetime 12:00:00 Mar 2 2012 14:00:00 Mar 7 2012	设置应用时间范围
Switch(config-keychain-key)# send-lifetime 12:00:00 Mar 2 2012 12:00:00 Mar 7 2012	设置应用时间范围
Switch(config-keychain-key)# exit	退出
Switch(config-keychain)# key 2	创建 key id 2
Switch(config-keychain-key)# key-string Earth	设置密码
Switch(config-keychain-key)# accept-lifetime 12:00:00 Mar 7 2012 14:00:00 Mar 12 2012	设置应用时间范围
Switch(config-keychain-key)# send-lifetime 12:00:00 Mar 7 2012 12:00:00 Mar 12 2012	设置应用时间范围
Switch(config-keychain-key)# end	退出
Switch # configure terminal	进入配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# ip rip authentication key-chain SUN	定义接口上用验证名字
Switch(config-if)# ip rip authentication mode md5	定义接口上的验证方式

III. 命令验证

使用如下命令，验证上述配置：

show running-config，show ip rip，show ip protocols rip，show ip rip interface 和 show key chain

Validate on Switch A

```
Switch# show key chain
key chain SUN:
    key 1 -- text "key1"
      accept-lifetime <12:00:00 Mar 02 2012> - <14:00:00 Mar 07 2012>
      send-lifetime <12:00:00 Mar 02 2012> - < 12:00:00 Mar 07 2012>
    key 2 -- text "Earth"
      accept-lifetime <12:00:00 Mar 07 2012> - <14:00:00 Mar 12 2012>
      send-lifetime <12:00:00 Mar 07 2012> - < 12:00:00 Mar 12 2012>
Switch#
```

Validate on Switch B

```
Switch# show key chain
key chain SUN:
    key 1 -- text "key1"
      accept-lifetime <12:00:00 Mar 02 2012> - <14:00:00 Mar 07 2012>
      send-lifetime <12:00:00 Mar 02 2012> - < 12:00:00 Mar 07 2012>
    key 2 -- text "Earth"
      accept-lifetime <12:00:00 Mar 07 2012> - <14:00:00 Mar 12 2012>
      send-lifetime <12:00:00 Mar 07 2012> - < 12:00:00 Mar 12 2012>
```

8.3 OSPF 配置

8.3.1 简介

开放最短路径优先协议 OSPF（Open Shortest Path First）是 IETF 组织开发的一个基于链路状态的内部网关协议，它支持 IP 子网化以及对外部路由做标记。目前使用的是版本 2（RFC2328），其特性如下：

- 适应范围：支持各种规模的网络，最多可支持几百台路由器。
- 快速收敛：在网络的拓扑结构发生变化后立即发送更新报文，使这一变化在自治系统中同步。
- 无自环：由于 OSPF 根据收集到的链路状态用最短路径树算法计算路由，从算法本身保证了不会生成自环路由。
- 区域划分：允许自治系统的网络被划分成区域来管理，区域间传送的路由信息被进一步抽象，从而减少了占用的网络带宽。
- 等价路由：支持到同一目的地址的多条等价路由。
- 路由分级：使用 4 类不同的路由，按优先顺序来说分别是：区域内路由、区域间路由、第一类外部路由、第二类外部路由。
- 支持验证：支持基于接口的报文验证以保证路由计算的安全性。
- 组播发送：协议报文支持以组播形式发送。

当前的系统支持如下 OSPF 特性：

- **支持末梢区域**：支持路由重分布，这包括将其他路由协议学到的路由导入 OSPF 或者将 OSPF 学到的路由导出到其他路由协议中。
- **支持明文和 MD5 两种认证模式**：支持 OSPF interface 上的参数配置，包括输出度量值，重传时间，发送延时时间，路由器优先级，路由器 hello 报文时间间隔，认证密码等等。
- **不支持虚链路**：不支持 NSSA(Not-So-Stubby Area)

  OSPF 需要多个路由器协同工作，包括区域边界路由器(ABR)，自治系统边界路由器(ASBR)，内部路由器等。最简单的 OSPF 配置只需要使用默认的参数，并且将所有的 OSPF interface 加入同一个区域就可以了。

8.3.2 参考文献

OSPF 模块是基于以下 RFC：

RFC 2328 – OSPF version 2

8.3.3 配置基本 OSPF

在需要启用 OSPF 的路由器上先创建 OSPF 进程，指定需要发布的网段以及区域 ID。配置如下表所示。

Switch# configure terminal	进入配置模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面。你可以使用网络掩码来使多个 interface 加入 OSPF 域
Switch(config-router)# end	返回到配置模式
Switch# show ip protocols	检查配置的协议

在全局模式下通过命令“**no router ospf *process-id***”取消 OSPF 进程。

配置 OSPF 进程号为 109 且发布网段 131.108.0.0 到区域 24 里面：

Switch(config)# router ospf 109

Switch(config-router)# network 131.108.0.0 255.255.255.0 area 24

8.3.4 启用 OSPF

这个例子显示了一个接口上启用 OSPF 所需的最低配置。

一个接口只能属于一个区域，不同的接口可以属于不同的区域

I. 拓扑

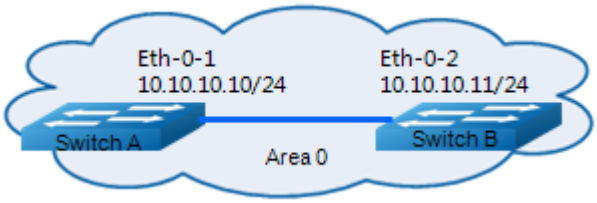


图8-10 OSPF 自治系统

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	配置接口 eth-0-1
Switch(config-if)# ip address 10.10.10.10/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-2	配置接口 eth-0-2
Switch(config-if)# ip address 10.10.10.11/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 200	创建 OSPF 进程号 200
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

III. 命令验证

使用如下命令，验证上述配置：

show ip ospf database， show ip ospf interface， show ip ospf neighbor 和 show ip ospf route

Switch A

```
Switch# show ip ospf database
            OSPF Router with ID (10.10.10.10) (Process ID 100)
                Router Link States (Area 0)
Link ID         ADV Router      Age  Seq#       CkSum  Link count
10.10.10.10     10.10.10.10     51 0x80000002 0xd012         1
Switch# show running-config router ospf
Building configuration...
!
router ospf 100
 network 10.10.10.0/24 area 0
!
```

Switch B

```
Switch# show ip ospf database
            OSPF Router with ID (10.10.10.10) (Process ID 200)
                Router Link States (Area 0)
Link ID         ADV Router      Age  Seq#       CkSum  Link count
10.10.10.10     10.10.10.10      267 0x80000002 0xd012 1
Switch# show running-config router ospf
Building configuration...
!
router ospf 200
 network 10.10.10.0/24 area 0
!
```

8.3.5 配置优先级

这个例子主要讲述了如何配置接口优先级，优先级高的成为 DR。优先级为 0 的不参与 DR 选举。Switch C 的优先级是 10，这比 Switch A 和 Switch B 的默认优先级 1 要高，因此 Switch C 将成为这个网络内的 DR。

I. 拓扑

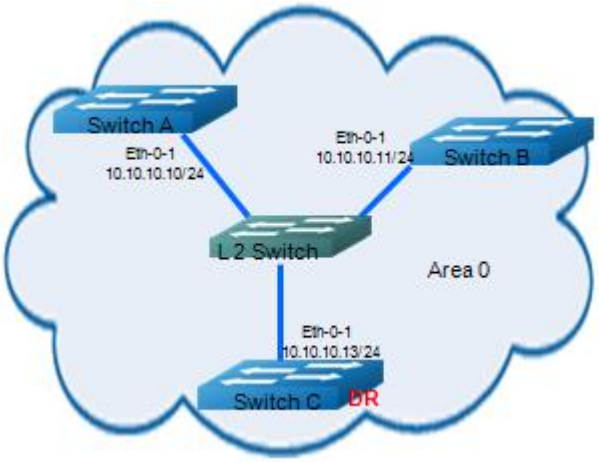


图8-11 OSPF 优先级

II. 配置

Switch C

Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip ospf priority 10	设置接口优先级
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

Switch B

Switch# configure terminal	进入接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

Switch A

Switch(config)# router ospf 200	创建 OSPF 进程号 100

Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

III. 命令验证

使用如下命令，验证以上配置是否正确：

show ip ospf neighbor 和 show ip ospf interface

Switch C

```
Switch# show ip ospf neighbor
OSPF process 0:
Neighbor ID     Pri   State              Dead Time   Address         Interface
10.10.10.10       1   Full/DROther     00:00:32    10.10.10.10    eth-0-1
10.10.10.11       1   Full/BDR          00:00:31    10.10.10.11    eth-0-1
Switch# show ip ospf interface
eth-0-10 is up, line protocol is up
  Internet Address 10.10.10.13/24, Area 0, MTU 1500
  Process ID 0, Router ID 10.10.10.13, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 10, TE Metric 1
  Designated Router (ID) 10.10.10.13, Interface Address 10.10.10.13
  Backup Designated Router (ID) 10.10.10.11, Interface Address 10.10.10.11
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:07
  Neighbor Count is 2, Adjacent neighbor count is 2
  Crypt Sequence Number is 1301567281
  Hello received 188 sent 110, DD received 34 sent 23
  LS-Req received 8 sent 6, LS-Upd received 28 sent 26
  LS-Ack received 32 sent 15, Discarded 0
```

8.3.6 配置 OSPF 区域参数

您可以选择性地配置多个 OSPF 区域参数。这些参数包括用于防止访问未经授权的区域的认证密码，以及将区域配置为末梢区域(Stub)。Stub 区域是一些特定的区域，Stub 区域的 ABR 不传播它们接收到的自治系统外部路由，在这些区域中路由器的路由表规模以及路由信息传递的数量都会大大减少。为保证到自治系统外的路由依旧可达，该区域的 ABR 将生成一条缺省路由，并发布给 Stub 区域中的其他非 ABR 路由器。

路由聚合是指 ABR 或 ASBR 将具有相同前缀的路由信息聚合，只发布一条路由到其它区域。AS 被划分成不同的区域后，区域间可以通过路由聚合来减少路由信息，减小路由表的规模，提高路由器的运算速度。如果网络号是连续的，你可以使用 area range 命令将这些连续的网段聚合成一个网段。这样 ABR 只发送一条聚合后的 LSA，所有属于本命令指定的聚合网段范围的 LSA 将不再会被单独发送出去，这样可减少其它区域中 LSDB 的规模。

I. 拓扑

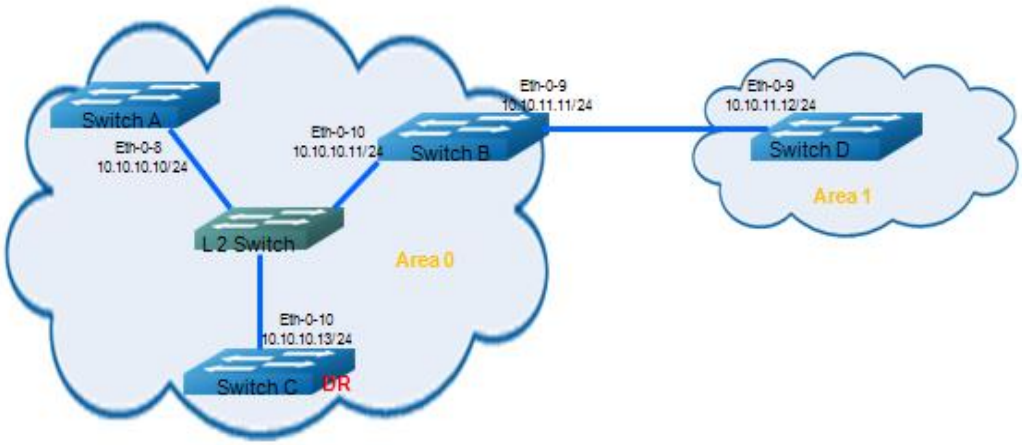


图8-12 OSPF 区域

II. 配置

Switch A

Switch# configure terminal	进入配置模式.
Switch(config)#interface eth-0-8	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.10.10/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

Switch B

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-10	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.10.11/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式

Switch(config)#interface eth-0-9	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.11.11/24	设置端口的 IP 地址
Swithc(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面
Switch(config-router)# network 10.10.11.0/24 area 1	发布 10.10.11.0/24 网段到 OSPF 区域 1 里面
Switch(config-router)# area 0 range 10.10.10.0/24	指定一段 IP 段发布到 OSPF 区域 0
Switch(config-router)# area 1 stub no-summary	区域 1 设置成 Stub 区域
Switch(config-router)# end	返回配置模式
Switch # show ip ospf 100 Switch # show ip ospf 100 database	显示 OSPF 100 的信息

Switch C

Switch # configure terminal	进入配置模式
Switch(config)#interface eth-0-10	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.10.13/24	设置端口的 IP 地址
Switch(config-if)# ip ospf priority 10	设置 OSPF 接口优先级
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

Switch D

Switch # configure terminal	进入配置模式
Switch(config)#interface eth-0-9	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.11.12/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 200	创建 OSPF 进程号 200
Switch(config-router)# network 10.10.11.0/24 area 1	发布 10.10.11.0/24 网段到 OSPF 区域 1 里面
Switch(config-router)# area 1 stub no-summary	区域 1 设置成 Stub 区域

III. 命令验证

使用 **show ip route** 命令验证上述配置。

Switch A

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C       10.10.10.0/24 is directly connected, eth-0-8
C       10.10.10.10/32 is in local loopback, eth-0-8
O IA    10.10.11.0/24 [110/2] via 10.10.10.11, eth-0-8, 00:14:46
```

Switch B

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C       10.10.10.0/24 is directly connected, eth-0-10
C       10.10.10.11/32 is in local loopback, eth-0-10
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.11/32 is in local loopback, eth-0-9
```

Switch C

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
C       10.10.10.0/24 is directly connected, eth-0-10
C       10.10.10.13/32 is in local loopback, eth-0-10
O IA    10.10.11.0/24 [110/2] via 10.10.10.11, eth-0-10, 00:20:35
```

Switch D

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
Gateway of last resort is 10.10.11.11 to network 0.0.0.0
O*IA    0.0.0.0/0 [110/2] via 10.10.11.11, eth-0-9, 00:12:46
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.12/32 is in local loopback, eth-0-9
```

8.3.7 配置 OSPF 重分布路由

区域内和区域间路由描述的是 AS 内部的网络结构，外部路由则描述了应该如何选择到 AS 以外目的地址的路由。OSPF 将引入的 AS 外部路由分为两类：Type1 和 Type2。

第一类外部路由是指接收的是 IGP（Interior Gateway Protocol，内部网关协议）路由（例如静态路由和 RIP 路由）。由于这类路由的可信程度较高，并且和 OSPF 自身路由的开销具有可比性，所以到第一类外部路由的开销等于本路由器到相应的 ASBR 的开销与 ASBR 到该路由目的地址的开销之和。

第二类外部路由是指接收的是 EGP（Exterior Gateway Protocol，外部网关协议）路由。由于这类路由的可信度比较低，所以 OSPF 协议认为从 ASBR 到自治系统之外的开销远远大于在自治系统之内到达 ASBR 的开销。所以计算路由开销时将主要考虑前者，即到第二类外部路由的开销等于 ASBR 到该路由目的地址的开销。如果计算出开销值相等的两条路由，再考虑本路由器到相应的 ASBR 的开销。下面例子 RIP 路由将作为外部路由被重分布到 OSPF 网络中。

I. 拓扑

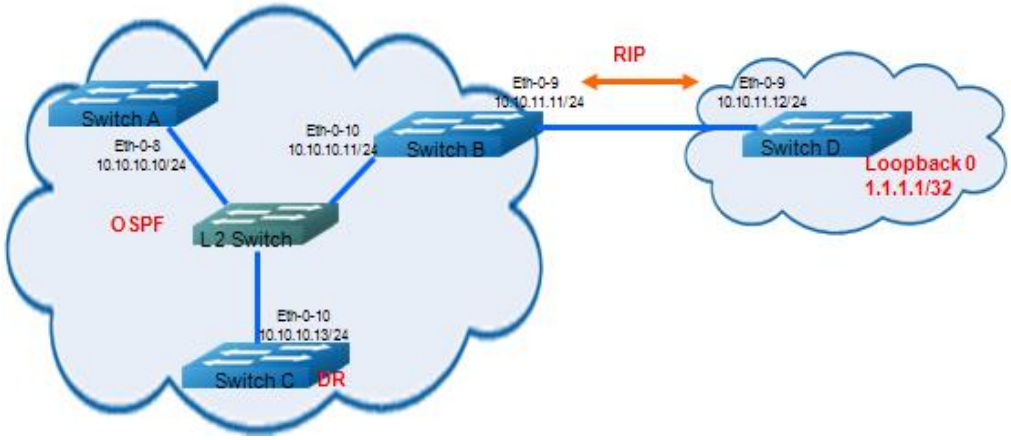


图8-13 OSPF 路由重分布

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-8	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.10.10/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

Switch B

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-10	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.10.11/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式

Switch(config)#interface eth-0-9	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.11.11/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面
Switch(config-router)# redistribute connected	重分布直连路由
Switch(config-router)#redistribute rip	重分布 RIP 路由
Switch(config-router)# exit	回到配置模式
Switch(config)# router rip	创建 RIP 路由
Switch(config-router)# network 10.10.11.0/24	发布网段到 RIP 路由
Switch(config-router)#redistribute connected	重分布直连路由

Switch C

Switch # configure terminal	进入配置模式
Switch(config)#interface eth-0-10	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.10.13/24	设置端口的 IP 地址
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0	发布 10.10.10.0/24 网段到 OSPF 区域 0 里面

Switch D

Switch # configure terminal	进入配置模式
Switch(config)#interface eth-0-9	进入接口模式

Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 10.10.11.12/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router rip	创建 RIP 路由
Switch(config-router)# network 10.10.11.0/24	发布网段到 RIP 路由中
Switch(config-router)# network 1.1.1.1/32	发布网段到 RIP 路由中
Switch(config-router)#redistribute connected	重分布直连路由

III. 命令验证

使用如下命令，验证上述配置：

show ip ospf database externa 和 show ip route

Switch A

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
O E2    1.1.1.1/32 [110/20] via 10.10.10.11, eth-0-8, 00:21:00
C       10.10.10.0/24 is directly connected, eth-0-8
C       10.10.10.10/32 is in local loopback, eth-0-8
O E2    10.10.11.0/24 [110/20] via 10.10.10.11, eth-0-8, 00:13:25
Switch# show ip ospf database external
            OSPF Router with ID (10.10.10.10) (Process ID 100)
                AS External Link States
  LS age: 1447
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: AS-external-LSA
  Link State ID: 1.1.1.1 (External Network Number)
  Advertising Router: 10.10.11.11
  LS Seq Number: 80000002
  Checksum: 0x414e
  Length: 36
  Network Mask: /32
        Metric Type: 2 (Larger than any link state path)
        TOS: 0
        Metric: 20
        Forward Address: 0.0.0.0
        External Route Tag: 0
  LS age: 993
```

```
  Options: 0x2 (*|-|-|-|-|-|E|-)
  LS Type: AS-external-LSA
  Link State ID: 10.10.11.0 (External Network Number)
  Advertising Router: 10.10.11.11
  LS Seq Number: 80000001
  Checksum: 0xfc78
  Length: 36
  Network Mask: /24
        Metric Type: 2 (Larger than any link state path)
        TOS: 0
        Metric: 20
        Forward Address: 0.0.0.0
        External Route Tag: 0
```

Switch B

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
R       1.1.1.1/32 [120/2] via 10.10.11.12, eth-0-9, 00:24:52
C       10.10.10.0/24 is directly connected, eth-0-10
C       10.10.10.11/32 is in local loopback, eth-0-10
C       10.10.11.0/24 is directly connected, eth-0-9
C       10.10.11.11/32 is in local loopback, eth-0-9
```

Switch C

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
O E2    1.1.1.1/32 [110/20] via 10.10.10.11, eth-0-10, 00:22:38
C       10.10.10.0/24 is directly connected, eth-0-10
C       10.10.10.13/32 is in local loopback, eth-0-10
O E2    10.10.11.0/24 [110/20] via 10.10.10.11, eth-0-10, 00:15:04
```

Switch D

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
```

```
        [*] - [AD/Metric]
        * - candidate default
C        1.1.1.1/32 is directly connected, loopback0
R        10.10.10.0/24 [120/2] via 10.10.11.11, eth-0-9, 00:17:36
C        10.10.11.0/24 is directly connected, eth-0-9
C        10.10.11.12/32 is in local loopback, eth-0-9
```

8.3.8 配置 OSPF Cost

你可以通过修改接口的 COST 值来使路由成为最优路由。在下面的例子中，通过修改 COST 值可以使 Switch B 成为 Switch A 的下一跳。

默认接口的 COST 值是 1(1000M speed)。Switch B 的 Eth-0-2 优先级 100，Switch C 的 Eth-0-2 优先级 150. 那么到达 Switch D 的网络 10.10.14.0 的 Cost 值将不一样：

Switch B: 1+1+100 = 102

Switch C: 1+1+150 = 152

I. 拓扑

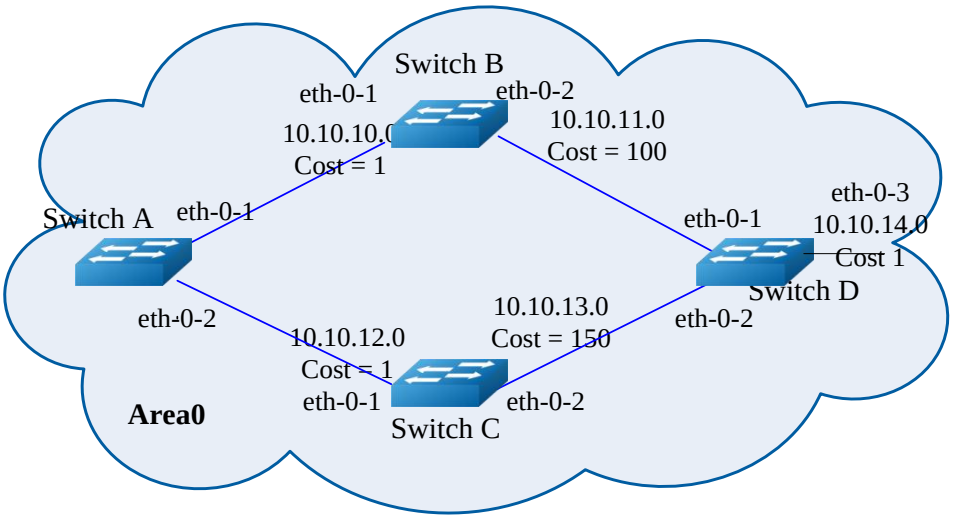


图8-14 OSPF Cost

II. 配置

Switch A

Switch# configure terminal	进入配置模式.
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.10.1/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式

Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.12.1/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0 Switch(config-router)# network 10.10.12.0/24 area 0	发布 10.10.10.0/24, 10.10.12.0/24 网段到 OSPF 区域 0 里面

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.10.2/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.11.2/24	设置端口的 IP 地址
Switch(config-if)# ip ospf cost 100	设置 OSPF 的接口的 COST
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.10.0/24 area 0 Switch(config-router)# network 10.10.11.0/24 area 0	发布 10.10.10.0/24，10.10.11.0/24 网段到 OSPF 区域 0 里面

Switch C

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口

Switch(config-if)# ip address 10.10.12.2/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.13.2/24	设置端口的 IP 地址
Switch(config-if)# ip ospf cost 150	设置 OSPF 的接口的 COST
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100
Switch(config-router)# network 10.10.12.0/24 area 0 Switch(config-router)# network 10.10.13.0/24 area 0	发布 10.10.12.0/24，10.10.13.0/24 网段到 OSPF 区域 0 里面

Switch D

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.11.1/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.13.1/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-3	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ip address 10.10.14.1/24	设置端口的 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf 100	创建 OSPF 进程号 100

Switch(config-router)# network 10.10.11.0/24 area 0 Switch(config-router)# network 10.10.13.0/24 area 0 Switch(config-router)# network 10.10.14.0/24 area 0	发布 10.10.11.0/24，10.10.13.0/24 网段到 OSPF 区域 0 里面

III. 命令验证

使用命令 **show ip ospf route** 验证以上配置。

Switch A

```
Switch# show ip ospf route
OSPF process 0:
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
C  10.10.10.0/24 [1] is directly connected, eth-0-1, Area 0
O  10.10.11.0/24 [101] via 10.10.10.2, eth-0-1, Area 0
C  10.10.12.0/24 [1] is directly connected, eth-0-2, Area 0
O  10.10.13.0/24 [102] via 10.10.10.2, eth-0-1, Area 0
O  10.10.14.0/24 [102] via 10.10.10.2, eth-0-1, Area 0
```

Switch B

```
Switch# show ip ospf route
OSPF process 100:
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
C  10.10.10.0/24 [10] is directly connected, eth-0-1, Area 0
C  10.10.11.0/24 [100] is directly connected, eth-0-2, Area 0
O  10.10.12.0/24 [11] via 10.10.10.1, eth-0-1, Area 0
O  10.10.13.0/24 [101] via 10.10.11.1, eth-0-2, Area 0
O  10.10.14.0/24 [101] via 10.10.11.1, eth-0-2, Area 0
```

Switch C

```
Switch# show ip ospf route
OSPF process 100:
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
O  10.10.10.0/24 [1] via 10.10.12.1, eth-0-1, Area 0
O  10.10.11.0/24 [101] via 10.10.12.1, eth-0-1, Area 0
C  10.10.12.0/24 [1] is directly connected, eth-0-1, Area 0
O  10.10.13.0/24 [102] via 10.10.12.1, eth-0-1, Area 0
O  10.10.14.0/24 [102] via 10.10.12.1, eth-0-1, Area 0
```

Switch D

```
Switch# show ip route
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       [*] - [AD/Metric]
       * - candidate default
O       10.10.10.0/24 [110/1] via 10.10.11.2, eth-0-1, 00:06:27
C       10.10.11.0/24 is directly connected, eth-0-1
O       10.10.12.0/24 [110/1] via 10.10.13.2, eth-0-2, 00:06:17
C       10.10.13.0/24 is directly connected, eth-0-2
C       10.10.14.0/24 is directly connected, eth-0-3
```

8.3.9 配置 OSPF Authentication

系统目前支持三种类型的 OSPF 认证： 无认证（类型 0），明文认证（类型 1）和 MD5 认证（类型 2）。无认证，网络中的路由信息交换不需要经过任何认证。明文认证，所有的路由器上配置的认证模式和密码都必须是一样的。MD5 认证，你需要在每台路由器上配置相同的密钥和密钥 ID。路由器会根据密钥，密钥 ID 和 OSPF 报文内容生成消息摘要加到 OSPF 报文里面。

认证类型可以基于 area 配置，也可以基于 interface 配置，这两者可以同时使用。如果 interface 上配置的认证类型和区域内配置的认证类型不一样，则优先使用 interface 上的认证类型。如果 interface 上没有配置认证类型，那么就使用区域内配置的认证类型。.

下面例子简单介绍了下 OSPF 的三种类型的验证。Switch A 和 Switch B 之间不使用认证；Switch B 和 Switch C 之间使用明文认证；Switch C 和 Switch D 之间使用 MD5 认证。

I. 拓扑

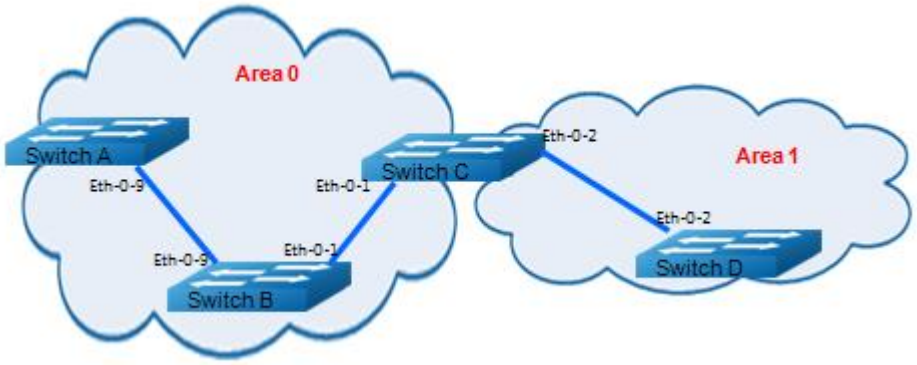


图8-15 OSPF 认证

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-9	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 9.9.9.1/24	设置 IP 地址
Switch(config-if)#ip ospf authentication	接口上启用验证功能
Switch(config-if)#ip ospf authentication null	指定接口验证类型为空
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf	创建 OSPF 进程
Switch(config-router)# network 9.9.9.0/24 area 0	发布网段到 OSPF 中
Switch(config-router)# end	退出路由模式

Switch B

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-9	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 9.9.9.2/24	设置 IP 地址

Switch(config-if)#ip ospf authentication	接口上启用验证功能
Switch(config-if)#ip ospf authentication null	指定接口验证类型为空
Switch(config-if)# exit	退出接口模式
Switch(config)#interface eth-0-1	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 1.1.1.1/24	设置 IP 地址
Switch(config-if)#ip ospf authentication	接口上启用明文验证功能
Switch(config-if)# ip ospf authentication-key test	指定接口认证密码
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf	创建 OSPF 进程
Switch(config-router)# network 9.9.9.0/24 area 0 Switch(config-router)# network 1.1.1.0/24 area 0	发布网段到 OSPF 中
Switch(config-router)# end	退出路由模式

Switch C

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-2	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 2.2.2.1/24	设置 IP 地址
Switch(config-if)# ip ospf message-digest-key 2 md5 ospf	设置接口的 OSPF 验证 KEY
Switch(config-if)# exit	退出接口模式
Switch(config)#interface eth-0-1	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 1.1.1.2/24	设置 IP 地址
Switch(config-if)#ip ospf authentication	接口上启用明文验证功能
Switch(config-if)# ip ospf authentication-key test	设置接口的 OSPF 认证密码
Switch(config-if)# exit	退出接口模式

Switch(config)# router ospf	创建 OSPF 进程
Switch(config-router)# area 1 authentication message-digest Switch(config-router)# network 2.2.2.0/24 area 1 Switch(config-router)# network 1.1.1.0/24 area 0	发布网段到 OSPF 中，配置 area 1 的认证类型为 MD5
Switch(config-router)# end	退出路由模式

Switch D

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-2	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#ip address 2.2.2.2/24	设置 IP 地址
Switch(config-if)# ip ospf message-digest-key 2 md5 ospf	设置接口的 OSPF 验证 KEY
Switch(config-if)# exit	退出接口模式
Switch(config)# router ospf	创建 OSPF 进程
Switch(config-router)# area 1 authentication message-digest Switch(config-router)# network 2.2.2.0/24 area 1	发布网段到 OSPF 中，配置 area 1 的认证类型为 MD5
Switch(config-router)# end	退出路由模式

III. 命令验证

使用 **show ip ospf neighbor** 命令验证上述配置。

Switch A

```
Switch# show ip ospf neighbor
OSPF process 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
9.9.9.2           1   Full/DR         00:00:38    9.9.9.2         eth-0-9
```

Switch B

```
Switch# show ip ospf neighbor
OSPF process 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
2.2.2.1           1   Full/Backup     00:00:35    1.1.1.2         eth-0-1
```

```
1.1.1.1           1   Full/Backup     00:00:38    9.9.9.1         eth-0-9
```

Switch C

```
Switch# show ip ospf neighbor
OSPF process 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
9.9.9.2           1   Full/DR         00:00:35    1.1.1.1         eth-0-1
2.2.2.2           1   Full/DR         00:00:38    2.2.2.2         eth-0-2
Switch# show ip ospf interface
eth-0-1 is up, line protocol is up
  Internet Address 1.1.1.2/24, Area 0, MTU 1500
  Process ID 0, Router ID 2.2.2.1, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State Backup, Priority 1, TE Metric 1
  Designated Router (ID) 9.9.9.2, Interface Address 1.1.1.1
  Backup Designated Router (ID) 2.2.2.1, Interface Address 1.1.1.2
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:01
  Neighbor Count is 1, Adjacent neighbor count is 1
  Crypt Sequence Number is 1301244696
  Hello received 385 sent 384, DD received 3 sent 5
  LS-Req received 1 sent 1, LS-Upd received 11 sent 14
  LS-Ack received 12 sent 10, Discarded 1
  Simple password authentication enabled
Switch# show ip ospf
 Routing Process "ospf 0" with ID 2.2.2.1
 Process uptime is 1 hour 7 minutes
 Process bound to VRF default
 Conforms to RFC2328, and RFC1583 Compatibility flag is disabled
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 This router is an ABR, ABR Type is Alternative Cisco (RFC3509)
 SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
 Refresh timer 10 secs
 Number of incomming current DD exchange neighbors 0/5
 Number of outgoing current DD exchange neighbors 0/5
 Number of external LSA 0. Checksum 0x000000
 Number of opaque AS LSA 0. Checksum 0x000000
 Number of non-default external LSA 0
 External LSA database is unlimited.
 Number of LSA originated 17
 Number of LSA received 57
 Number of areas attached to this router: 2
    Area 0 (BACKBONE)
        Number of interfaces in this area is 1(1)
        Number of fully adjacent neighbors in this area is 1
        Area has no authentication
        SPF algorithm last executed 01:06:56.340 ago
        SPF algorithm executed 16 times
        Number of LSA 6. Checksum 0x034b09
    Area 1
        Number of interfaces in this area is 1(1)
        Number of fully adjacent neighbors in this area is 1
```

```
Number of fully adjacent virtual neighbors through this area is 0
Area has message digest authentication
SPF algorithm last executed 00:03:29.430 ago
SPF algorithm executed 17 times
Number of LSA 5. Checksum 0x0230e3
```

Switch D

```
Switch# show ip ospf neighbor
OSPF process 0:
Neighbor ID     Pri   State           Dead Time   Address         Interface
2.2.2.1           1   Full/Backup     00:00:35    2.2.2.1         eth-0-2
```

8.3.10 配置监听 OSPF

您可以通过命令显示具体的统计数据，如 IP 路由表的内容，缓存和数据库。

Switch# show ip ospf 100	显示 OSPF 进程信息
Switch # show ip ospf 100 database router 10.10.25.21 adv-router 3.3.3.3 Switch # show ip ospf 100 database network self-originate Switch # show ip ospf 100 database summary Switch # show ip ospf 100 database asbr-summary Switch # show ip ospf 100 database external	显示 OSPF 链路状态信息库
Switch # show ip ospf border-routes	显示边界路由器的 OSPF 信息
Switch # show ip ospf interface eth-0-1	显示 OSPF 接口信息
Switch # show ip ospf neighbor 172.16.12.100	显示 OSPF 邻居信息

8.4 Prefix-list 配置

8.4.1 简介

路由策略（Routing Policy）是为了改变网络流量所经过的途径而修改路由信息的技术，主要通过改变路由属性（包括可达性）来实现。地址前缀列表是路由策略的一种，作用比较灵活。一个地址前缀列表由前缀列表名标识。每个前缀列表可以包含多个表项，每个表项可以独立指定一个网络前缀形式的匹配范围，并用一个索引号来标识，索引号指明了进行匹配检查的顺序。在匹配的过程中，交换机按升序依次检查由

索引号标识的各个表项。只要有某一表项满足条件，就意味着本次匹配过程结束，而不再进行下一个表项的匹配。

8.4.2 基础配置

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# ip prefix-list test seq 1 deny 35.0.0.0/8 le 16	创建地址前缀列表 test，并创建一条表项，指定序号为 1
Switch(config)# ip prefix-list test permit any	创建一个表项为了防止不匹配条目出现时候遭遇拒绝
Switch(config)# ip prefix-list test description this prefix list is fot test	添加地址前缀列表描述
Switch(config)# ip prefix-list test permit 36.0.0.0/24	创建一条表项，使用默认序号
Switch(config)# exit	退出全局模式

II. 命令验证

Switch# show ip prefix-list detail

```
Prefix-list list number: 1
Prefix-list entry number: 3
Prefix-list with the last deletion/insertion: test
ip prefix-list test:
   Description: this prefix list is fot test
   count: 3, range entries: 0, sequences: 1 - 10
     seq 1 deny 35.0.0.0/8 le 16 (hit count: 0, refcount: 0)
     seq 5 permit any (hit count: 0, refcount: 0)
     seq 10 permit 36.0.0.0/24 (hit count: 0, refcount: 0)
```

8.4.3 配置 Rip 简单应用

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# ip prefix-list aa seq 11 deny 35.0.0.0/8 le 16	创建地址前缀列表 aa，并创建一条表项
Switch(config)# ip prefix-list aa permit any	创建一个表项为了防止不匹配条目出现时候遭遇拒绝
Switch(config)# router rip	进入 Rip 路由模式

Switch(config-router)# distribute-list prefix aa out	应用策略
Switch(config-router)# end	退出 Rip 路由模式

II. 命令验证

Switch# show ip prefix-list

```
ip prefix-list aa: 2 entries
      seq 11 deny 35.0.0.0/8 le 16
      seq 15 permit any
Switch# show running-config
Building configuration...
…
ip prefix-list aa seq 11 deny 35.0.0.0/8 le 16
ip prefix-list aa seq 15 permit any
…
router rip
 distribute-list prefix aa out
```

8.4.4 配置 Route-map 简单应用

I. 配置 prefix-list 应用到 route-map

Switch# configure terminal	进入配置模式
Switch(config)# ip prefix-list aa seq 11 deny 3.3.3.0/8 le 24	创建地址前缀列表 aa，并创建一条表项
Switch(config)# ip prefix-list aa permit any	创建一个表项为了防止不匹配条目出现时候遭遇拒绝
Switch(config)# route-map abc permit	创建 route-map
Switch(config-route-map)# match ip address prefix-list aa	匹配地址前缀列表 aa
Switch(config-route-map)# set local-preference 200	设置行为
Switch(config-route-map)# exit	退出路由模式
Switch(config)# route-map abc permit 20	重新定义一个策略为了防止不匹配条目出现时候遭遇拒绝
Switch(config-route-map)# exit	退出路由模式
Switch(config)# router bgp 1	进入 BGP 路由模式
Switch(config-router)# neighbor 1.1.1.2 remote-as 1	配置 BGP 邻居

Switch(config-router)# neighbor 1.1.1.2 route-map abc out	在 BGP 路由上应用路由策略
Switch(config-router)# network 2.2.2.2/32	BGP 路由中宣告网段
Switch(config-router)# network 3.3.3.3/32	BGP 路由中宣告网段
Switch(config-router)# end	退出 BGP 模式

II. 命令验证

Switch # show route-map

```
route-map abc, permit, sequence 10
  Match clauses:
    ip address prefix-list aa
  Set clauses:
    local-preference 200
route-map abc, permit, sequence 20
  Match clauses:
Set clauses:
Switch # show running-config
Building configuration...
…
ip prefix-list aa seq 11 deny 3.3.3.0/8 le 24
ip prefix-list aa seq 15 permit any
!
!
route-map abc permit 10
 match ip address prefix-list aa
 set local-preference 200
!
route-map abc permit 20
…
router bgp 1
 neighbor 1.1.1.2 remote-as 1
 !
 address-family ipv4
 no synchronization
 network 2.2.2.2 mask 255.255.255.255
 network 3.3.3.3 mask 255.255.255.255
 neighbor 1.1.1.2 activate
 neighbor 1.1.1.2 route-map abc out
 exit-address-family
 !
 address-family vpnv4 unicast
 no synchronization
 exit-address-family
```

8.5 Route-map 配置

8.5.1 简介

路由策略（Routing Policy）是为了改变网络流量所经过的途径而修改路由信息的技术，主要通过改变路由属性（包括可达性）来实现。

路由器在发布与接收路由信息时，可能需要实施一些策略，以便对路由信息进行过滤，例如只接收或发布满足一定条件的路由信息。一种路由协议可能需要引入其它的路由协议发现的路由信息，路由器在引入其它路由协议的路由信息时，可能只需要引入一部分满足条件的路由信息，并控制所引入的路由信息的某些属性，以使其满足本协议的要求。为实现路由策略，首先要定义将要实施路由策略的路由信息的特征，即定义一组匹配规则。可以以路由信息中的不同属性作为匹配依据进行设置，如目的地址、发布路由信息的路由器地址等。匹配规则可以预先设置好，然后再将它们应用于路由的发布、接收和引入等过程的路由策略中。

8.5.2 配置 route-map 应用到 OSPF

I. 配置

DUT# configure terminal	进入配置模式
DUT(config)# route-map abc permit	创建一个路由策略
DUT(config-route-map)# match metric 20	设置规则
DUT(config-route-map)# set tag 2	设置行为
DUT(config-route-map)# exit	退出策略模式
DUT(config)# route-map abc permit 20	重新定义一个策略为了防止不匹配条目出现时候遭遇拒绝
DUT(config-route-map)# exit	退出策略模式
DUT(config)# router ospf 100	进入 OSPF 路由模式
DUT(config-router)# redistribute rip route-map abc	把 RIP 协议重分布到 OSPF 中，并且使用策略 abc
DUT(config-router)# end	退出 OSPF 路由模式

II. 命令验证

Switch# show route-map

```
route-map abc, permit, sequence 10
  Match clauses:
    metric 20
  Set clauses:
```

```
    tag 2
route-map abc, permit, sequence 20
  Match clauses:
  Set clauses:
```

8.5.3 配置 route-map 应用到 BGP

I. 配置

命令	描述
DUT# configure terminal	进入配置模式
DUT(config)# ip access-list acl1	创建一个 ACL
DUT(config-ip-acl)# permit any 3.3.3.0 0.0.0.255 any	设置匹配的条目
DUT(config-ip-acl)# exit	退出 ACL 模式
DUT(config)# route-map abc permit	创建路由策略
DUT(config-route-map)# match ip address acl1	匹配 ACL
DUT(config-route-map)# set local-preference 200	设置行为
DUT(config-route-map)# exit	退出路由模式
DUT(config)# route-map abc permit 20	重新定义一个策略为了防止不匹配条目出现时候遭遇拒绝
DUT(config-route-map)# exit	退出路由模式
DUT(config)# router bgp 1	进入 BGP 路由模式
DUT(config-router)# neighbor 1.1.1.2 remote-as 1	配置 BGP 邻居
DUT(config-router)# neighbor 1.1.1.2 route-map abc out	在 BGP 路由上应用路由策略
DUT(config-router)# network 2.2.2.2/32	BGP 路由中宣告网段
DUT(config-router)# network 3.3.3.3/32	BGP 路由中宣告网段
DUT(config-router)# end	退出 BGP 模式

II. 命令验证

DUT1# show route-map

```
route-map abc, permit, sequence 10
  Match clauses:
    ip address acl1
  Set clauses:
    local-preference 200
route-map abc, permit, sequence 20
  Match clauses:
  Set clauses:
DUT2# show ip bgp
BGP table version is 6, local router ID is 1.1.1.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
   Network          Next Hop            Metric LocPrf Weight Path
*>i2.2.2.2/32       1.1.1.1                  0    100      0 i
*>i3.3.3.3/32       1.1.1.1                  0    200      0 i
```

8.6 策略路由(PBR) 配置

8.6.1 简介

与单纯根据 IP 报文的目的地址进行转发不同，策略路由是一种根据用户制定的策略进行路由转发的机制。

8.6.2 拓扑

下图是策略路由的一个典型配置：你可以在 Switch 的 eth-0-1 端口上应用一个 PBR，源地址是 172.16.6.1 的报文将会被转发给 Lucy, 源地址是 172.16.7.1 将会进行正常的路由转发。

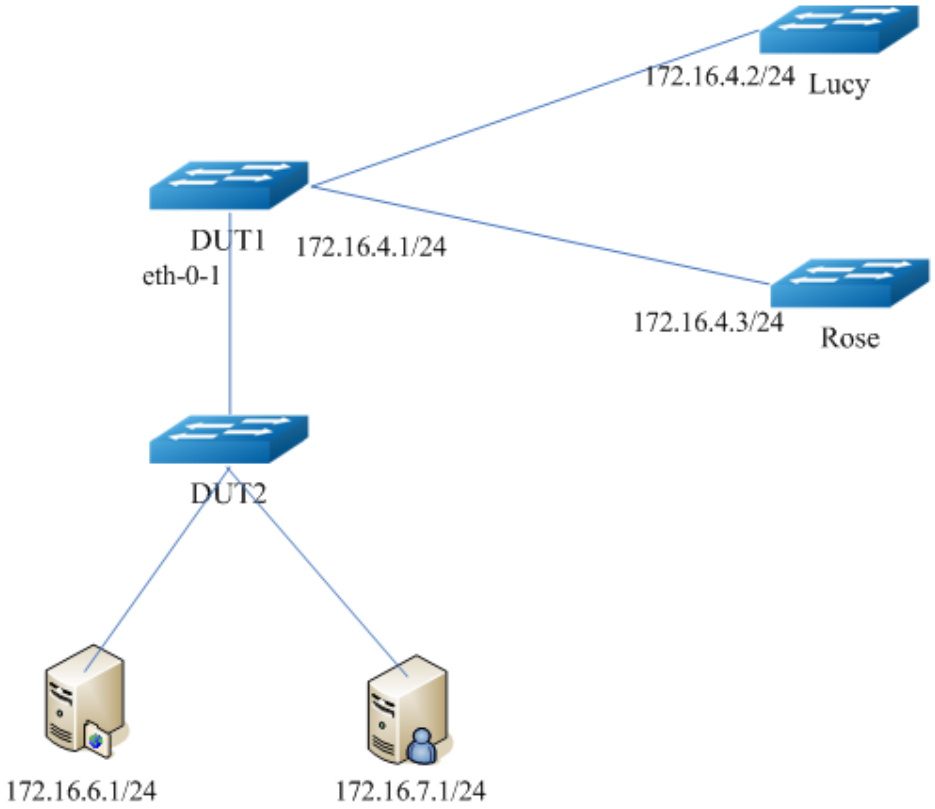


图8-16 典型的 PBR 拓扑

8.6.3 配置

DUT# configure terminal	进入配置模式
Switch(config)# ip access-list acl1	定义一个 IPV4 ACL 并且进入 ACL 配置模式
Switch(config-ip-acl)# 10 permit any 172.16.6.0 0.0.0.255 any	配置一个允许源地址为 172.16.6.0 的报文进入的 ACE
Switch(config-ip-acl)# exit	退出 ACL 配置模式
Switch(config)# route-map richard permit 10	创建一个名为 Richard 的 route-map 并且进入 route-map 配置模式
Switch(config-route-map)# match ip address acl1	配置一个匹配 acl1 的 match 语句
Switch(config-route-map)# set ip next-hop 172.16.4.2	设置满足匹配条件的数据包的转发地址为 172.16.4.2

Switch(config-route-map)# exit	退出 Route-map 配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将端口设置为三层口
Switch(config-if)# ip address 172.16.5.2/24	设置接口的 IP 地址
Switch(config-if)# no shutdown	启用这个接口
Switch(config-if)# ip policy route-map richard	在这个接口上应用 richard 这个策略
Switch(config-if)# exit	退出接口配置模式

8.6.4 命令验证

Switch# show ip policy route-map

```
Route-map                        interface
richard                           eth-0-1
```

8.7 BGP 配置

8.7.1 简介

边界网关协议（BGP）是一个内部自治系统路由协议。

BGP 通告系统的主要功能，是用其他的 BGP 系统来交换网络上的可达信息。这个网络可达信息包括自治系统（AS）中的可达性信息。这个信息，对构建一个可联通的 AS（如果出现路由环路的话会被切断，以及在这个 AS 级别中，有些策略会被强制执行）来说，是足够的。

BGP-4 提供了一组机制，来支持无类域内路由（CIDR）[RFC1518, RFC1519]. 这些机制包括发布一组 IP 前缀的目的地址，已经消除 BGP 中“类”的概念。BGP-4 也引入了一些允许路由集合（包括 AS 路径的集合）这样的概念。

被 BGP 交换的路由信息只支持基于目的的范例，假定路由器只通过在 IP 报文头中的目的地址来转发报文。这样，反过来说，反而导致了这些策略决策是否可以被强制的使用 BGP。BGP 可以支持那些基于目的地址转发的策略。

更多的 BGP 信息请参考[RFC 1771, RFC 4271].

8.7.2 基本拓扑 Topology (EBGP)

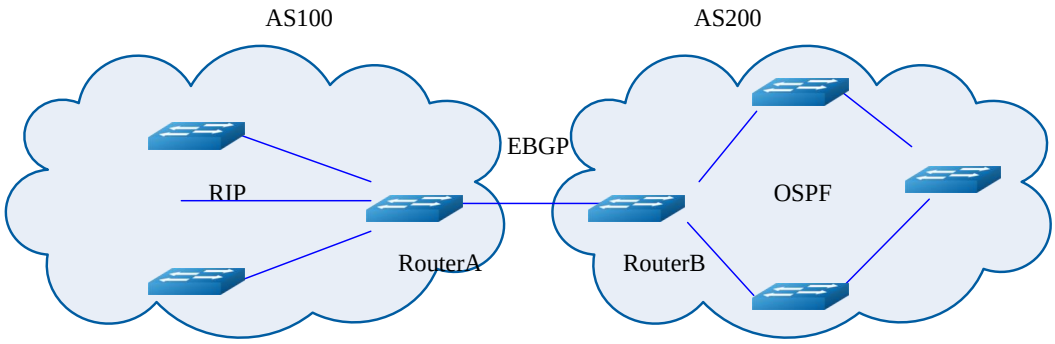


图8-17 EBGP 拓扑

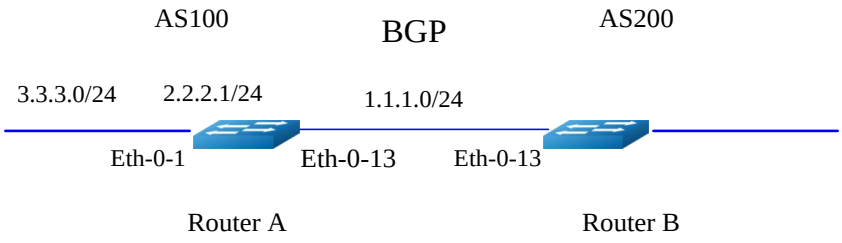


图8-18 EBGP 拓扑

II. 配置

Router A

Switch #configure terminal	进入配置模式。
Switch (config)#interface eth-0-13	进入接口模式。
Switch (config-if)#no shutdown	启用端口。
Switch (config-if) # no switchport	将该端口转换为三层 interface。
Switch (config-if) # ip address 1.1.1.1/24	配置 IP 地址为 1.1.1.1/24。
Switch (config-if)#exit	退出接口模式并且进入配置模式。
Switch (config)#interface eth-0-1	进入接口模式。
Switch (config-if)# no shutdown	启用端口。
Switch (config-if) # no switchport	将该端口转换为三层 interface。
Switch (config-if) # ip address 2.2.2.1/24	配置 IP 地址为 2.2.2.1/24。

Switch (config-if)#exit	退出接口模式并且进入配置模式。
Switch (config)#ip route 3.3.3.0/24 2.2.2.2	增加一条静态路由。
Switch (config)#router bgp 100	创建 BGP 100 并进入路由模式。
Switch (config-router)#bgp router-id 10.10.10.10	配置 BGP router-id。
Switch (config-router)#neighbor 1.1.1.2 remote-as 200	配置 EBGP 邻居号 200。
Switch (config)# neighbor 1.1.1.2 ebgp-multihop	配置邻居为 ebgp-multihop。
Switch (config-router)#network 4.0.0.0/8	宣告网络号。
Switch (config-router)#redistribute static	重分布静态路由到 BGP。
Switch (config-router)#redistribute connected	重分布直连路由到 BGP。
Switch (config-router)#exit	退出路由模式并且进入配置模式。

Router B

Switch #configure terminal	进入配置模式。
Switch (config)#interface eth-0-13	进入接口模式。
Switch (config-if)#no shutdown	启用端口。
Switch (config-if) # no switchport	将该端口转换为三层 interface。
Switch (config-if) # ip address 1.1.1.2/24	配置 IP 地址为 1.1.1.2/24。
Switch (config-if)#exit	退出接口模式并且进入配置模式。
Switch (config)#router bgp 200	创建 BGP 200 并进入路由模式。
Switch (config-router)#bgp router-id 11.11.11.11	配置 BGP router-id。
Switch (config-router)#neighbor 1.1.1.1 remote-as 100	配置 EBGP 邻居号 100。
Switch (config)# neighbor 1.1.1.1 ebgp-multihop	配置邻居为 ebgp-multihop。

Switch (config-router)#redistribute connected	重分布直连路由到 BGP。
Switch (config-router)#exit	退出路由模式并且进入配置模式。

III. 试验结果

SwitchA# show ip bgp neighbors

```
BGP neighbor is 1.1.1.2, remote AS 200, local AS 100, external link
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Active
  Last read 00:26:00, hold time is 180, keepalive interval is 60 seconds
  Received 0 messages, 0 notifications, 0 in queue
  Sent 0 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes
 Connections established 0; dropped 0
  External BGP neighbor may be up to 255 hops away.
```

Next connect timer due in 87 seconds

```
SwitchB# show ip bgp neighbors
BGP neighbor is 1.1.1.1, remote AS 100, local AS 200, external link
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Active
  Last read 00:21:39, hold time is 180, keepalive interval is 60 seconds
  Received 0 messages, 0 notifications, 0 in queue
  Sent 0 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 30 seconds
 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes
 Connections established 0; dropped 0
  External BGP neighbor may be up to 255 hops away.
Next connect timer due in 97 seconds
```

8.7.3 基本拓扑(IBGP)

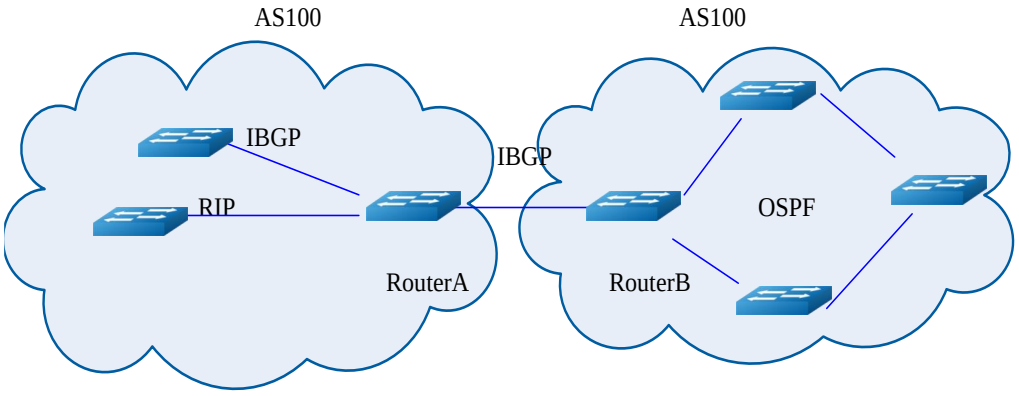


图8-19 IBGP

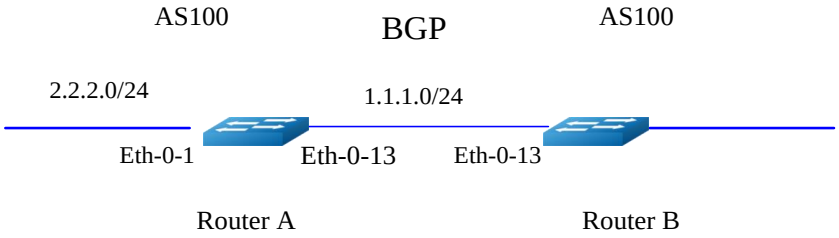


图8-20 IBGP 拓扑

II. 配置

Router A

Switch #configure terminal	进入配置模式。
Switch (config)#interface eth-0-13	进入接口模式。
Switch (config-if)#no shutdown	启用端口。
Switch (config-if) # no switchport	将该端口转换为三层 interface。
Switch (config-if) # ip address 1.1.1.1/24	配置 IP 地址为 1.1.1.1/24。
Switch (config-if)#exit	退出接口模式并且进入配置模式。
Switch (config)#interface loopback 0	进入接口模式。
Switch (config-if) # ip address 10.10.10.10/32	配置 IP 地址为 10.10.10.10/32。
Switch (config-if)#exit	退出接口模式并且进入配置模式。

Switch (config)# ip route 11.11.11.11/32 1.1.1.2	增加一条静态路由。
Switch (config)#interface eth-0-1	进入接口模式。
Switch (config-if)# no shutdown	启用端口
Switch (config-if) # no switchport	将该端口转换为三层 interface。
Switch (config-if) # ip address 2.2.2.1/24	配置 IP 地址为 2.2.2.1/24。
Switch (config-if)#exit	退出接口模式并且进入配置模式。
Switch (config)#ip route 3.3.3.0/24 2.2.2.2	增加一条静态路由。
Switch (config)#router bgp 100	创建 BGP 100 并进入路由模式。
Switch (config-router)#bgp router-id 10.10.10.10	配置 BGP router-id。
Switch (config-router)#neighbor 11.11.11.11 remote-as 100	配置 IBGP 邻居 AS 号 100。
Switch (config-router)#neighbor 11.11.11.11 update-source loopback 0	配置 loopback0 为更新源端口。
Switch (config-router)#network 4.0.0.0/8	宣告网络号。
Switch (config-router)#redistribute static	重分布静态路由到 BGP。
Switch (config-router)#redistribute connected	重分布直连路由到 BGP。
Switch (config-router)#exit	退出路由模式并且进入配置模式。

Router B

Switch #configure terminal	进入配置模式。
Switch (config)#interface eth-0-13	进入接口模式。
Switch (config-if)#no shutdown	启用端口。
Switch (config-if) # no switchport	将该端口转换为三层 interface。
Switch (config-if) # ip address 1.1.1.2/24	配置 IP 地址为 1.1.1.2/24。
Switch (config-if)#exit	退出接口模式并且进入配置模式。

Switch (config)#interface loopback 0	进入接口模式。
Switch (config-if) # ip address 11.11.11.11/32	配置 IP 地址为 11.11.11.11/32。
Switch (config-if)#exit	退出接口模式并且进入配置模式。
Switch (config)# ip route 10.10.10.10/32 1.1.1.1	增加一条静态路由。
Switch (config)#router bgp 100	创建 BGP 100 并进入路由模式。
Switch (config-router)#bgp router-id 11.11.11.11	配置 BGP router-id。
Switch (config-router)#neighbor 10.10.10.10 remote-as 100	配置 IBGP 邻居 AS 号 100。
Switch (config-router)#neighbor 10.10.10.10 update-source loopback 0	配置 loopback0 为更新源端口。
Switch (config-router)#redistribute connected	重分布直连路由到 BGP。
Switch (config-router)#exit	退出路由模式并且进入配置模式。

III. 试验结果

SwitchA# show ip bgp neighbors

```
BGP neighbor is 11.11.11.11, remote AS 100, local AS 100, internal link
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Active
  Last read 00:02:32, hold time is 180, keepalive interval is 60 seconds
  Received 0 messages, 0 notifications, 0 in queue
  Sent 0 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 5 seconds
  Update source is loopback0
 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes
 Connections established 0; dropped 0
Next connect timer due in 62 seconds
SwitchB# show ip bgp neighbors
BGP neighbor is 10.10.10.10, remote AS 100, local AS 100, internal link
  BGP version 4, remote router ID 0.0.0.0
  BGP state = Active
  Last read 00:01:58, hold time is 180, keepalive interval is 60 seconds
  Received 0 messages, 0 notifications, 0 in queue
```

```
  Sent 0 messages, 0 notifications, 0 in queue
  Route refresh request: received 0, sent 0
  Minimum time between advertisement runs is 5 seconds
  Update source is loopback0
 For address family: IPv4 Unicast
  BGP table version 1, neighbor version 0
  Index 1, Offset 0, Mask 0x2
  0 accepted prefixes
  0 announced prefixes
 Connections established 0; dropped 0
Next connect timer due in 17 seconds
```

9 流量管理配置指导

9.1 QoS 配置

9.1.1 简介

QoS（Quality of Service，服务质量）是各种存在服务供需关系的场合中普遍存在的概念，它评估服务方满足客户服务需求的能力。评估通常不是精确的评分，而是注重分析在什么条件下服务是好的，在什么情况下还存在着不足，以便有针对性地作出改进。在因特网中，QoS 所评估的就是网络投递分组的服务能力。由于网络提供的服务是多样的，因此对 QoS 的评估可以基于不同方面。通常所说的 QoS，是对分组投递过程中为延迟、延迟抖动、丢包率等核心需求提供支持的服务能力的评估。QoS 是网络的一种安全机制，是用来解决网络延迟和阻塞等问题的一种技术。在正常情况下，如果网络只用于特定的无时间限制的应用系统，并不需要 QoS，比如 Web 应用，或 E-mail 设置等。但是对关键应用和多媒体应用就十分必要。当网络过载或拥塞时，QoS 能确保重要业务量不受延迟或丢弃，同时保证网络的高效运行。

9.1.2 术语

以下是用来形容 QoS 的术语和概念的简要描述：

访问控制列表（ACLs）

具有相同特征的流量进行分类。IP ACL 用来分类 IP 流量，MAC ACL 用来分类除了 IPV6 和 MPLS 以外的所有流量。

服务种类（CoS）

在网络的第 2 层中确定报文优先级的字段。QOS 可以通过设置不同的 COS 值来区分不同优先级的流量。802.1Q 二层报文中可以携带 2 字节的 VLAN 标签，最高的 3 个比特用于用户指定的优先级。其它类型的报文不能携带 VLAN 标签。CoS 有 3 个比特，其值的范围为 0-7。

差分服务代码点（DSCP）

有 6 个比特位，用来区分三层网络中的报文的优先级。DSCP 值范围是 0-63。

IP-Precedence

有 3 个比特位，用来区分三层网络中报文的优先级。IP-Precedence 范围是 0-7。

EXP

有 3 个比特位，用来区分 MPLS 网络中的报文的优先级。MPLS EXP 值范围在 0-7。

流分类（Traffic Classification）

指采用一定的规则识别出符合某类特征的报文。分类规则（classification rule）是用户根据管理需求配置的过滤规则。报文进入系统时，流分类处理引擎会为报文分配一个内部优先级，基于这个优先级，系统对报文进行一系列的处理。系统可以基于报文中的 CoS、inner-CoS、DSCP、IP-Precedence，或者端口上的配置的默认 CoS，或者依据 policy-map 配置映射出的内部优先级。

流量整形（Shaping）

是通过缓存报文来改变并调节入方向的流速率，从而使出方向的流速率表现地更加平滑的一种方法。当入方向的流量出现高突发的时候，就需要将报文缓存并在后面发送，从而使出方向的流更加平滑，因此 shaping 可能会增加报文的抖动。

流量整形可以应用在以下角色：

- 物理接口（port shaping）
- 出方向的队列（queue shaping）

当 queue 应用双速率的 shaping 时，需要保证该接口下所有 queue 的 CIR 之和不大于端口速率，并且不大于接口 shaping 的速率。

流量监管（Policing）

会对流量进行测速，从而决定报文是保证速率内还是保证速率外，保证速率外的流量可能会被丢弃。

系统支持两种类型的 policer：

- 配置在 class-map 中，用于对匹配某个 class-map 的流量限制带宽。
- 配置一个聚合 policer，用户可以将匹配 class-map 的流量加入这个 policer 中。聚合 policer 限制的是其中所有的流量的带宽。

标记（Marking）

定义了对超出保证速率的流量的处理行为。系统采用两种行为中的一种：给报文标记颜色，后面会继续处理；直接丢弃报文。

标记能够在进口和出口方向使用。

Queueing

Basic 和 enterprise 模式下下每个出端口有 8 个队列，范围 0-7，优先级最高是 7，最低是 0。Enterprise advance 模式下每个出端口有 12 个队列，范围 0-11，其中 0-7 是单播流量队列，8-11 是组播流量队列，单播流量队列中优先级最高是 7，最低是 0。组播流量队列中优先级最高的是 11，最低是 8。每个 queue 中支持 3 个丢弃优先级。队列长度的单位是 buffer cell。Buffer Cell 是报文的存储粒度单位，其大小为 256 字节，报文越大，占用的 buffer cell 越多。

Tail Drop

是一种简单的丢弃算法，即队列中报文达到一定阈值（可配置）时，后来的报文会被丢弃。默认情况下，端口上的丢弃算法就是 Tail Drop。系统支持在每个端口上为每个 queue 每种丢弃优先级制定一个 Tail Drop 的阈值。

WRED（Weighted Random Early Detection）

WRED（Weighted Random Early Detection）可以提前以一定概率丢弃报文，达到避免拥塞的目的，通过提前丢弃报文，WRED 模式可以避免短时间内丢弃大量报文，导致大量 TCP 连接同时触发慢启动和拥塞退避，网络带宽利用率瞬间降低的现象。系统支持在端口上为每个 queue 每种丢弃优先级制定两个阈值。这两个阈值前者小于后者。当队列中报文达到前者时报文开始丢弃，队列中报文越多，丢弃概率越大。当队列中报文大于后一阈值时，报文全部丢弃。

Scheduling

系统为每个队列分配一个优先级（class），范围是 0 到 7，数字越高表示优先级越高。Basic 模式下端口上的 8 个队列的优先级是可配置的。

QoS 启用，basic 模式下队列 0 到 7 对应的优先级为：0/1/2/3/4/5/6/7；enterprise 模式下队列 0 到 7 对应的优先级为：3/3/4/4/4/4/5/7；enterprise advance 模式下队列 0 到 11 对应的优先级为：3/3/4/4/4/4/5/7/0/1/2/3。

QoS 禁用，所有队列的优先级均为 0。

一个端口上，不同的优先级之间使用的 SP 调度，即高优先级队列先被调度，当高优先级队列为空时才会调度低优先级队列。相同的优先级内的队列采用 WDRR 调度。用户可以为各个队列设置权重。

QoS 启用，basic 模式下队列 0 到 7 对应的 WDRR 权重为：1::1:1:1:1:1:1:1；enterprise 模式下队列 0 到 7 对应的 WDRR 权重为：1:1:4:10:10:10:1:1；enterprise advance 模式下队列 0 到 11 对应的 WDRR 权重为：1:1:4:10:10:10:1:1:1:1:1:1

Class Map

通过指定一些 ACL 定义一组流。这些 ACL 可以是 match-all 或 match-any 的，分别表示流量要同时匹配所有的 ACL 或匹配任意的 ACL。

Policy Map

用来指定不同种类流量的具体行为，可实现如下需求：

- 将流按照指定的优先级和颜色区分出来
- 为相应的优先级和颜色设置指定的信任策略
- 为满足某个信任策略的流按照预先的配置做流量监管
- 为指定的流做重定向
- 为指定的流做镜像
- 为指定的流做统计

Policy Map 有如下属性：

- 一个 Policy Map 可以包含多个流分类定义，并给出单独的行为
- 每一个流分类定义可以匹配接口上的每一种流量
- 每一个端口的每一个方向只能应用一个 Policy Map。相同的 Policy Map 可以在不同端口的不同方向上应用。
- 如果要使得 Policy Map 生效，其必须被附加到一个端口上。
- 一个 Policy Map 可以应用于物理接口（非聚合端口成员），聚合端口以及 VLAN 接口。

Mapping Tables

在 QoS 处理中，交换机将所有流量都映射到内部优先级处理。

- 在流分类时，QoS 使用可配置的映射表进行报文映射，内部优先级共 6 个比特，是从 CoS、EXP、DSCP、IP-Precedence 的值映射而来，这些映射表包含了 CoS-Priority-Color/COS-PHB 表、EXP-Priority-Color/EXP-PHB 表、DSCP-Priority-Color/DSCP-PHB 表和 IP-Precedence-Priority-Color/IP-PREC-PHB。
- 在流量监管时，QoS 给报文分配一个新的优先级和颜色，比如依据 Class-Map。
- 当流量结束调度阶段后，如果替换 CoS 或者 DSCP 被置起来，那么 QoS 使用 Priority-Color-Cos/PHB-COS 或者 Priority-Color-DSCP/PHB-DSCP 根据内部的优先级和颜色重新映射到 CoS 或者 DSCP
- 每一个 QoS 域的上述行为都是不同的

Time-range

通过使用 Time-Range，Class-Map 的行为可以按照每周的特定时间来启用或者禁用。首先，定义 Time-Range 的名字并设置其在一周内的时间，然后将其应用到 ACE。可以使用 Time-Range 来制定 Class-Map 中独立的一条 ACE 在每周的制定时间生效。

RTCM

单速率三色标记（Single Rate Three Color Marker）

TRTCM

双速率三色标记（Two Rate Three Color Marker）

CIR

提交信息速率（Committed Information Rate）

CBS

提交组量大小（Committed Burst Size）

EBS

超量组量大小（Excess Burst Size）

PIR

峰值信息速率（Peak Information Rate）

9.1.3 模块化的 QoS 命令行

入口流量应用 QoS 策略进行分类。

class-map 类 QoS

用于定义一组流，定义的规则有 CoS/DSCP/IP Precedence/EXP/ACL。

policy-map 类 QoS

用于划分流类型，相同类型的 policy-map 关联同一个 class-map 类 QoS。

class-map 类流优先级

用于定义流优先级，定义规则是流优先级。

policy-map 类流优先级

用于划分 QoS 流量监管，相同类型的 policy-map 关联同一个 class-map 类流分类。

9.1.4 配置指导

配置 QOS 之前需知以下信息：

- QoS policing 不能在 LinkAGG 上配置。
- 只能在进口方向进行分类。
- Class map 可以有多个 ACL，一个 ACL 可以有多个条目。
- Policing 不能在交换机的虚拟接口上使用。

9.1.5 拓扑

图9-1 交换机

9.1.6 配置

I. 配置出口队列

Tail Drop

尾丢弃是默认的每个出口队列拥塞避免技术。在没有超过 queue 长度的时候，报文会在 queue 中缓存。

下面例子说明了如何根据不同的丢弃优先级配置尾丢弃阈值。

- configure terminal；
- 创建 class-map 类流优先级，并配置优先级；
- 创建 policy-map 类流优先级，并关联之前定义的 class-map；
- 在 policy-map 优先级模式下，设置该优先级的尾丢弃上限值；
- interface IFNAME 进入匹配相应策略表的接口，其中 IFNAME 是该接口的名称。

下面是对流优先级为 3 的尾丢弃上限的配置实例，例子中的尾丢弃上限为 2000.

表9-1 配置 Tail Drop

Switch# configure terminal	进入全局配置模式
Switch(config)# class-map type traffic-class tc3	创建 class-map 并进入其配置模式
Switch(config-cmap-tc)# match traffic-class 3	设置流优先级为 3
Switch(config-cmap-tc)# exit	退出该配置模式
Switch(config)# policy-map type traffic-class tc	创建 policy-map 并进入其配置模式
Switch(config-pmap-tc)# class type traffic-class tc3	关联 class-map

Switch(config-pmap-tc-c)# queue-limit 2000	配置丢包上限 2000
Switch(config-pmap-tc-c)# exit	退出该配置模式
Switch(config-pmap-tc)# exit	退出到全局配置模式
Switch(config)# interface eth-0-1	进入端口配置模式
Switch(config-if)# service-policy type traffic-class tc	应用 QoS policy
Switch(config-if)# end	退出全局配置模式
Switch# show qos interface eth-0-1 egress	显示 QoS 配置

验证配置

Switch# show qos interface eth-0-1 egress

```
TC  Priority  Bandwidth  Shaping(kbps)  Drop-Mode   Max-Queue-Limit(Cell)   ECN
0   0         -          -              dynamic     level 0                 -
1   0         -          -              dynamic     level 0                 -
2   0         -          -              dynamic     level 0                 -
3   0         -          -              tail-drop   2000                    2000
4   0         -          -              dynamic     level 0                 -
5   0         -          -              dynamic     level 0                 -
6   0         -          -              dynamic     level 0                 -
7   7         -          -              tail-drop   64                      -
```

WRED

WRED 通过选择性地丢弃部分报文，降低接口拥塞时发生尾丢弃的概率。通过早期选择性地丢弃部分报文而不是在 queue 真正满时才开始丢弃，WRED 可以避免出现 TCP 同步丢包的问题，从而提高网络的吞吐量。

下面例子说明了如何针对不同颜色的报文配置相应的 WRED 阈值。

- configure terminal
- 创建 class-map 类流优先级，并配置优先级；
- 创建 policy-map 类流优先级，并关联之前定义的 class-map；
- 在 policy-map 优先级模式下配置对应流优先级的 WRED 丢包上限；
- interface IFNAME 进入匹配相应策略表的接口，其中 IFNAME 是该接口的名称

下面的例子所示的是对流优先级为 1 设定其 WRED 丢包上限。其最大丢包上限为 596，最小丢包上限为 596/8=71。如果缓冲区中的报文超过最小丢包上限，后续收到的报文将随机丢弃。

表9-2 配置 WRED

Switch# configure terminal	进入全局配置模式
Switch(config)# class-map type traffic-class tc1	创建 class-map 并进入其配置模式
Switch(config-cmap-tc)# match traffic-class 1	设置流优先级为 1
Switch(config-cmap-tc)# exit	退出该配置模式
Switch(config)# policy-map type traffic-class tc	创建 policy-map 并进入其配置模式
Switch(config-pmap-tc)# class type traffic-class tc1	关联 class-map
Switch(config-pmap-tc-c)# random-detect maximum-threshold 596	配置丢包上限 596
Switch(config-pmap-tc-c)# exit	退出该配置模式
Switch(config-pmap-tc)# exit	退出到全局配置模式
Switch(config)# interface eth-0-1	进入端口配置模式
Switch(config-if)# service-policy type traffic-class tc	应用 QoS policy
Switch(config-if)# end	退出全局配置模式
Switch# show qos interface eth-0-1 egress	显示 QoS 配置

验证配置

Switch# show qos interface eth-0-1 egress

```
TC  Priority  Bandwidth  Shaping(kbps)  Drop-Mode    Max-Queue-Limit(Cell)    ECN
0   0         -          -              dynamic      level 0                  -
1   0         -          -              random-drop  596                      Disable
2   0         -          -              dynamic      level 0                  -
3   0         -          -              tail-drop    2000                     2000
4   0         -          -              dynamic      level 0                  -
5   0         -          -              dynamic      level 0                  -
6   0         -          -              dynamic      level 0                  -
7   7         -          -              tail-drop    64                       -
```

Schedule

在不同的 CLASS 之间，报文是按照 SP（严格优先级）调度的；在同一个 CLASS 之间，报文是按照 WDRR 调度的。

下面的例子显示了将 queue 映射到不同的 CLASS 中间并且配置 WDRR 调度的权重。

- configure terminal

- 创建 class-map 类流优先级，并配置优先级；
- 创建 policy-map 类流优先级，并关联之前定义的 class-map；
- 在 policy-map 优先级模式下配置对应流优先级的调度优先级；
- 在 policy-map 优先级模式下配置对应流优先级的带宽；
- interface IFNAME 进入匹配相应策略表的接口，其中 IFNAME 是该接口的名称。

下面例子显示了出队列调度参数的配置。编号为 5 和 6 的流的优先级是最高的值 6，编号为 2 的流的优先级是 2，带宽为 link 带宽的 20%。

表9-3 配置调度

Switch# configure terminal	进入全局配置模式
Switch(config)# class-map type traffic-class tc5	创建 class-map 并进入其配置模式
Switch(config-cmap-tc)# match traffic-class 5	设置流优先级为 5
Switch(config-cmap-tc)# exit	退出该配置模式
Switch(config)# class-map type traffic-class tc6	创建 class-map 并进入其配置模式
Switch(config-cmap-tc)# match traffic-class 6	设置流优先级为 6
Switch(config-cmap-tc)# exit	退出该配置模式
Switch(config)# class-map type traffic-class tc2	创建 class-map 并进入其配置模式
Switch(config-cmap-tc)# match traffic-class 2	设置流优先级为 2
Switch(config-cmap-tc)# exit	退出该配置模式
Switch(config)# policy-map type traffic-class tc	创建 policy-map 并进入其配置模式
Switch(config-pmap-tc)# class type traffic-class tc5	关联 class-map tc5
Switch(config-pmap-tc-c)# priority level 6	设置优先级为 6
Switch(config-pmap-tc-c)# exit	退出到 policy-map 模式
Switch(config-pmap-tc)# class type traffic-class tc6	关联 class-map tc6
Switch(config-pmap-tc-c)# priority level 6	设置优先级为 6
Switch(config-pmap-tc-c)# exit	退出到 policy-map 模式
Switch(config-pmap-tc)# class type traffic-class tc2	关联 class-map tc2
Switch(config-pmap-tc-c)# bandwidth percentage 20	配置带宽为 link 带宽的 20%

Switch(config-pmap-tc-c)# exit	退出至 policy-map 模式
Switch(config-pmap-tc)# exit	退出该配置模式
Switch(config)# interface eth-0-1	进入端口配置模式
Switch(config-if)# service-policy type traffic-class tc	应用 QoS policy
Switch(config-if)# end	退出全局配置模式
Switch# show qos interface eth-0-1 egress	显示 QoS 配置

验证配置

Switch# show qos interface eth-0-1 egress

TC	Priority	Bandwidth	Shaping(kbps)	Drop-Mode	Max-Queue-Limit(Cell)	ECN
0	0	-	-	dynamic	level 0	-
1	0	-	-	random-drop	596	Disable
2	0	20	-	dynamic	level 0	-
3	0	-	-	tail-drop	2000	2000
4	0	-	-	dynamic	level 0	-
5	6	-	-	dynamic	level 0	-
6	6	-	-	dynamic	level 0	-
7	7	-	-	tail-drop	64	-

Port policing

经过交换机物理接口的所有流量都可以设置保证速率，超过保证速率的流量都会被丢弃。

下面的例子说明了如何配置端口 Policer 来实现保证速率。

- configure terminal
- interface IFNAME 进入匹配相应策略表的接口，其中 IFNAME 是该接口的名称。
- port-policer input|output color-blind|color-aware cir <8- 100000000 > cbs <1000-640000> ebs <1000-640000>| eir <8-100000000> ebs <1000-640000> drop-color exceed|violate 可以配置端口 Policer。

no port-policier input|output 命令删除端口 Policer 配置。

下面例子显示创建进端口 Policer，当收到的报文平均速率超过 48000-kbps，其将被丢弃。

表9-4 配置端口 policing

Switch# configure terminal	进入全局配置模式

Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# qos policer input color-blind cir 48000 cbs 10000 ebs 20000 violate drop	配置该接口的保证速率为 48000kbps
Switch(config-if)# end	退出到特权模式
Switch# show qos interface eth-0-1 statistics policer port input	显示 QoS 配置状态

验证配置

Switch# show qos interface eth-0-1 statistics policer port input

```
Interface: eth-0-1
  input port policer:
    color blind
     CIR 48000 kbps, CBS 10000 bytes, EBS 20000 bytes
     drop violate packets
```

I. Shaping

接口整形

经过交换机物理接口的所有流量都可以被整形，超过整形速率的流量会被缓存，但是如果缓存耗尽，则后续的报文会被丢弃直到缓存被释放。

下面的例子说明了如何配置基于物理接口的流量整形。

- configure terminal
- interface IFNAME 进入匹配相应策略表的接口，其中 IFNAME 是该接口的名称。
- qos shape rate <0-100000000>用来配置端口流量整形的阈值。

NOTE

no shape 命令删除该流量整形配置。

下面示例显示了配置流量整形的过程。当接收流的速率超过 1000Mbps 将被丢弃。

表9-5 配置端口流量整形

Switch#configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# qos shape rate 1000000	配置端口流量速率超过 1000Mbps 时，将被丢弃
Switch(config-if)# end	退出到特权模式

Switch# show running-config interface eth-0-1	显示 QoS 配置状态

验证配置

Switch# show running-config interface eth-0-1

```
Building configuration...
!
interface eth-0-1
 service-policy type traffic-class tc
 qos policer input color-blind cir 48000 cbs 10000 ebs 20000 violate drop
 qos shape rate 1000000
!
```

队列整形

流量在经过交换机出方向的队列的时候可以被整形，超过整形速率的流量会被缓存，但是如果缓存耗尽，则后续的报文会被丢弃直到缓存被释放。

下面的例子显示了如何在出方向队列上配置流量整形

- configure terminal
- 创建 class-map 类流优先级，并配置优先级；
- 创建 policy-map 类流优先级，并关联之前定义的 class-map；
- 在 policy-map 优先级模式下配置对应流优先级的流量整形；
- interface IFNAME 进入匹配相应策略表的接口，其中 IFNAME 是该接口的名称

no shape rate 命令删除队列整形

示例显示对队列 3 进行队列整形的配置。当队列 3 中流速率超过 1000Mbps，将丢弃报文。

表9-6 配置出方向队列流量整形

Switch# configure terminal	进入全局配置模式
Switch(config)# class-map type traffic-class tc3	创建 class-map 并进入其配置模式
Switch(config-cmap-tc)# match traffic-class 3	配置流优先级为 3
Switch(config-cmap-tc)# exit	退出该配置模式
Switch(config)# policy-map type traffic-class tc	创建 policy-map 并进入其配置模式
Switch(config-pmap-tc)# class type traffic-class tc3	关联 class-map

Switch(config-pmap-tc-c)# shape rate 1000000	按 1000Mbps 速率进行队列整形
Switch(config-pmap-tc-c)# exit	退出至 policy-map 模式
Switch(config-pmap-tc)# exit	退出至配置模式
Switch(config)# interface eth-0-1	进入端口配置模式
Switch(config-if)# service-policy type traffic-class tc	应用 QoS policies
Switch(config-if)# end	退出全局配置模式
Switch# show qos interface eth-0-1 egress	显示 QoS 配置

验证配置

Switch# show qos interface eth-0-1 egress

```
TC  Priority  Bandwidth  Shaping(kbps)  Drop-Mode    Max-Queue-Limit(Cell)    ECN
0   0         -          -              dynamic      level 0                  -
1   0         -          -              random-drop  596                      Disable
2   0         20         -              dynamic      level 0                  -
3   0         -          1000000        tail-drop    2000                     2000
4   0         -          -              dynamic      level 0                  -
5   6         -          -              dynamic      level 0                  -
6   6         -          -              dynamic      level 0                  -
7   7         -          -              tail-drop    64                       -
```

I. Policy

在部署 QoS 流量策略时需要执行如下几个步骤。

- 识别并区分流量到不同的类别。
- 对不同的流量类别配置策略。
- 在接口上应用策略。

使用 ACL 实现流量分类

IP 流量使用 IP ACL 作流量分类。

下面的例子说明了如何创建 IP ACL 来区分不同的流量并将其分类。

- configure terminal
- ip access-list ACCESS-LIST-NAME 创建 ACL，其中 ACCESS-LIST-NAME 为 ACL 名。
- 根据需要创建一到多条 ACE，详细方法请参见 *ACL 用户配置手册*。

NOTE

no ip access-list 命令删除 access list 配置。

示例显示允许三类 IP 地址的主机访问，网络地址主机部分对应为通配符。如果一台主机的 IP 地址不在 list 的匹配范围，则该主机将被拒绝访问。

表9-7 配置流策略

Switch# configure terminal	进入全局配置模式
Switch(config)# ip access-list ip-acl	进入 IP ACL 配置模式
Switch(config-ip-acl)# permit any 128.88.12.0 0.0.0.255 any	配置允许源 IP 地址 128.88.12.x 的流量
Switch(config-ip-acl)# permit any 28.88.0.0 0.0.255.255 any	配置允许源 IP 地址 28.88.x.x 的流量
Switch(config-ip-acl)# permit any 11.0.0.0 0.255.255.255 any	配置允许源 IP 地址 11.xx.x.x 的流量
Switch(config-ip-acl)# end	退出到特权模式
Switch# show access-list ip ip-acl	显示 ACL 配置状态

验证配置

Switch# show access-list ip ip-acl

```
ip access-list ip-acl
  10 permit any 128.88.12.0 0.0.0.255 any
  20 permit any 28.88.0.0 0.0.255.255 any
  30 permit any 11.0.0.0 0.255.255.255 any
```

创建分类映射表

下面的例子说明了如何将指定接口的 IP 流量按照分类表作流量分类。期间涉及到创建分类映射表以及匹配准则。

- configure terminal
- ip access-list ACCESS-LIST-NAME 创建 ACL，其中 ACCESS-LIST-NAME 为 ACL 名。
- 根据需要创建一至多条 ACE。详细方法请参见 *ACL 用户配置手册*。
- class-map (match-any|match-all) NAME 用来创建分类映射表。match-any 表示映射表中的分类按照逻辑或的关系进行匹配，即分类映射表中至少匹配一条即可分类。match-all 表示映射表中的分类按照逻辑与的关系进行匹配，即分类映射表中必须所有都匹配才可分类。NAME 表示分类映射表的名称。

NOTE

默认是按照 match-any 方式进行流量分类的。

- match access-group NAME 用来定义分类标准，NAME 表示需要关联的 ACL 表名

NOTE

no class-map 删除分类映射表的配置。

示例显示使用 IP access list 创建一个名为 cmap1 的分类映射表，允许任意源主机到目的主机的流量传输。

表9-8 创建分类映射表

Switch# configure terminal	进入全局配置模式
Switch(config)# ip access-list ip-acl	进入 IP ACL 配置模式.
Switch(config-ip-acl)# permit any any any	允许所有报文
Switch(config-ip-acl)# quit	退出到全局配置模式
Switch(config)# class-map cmap1	创建 cmap1 并进入分类映射表配置模式
Switch (config-cmap)# match access-group ip-acl	将 ip-acl 加入到 cmap1 中
Switch (config-cmap)# quit	退出到特权模式
Switch# show class-map cmap1	显示分类表配置

验证配置

Switch# show class-map cmap1

```
    CLASS-MAP-NAME: cmap1 (match-any)
      match access-group: ip-acl
```

创建策略表

下面的例子说明了如何创建策略表用于对流量进行分类，标记和限流。

- configure terminal
- ip access-list 创建 IP ACL。
- class-map type qos NAME 创建分类映射表。
- policy-map type qos NAME 创建策略表，其中 NAME 表示策略表名称。
- class NAME 定义一个流分类条目，其中 NAME 表示流分类条目的名称。
- set traffic-class <1-6>用于设置匹配流分类表的报文的优先级。

- set color red|yellow|green 用于设置匹配流分类表的报文的颜色。
- policer color-blind|color-aware cir <8-10000000> cbs <1000-640000> ebs <1000-640000>| eir <8-10000000> ebs <1000-128000> (exceed | violate) drop 用来定义一条策略。
- exit
- exit
- interface IFNAME 进入匹配相应策略表的接口，其中 IFNAME 是该接口的名称。
- service-policy type qos input NAME 在指定接口上对输入和输出地流量应用策略表。

接口下地每一个方向只允许配置一个策略映射表。

no policy-map 命令删除已经存在的策略表；no set priority color 命令移除优先级颜色；no policer 命令移除一个已存在的 policer；no service-policy input|output 命令在端口删除策略表配置。

示例显示创建了一个策略表，并应用到一个端口的进口流量。配置的 IP ACL 允许来自 10.1.00 地址的流量，如果这些流量的平均速率超过 48000-kbps，将被丢弃。

表9-9 创建策略表

Switch#configure terminal	进入全局配置模式
Switch(config)# ip access-list ip-acl	进入 IP ACL 配置模式.
Switch(config-ip-acl)# permit any 10.1.0.0 0.0.255.255 any	配置允许源 IP 地址 10.1.x.x 的流量
Switch(config-ip-acl)# quit	退出到全局配置模式
Switch(config)# class-map type qos cmap1	创建 cmap1 并进入分类映射表配置模式
Switch(config-cmap)# match access-group ip-acl	将 ip-acl 加入到 cmap1 中
Switch(config-cmap)# quit	退出到全局配置模式
Switch(config)# policy-map type qos pmap1	配置策略表 pmap1 并进入策略表配置模式
Switch(config-pmap)# class type qos cmap1	将流分类映射表 cmap1 加入策略表 pmap1
Switch(config-pmap-c)# policer color-blind cir 48000 cbs 10000 ebs 16000 violate drop	配置该接口的保证速率为 48000kbps
Switch(config-pmap-c)# quit	退出到策略表配置模式
Switch(config-pmap)# quit	退出到全局配置模式

Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# service-policy type qos input pmap1	将策略表 pmap1 应用到接口
Switch(config-if)# end	退出到特权模式
Switch# show policy-map pmap1	显示策略表配置.

验证配置

Switch# show policy-map pmap1

```
  POLICY-MAP-NAME: pmap1 ( type qos)
    State: attached
    CLASS-MAP-NAME: cmap1
      match access-group: ip-acl
        policer color-blind cir 48000 cbs 10000 ebs 16000 violate drop
```

创建聚合策略

下面的例子说明了如何创建聚合策略表用于对流量进行分类，标记和限流。

- configure terminal
- qos aggregate-policer NAME color-blind|color-aware cir <0-100000000> cbs <0-640000> ebs <0- 640000>| eir <0-100000000> ebs <0-640000> exceed|violate drop 用来指定需要被应用一个或多个策略表中的多个流分类策略的聚合策略的各项参数。
- class-map type qos NAME 创建分类映射表。
- policy-map type qos NAME 创建策略表。
- class type qos NAME 定义一个流分类条目。
- aggregate-policer NAME 用来应用一个或多个策略表中的多个流分类策略的聚合策略。
- exit
- exit
- interface IFNAME 进入匹配相应策略表的接口。
- service-policy type qos input NAME 在指定接口上对输入和输出地流量应用策略表。

NOTE

接口下地每一个方向只允许配置一个策略映射表。

no policer-aggregate 命令从策略表中删除一条聚合策略；no qos aggregate-policer 命令删除一条聚合策略。

示例显示创建了一条聚合策略，并运用于策略表中的多个表项。示例中，IP ACLs 允许来自网络地址 10.1.0.0 和主机地址为 11.3.1.1 的流量，且配置了其平均速率。当流量平均速率超过 48000-kbps 且流量大小超过 8000-byte，该流量将被丢弃。该策略表运用于端口的进流量。

表9-10 创建聚合策略

Switch#configure terminal	进入全局配置模式
Switch(config)# ip access-list ip-acl1	进入 IP ACL 配置模式.
Switch(config-ip-acl)# permit any 10.1.0.0 0.0.255.255 any	配置允许源 IP 地址 10.1.x.x 的流量
Switch(config-ip-acl)# exit	退出到全局配置模式
Switch(config)# ip access-list ip-acl2	进入 IP ACL 配置模式.
Switch(config-ip-acl)# permit any host 11.3.1.1 any	配置允许源 IP 地址 11.3.1.1 的流量
Switch(config-ip-acl)# exit	退出到全局配置模式
Switch(config)# qos aggregate-policer transmit1 color-blind cir 48000 cbs 8000 ebs 10000 violate drop	配置聚合策略的保证速率为 48000kbps
Switch(config)# class-map type qos cmap1	创建 cmap1 并进入分类映射表配置模式
Switch(config-cmap)# match access-group ip-acl1	将 ip-acl1 加入到 cmap1 中
Switch(config-cmap)# exit	退出到全局配置模式
Switch(config)# class-map type qos cmap2	创建 cmap2 并进入分类映射表配置模式
Switch(config-cmap)# match access-group ip-acl2	将 ip-acl2 加入到 cmap2 中
Switch(config-cmap)# exit	退出到全局配置模式
Switch(config)# policy-map type qos aggflow1	配置策略表 aggflow1 并进入策略表配置模式
Switch(config-pmap)# class type qos cmap1	将流分类映射表 cmap1 加入策略表 aggflow1
Switch(config-pmap-c)# aggregate-policer transmit1	将 cmap1 设置为聚合策略 transmit1
Switch(config-pmap-c)# exit	退出到策略表配置模式
Switch(config-pmap)# class type qos cmap2	将流分类映射表 cmap2 加入策略表 pmap1

Switch(config-pmap-c)# aggregate-policer transmit1	将 cmap2 设置为聚合策略 transmit1
Switch(config-pmap-c)# exit	退出到策略表配置模式
Switch(config-pmap)# exit	退出到全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# service-policy type qos input aggflow1	将聚合策略 aggflow1 应用到接口
Switch(config-if)# exit	退出到端口配置模式
Switch(config)# exit	退出到全局配置模式
Switch# show qos aggregate-policer	显示聚合策略配置状态

验证配置

Switch# show qos aggregate-policer

```
Aggreate policer: transmit1
  color blind
  CIR 48000 kbps, CBS 8000 bytes, EBS 10000 bytes
  drop violate packets
```

10 IPv6 安全配置指导

10.1 DHCPv6 Snooping 配置

10.1.1 简介

DHCPv6 Snooping 是一种安全功能，如不受信任的 DHCPv6 客户端和信任的 DHCPv6 服务器之间的防火墙行为，DHCPv6 Snooping 功能执行如下：

- 验证 DHCPv6 消息接收来自不信任的源和过滤掉无效消息。
- 建立和维护 DHCPv6 Snooping 绑定数据库，其中包含 DHCPv6 客户端租用的 IPv6 地址信息。
- DHCPv6 Snooping 功能在软件中实现，所有 DHCPv6 消息在芯片中被拦截直接发往 CPU 进行处理。

10.1.2 拓扑

错误!未找到引用源。为测试 DHCPv6 snooping 功能的网络拓扑，需要两台 PC 机和一台交换机构建测试环境，具体分配可参照如下描述。

- 计算机 A 作为 DHCPv6 服务器
- 计算机 B 作为 DHCPv6 客户端
- 交换机作为 DHCPv6 snooping

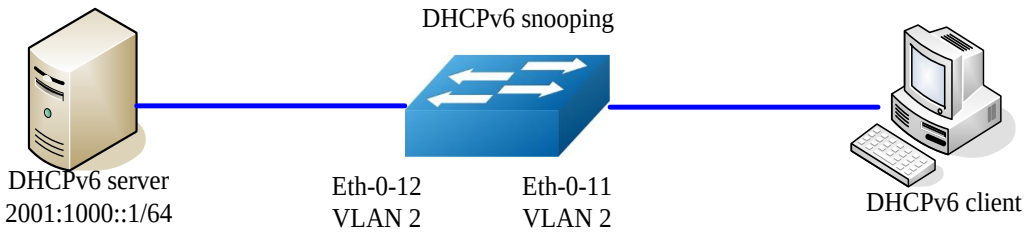


图10-1 DHCP v6 snooping 拓扑图

10.1.3 配置

配置 VLAN

Switch# configure terminal	进入全局配置模式
Switch(config)# vlan database	配置 VLAN 数据库
Switch(config-vlan)# vlan 2	创建 VLAN 2
Switch(config-vlan)# exit	退出到全局配置模式

配置接口 eth-0-12

Switch(config)# interface eth-0-12	进入接口配置模式
Switch(config-if)# switchport	设置为交换接口
Switch(config-if)# switchport access vlan 2	添加接口到 VLAN 2
Switch(config-if)# dhcpv6 snooping trust	配置接口为信任状态
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出接口配置模式

配置接口 eth-0-11

Switch(config)# interface eth-0-11	进入接口配置模式
Switch(config-if)# switchport	设置为交换接口
Switch(config-if)# switchport access vlan 2	添加接口到 VLAN 2

Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出全局配置模式

使能 DHCPv6 snooping 全局特性

Switch(config)# service dhcpv6 enable	使能 dhcp 服务
Switch(config)# dhcpv6 snooping	使能 dhcp snooping 特性
Switch(config)# dhcpv6 snooping vlan 2	在 VLAN 2 上使能 dhcp snooping 特性

10.1.4 命令验证

步骤 1 根据如下步骤，检查接口配置是否正确。

Switch# show running-config interface eth-0-12

```
!
interface eth-0-12
switchport access vlan 2
dhcpv6 snooping trust
!
```

Switch# show running-config interface eth-0-11

```
!
interface eth-0-11
 switchport access vlan 2
!
```

步骤 2 使用如下命令，检查 DHCPv6 服务状态。

Switch# show services

```
Networking services configuration:
Service Name          Status
============================================================
dhcp                    disable
dhcpv6                  enable
```

步骤 3 使用如下命令，打印 dhcpv6 snooping 配置，检查当前配置。

Switch# show dhcpv6 snooping config

```
dhcpv6 snooping service: enabled
dhcpv6 snooping switch: enabled
dhcpv6 snooping vlan 2
```

步骤 4 使用如下命令，检查 dhcpv6 snooping 的统计信息。

Switch# show dhcpv6 snooping statistics

```
DHCPv6 snooping statistics:
============================================================
DHCPv6 packets                                  21
Packets forwarded                             21
Packets invalid                                0
Packets dropped                                0
```

步骤 5 使用如下命令，显示 dhcpv6 snooping 绑定信息。

Switch# show dhcpv6 snooping binding all

```
DHCPv6 snooping binding table:
VLAN MAC Address    Lease(s)    Interface   IPv6 Address
============================================================
2    0016.76a1.7ed9 978         eth-0-11    2001:1000::2
```

11 IPv6 路由配置指导

11.1 IPv6 单播路由配置

11.1.1 简介

静态路由是一种特殊的路由，由管理员手工配置。当网络结构比较简单时，只需配置静态路由就可以使网络正常工作。合理设置和使用静态路由可以改进网络性能，并可为重要的网络应用保证带宽。静态路由的缺点在于：当网络发生故障或者拓扑发生变化后，可能会出现路由不可达，从而导致网络中断。此时必须由网络管理员手工修改静态路由的配置。

这个例子说明在一个简单的网络拓扑结构下如何使能静态路。静态路由在小型网络中非常有用。静态路由可提供使几个目的地可达的简单解决方案。大型网络使用动态路由协议。静态路由是由网络前缀（主机地址）和下一跳（网关）组成。

11.1.2 拓扑

图11-1 IPv6 静态路由拓扑

11.1.3 配置 IPv6 静态路由

I. Switch1 的配置

Switch1# configure terminal	进入全局配置模式
Switch1 (config)# ipv6 enable	使能 IPv6
Switch1 (config)# interface eth-0-9	进入接口模式
Switch1 (config-if)# no switchport	设置接口为三层接口

Switch1 (config-if)# no shutdown	打开接口
Switch1 (config-if)# ipv6 address auto link-local	配置自动生成链路本地地址
Switch1 (config-if)# ipv6 address 2001:1::1/64	配置全球单播地址
Switch1 (config-if)# exit	退出接口模式
Switch1 (config)# ipv6 route 2001:2::/64 2001:1::2	配置 IPv6 静态路由
Switch1 (config)# end	退出全局配置模式

II. Switch2 的配置

Switch2# configure terminal	进入全局配置模式
Switch2 (config)# ipv6 enable	使能 IPv6
Switch2 (config)# interface eth-0-9	进入接口模式
Switch2 (config-if)# no switchport	设置接口为三层接口
Switch2 (config-if)# no shutdown	打开接口
Switch2 (config-if)# ipv6 address auto link-local	配置自动生成链路本地地址
Switch2 (config-if)# ipv6 address 2001:1::2/64	配置全球单播地址
Switch2 (config-if)# exit	退出接口模式
Switch2 (config)# interface eth-0-17	进入接口模式
Switch2 (config-if)# no switchport	设置接口为三层接口
Switch2 (config-if)# no shutdown	打开接口
Switch2 (config-if)# ipv6 address auto link-local	配置自动生成链路本地地址
Switch2 (config-if)# ipv6 address 2001:2::2/64	配置全球单播地址
Switch2 (config-if)# exit	退出接口模式
Switch2 (config)# end	退出全局配置模式

III. Switch3 的配置

Switch3# configure terminal	进入全局配置模式
Switch3 (config)# ipv6 enable	使能 IPv6
Switch3 (config)# interface eth-0-17	进入接口模式
Switch3 (config-if)# no switchport	设置接口为三层接口
Switch3 (config-if)# no shutdown	打开接口
Switch3 (config-if)# ipv6 address auto link-local	配置自动生成链路本地地址
Switch3 (config-if)# ipv6 address 2001:2::3/64	配置全球单播地址
Switch3 (config-if)# exit	退出接口模式
Switch3 (config)# ipv6 route 2001:1::/64 2001:2::2	配置 IPv6 静态路由
Switch3 (config)# end	退出全局配置模式

11.1.4 命令验证

Switch1# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, I - IS-IS, B - BGP
       [*] - [AD/Metric]
Timers: Uptime
C       2001:1::/64
       via ::, eth-0-9, 02:08:50
C       2001:1::1/128
       via ::1, eth-0-9, 02:08:50
S       2001:2::/64 [1/0]
       via 2001:1::2, eth-0-9, 02:05:36
C       fe80::/10
       via ::, Null0, 02:09:11
```

Switch2# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, I - IS-IS, B - BGP
       [*] - [AD/Metric]
Timers: Uptime
C       2001:1::/64
       via ::, eth-0-9, 00:03:37
C       2001:1::2/128
       via ::1, eth-0-9, 00:03:37
C       2001:2::/64
       via ::, eth-0-17, 00:03:21
```

```
C        2001:2::2/128
         via ::1, eth-0-17, 00:03:21
C        fe80::/10
         via ::, Null0, 00:03:44
```

Switch3# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, I - IS-IS, B - BGP
       [*] - [AD/Metric]
Timers: Uptime
S        2001:1::/64 [1/0]
         via 2001:2::2, eth-0-17, 00:02:14
C        2001:2::/64
         via ::, eth-0-17, 00:03:28
C        2001:2::3/128
         via ::1, eth-0-17, 00:03:28
C        fe80::/10
         via ::, Null0, 00:03:53
```

Ping Switch3 on Switch1:

Switch1# ping ipv6 2001:2::3

```
PING 2001:2::3(2001:2::3) 56 data bytes
64 bytes from 2001:2::3: icmp_seq=0 ttl=63 time=127 ms
64 bytes from 2001:2::3: icmp_seq=1 ttl=63 time=132 ms
64 bytes from 2001:2::3: icmp_seq=2 ttl=63 time=124 ms
64 bytes from 2001:2::3: icmp_seq=3 ttl=63 time=137 ms
64 bytes from 2001:2::3: icmp_seq=4 ttl=63 time=141 ms
--- 2001:2::3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4010ms
rtt min/avg/max/mdev = 124.950/132.719/141.251/5.923 ms, pipe 2
```

11.2 OSPFv3 配置

11.2.1 简介

开放最短路径优先协议 OSPF（Open Shortest Path First）是 IETF 组织开发的一个基于链路状态的内部网关协议，OSPFv3 是 OSPF 版本 3 的简称，主要提供对 IPv6 路由的支持，遵循的标准是 RFC5340(OSPF for IPv6，OSPFv3 和 OSPFv2 有很多方面是相同的：

- Router ID，Area ID，LSA Link State ID 仍然是 32 位的。
- 协议报文类型一样：Hello 报文、DD 报文、LSR 报文、LSU 报文和 LSAck 报文。
- 邻居发现和邻接建立机制一样
- LSA 泛洪和老化机制一样。

OSPFv3 和 OSPFv2 有如下不同点：

- OSPFv3 是基于 link 运行的，而 OSPFv2 是基于 network 运行的。

- OSPFv3 在同一个 link 上可以运行多个实例。
- OSPFv3 的拓扑关系和 IPv6 前缀信息分离。
- 使用 Link-local 地址作为路由下一条
- 新增了 link lsa 以及本地链路泛洪范围

当前的系统支持如下 OSPFv3 特性：

- **支持末梢区域**：支持路由重分布，这包括将其他路由协议学到的路由导入 OSPFv3 或者将 OSPFv3 学到的路由导出到其他路由协议中。
- **支持 OSPFv3 多 process**。
- **支持 link 上多 instance**。

11.2.2 参考文献

OSPF 模块是基于以下 RFC：

RFC 5340 – OSPF for IPv6

11.2.3 配置基本 OSPFv3

在需要启用 OSPFv3 的路由器上先创建 OSPFv3 进程，需要手工指定 OSPFv3 的 Router ID。配置如下表所示。

Switch# configure terminal	进入配置模式
Switch(config)# router ipv6 ospf 100	创建 OSPFv3 进程号 100
Switch(config-router)# router-id 1.1.1.1	指定 Router ID
Switch(config-router)# end	返回到配置模式
Switch# show ipv6 protocols	检查配置的协议

在全局模式下通过命令“no router ipv6 ospf *process-id*”取消 OSPFv3 进程。

11.2.4 启用 OSPF

这个例子显示了一个接口上启用 OSPFv3 所需的最低配置。

I. 拓扑

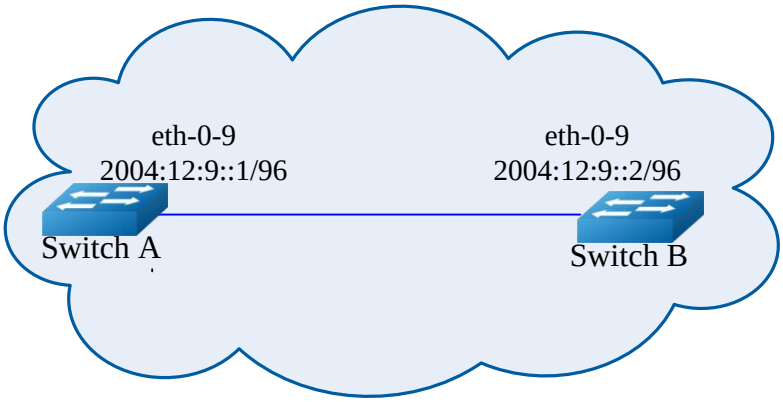


图11-2 OSPFv3 自治系统

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 100	创建 OSPFv3 进程号 100
Switch(config-router)# router-id 1.1.1.1	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 100 area 0 instance 0	将该端口加入到 OSPFv3 process100，area0， instance0 中
Switch(config-if)# end	退出配置模式

Switch B

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6

Switch(config)# router ipv6 ospf 200	创建 OSPFv3 进程号 200
Switch(config-router)# router-id 2.2.2.2	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 200 area 0 instance 0	将该端口加入到 OSPFv3 process200，area0， instance0 中
Switch(config-if)# end	退出配置模式

III. 命令验证

使用如下命令，验证上述配置：

show ipv6 ospf database

show ipv6 ospf interface

show ipv6 ospf neighbor

show ipv6 ospf route

Switch A

Switch# show ipv6 ospf database

```
            OSPFv3 Router with ID (1.1.1.1) (Process 100)
                Link-LSA (Interface eth-0-9)
Link State ID   ADV Router      Age  Seq#       CkSum  Prefix
0.0.0.9         1.1.1.1         614 0x80000001 0x6a40      1
0.0.0.9         2.2.2.2          68 0x80000001 0x4316      1
                Router-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age  Seq#       CkSum    Link
0.0.0.0         1.1.1.1          54 0x80000003 0xb74b      1
0.0.0.0         2.2.2.2          55 0x80000003 0x9965      1
                Network-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age  Seq#       CkSum
0.0.0.9         1.1.1.1          54 0x80000001 0x3ed1
                Intra-Area-Prefix-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age  Seq#       CkSum  Prefix  Reference
0.0.0.2         1.1.1.1          53 0x80000001 0x450a      1  Network-LSA
```

Switch# show ipv6 ospf neighbor

```
OSPFv3 Process (100)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
2.2.2.2           1   Full/Backup      00:00:33    eth-0-9    0
```

Switch# show ipv6 ospf route

```
OSPFv3 Process (100)
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2
   Destination                                  Metric
     Next-hop
C  2004:12:9::/96                                 1
     directly connected, eth-0-9, Area 0.0.0.0
```

Switch B

Switch# show ipv6 ospf database

```
            OSPFv3 Router with ID (2.2.2.2) (Process 200)
                Link-LSA (Interface eth-0-9)
Link State ID   ADV Router      Age  Seq#       CkSum  Prefix
0.0.0.9         1.1.1.1          774 0x80000001 0x6a40      1
0.0.0.9         2.2.2.2          228 0x80000001 0x4316      1
                Router-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age  Seq#       CkSum    Link
0.0.0.0         1.1.1.1          217 0x80000003 0xb74b      1
0.0.0.0         2.2.2.2          214 0x80000003 0x9965      1
                Network-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age  Seq#       CkSum
0.0.0.9         1.1.1.1          215 0x80000001 0x3ed1
                Intra-Area-Prefix-LSA (Area 0.0.0.0)
Link State ID   ADV Router      Age  Seq#       CkSum  Prefix  Reference
0.0.0.2         1.1.1.1          214 0x80000001 0x450a      1  Network-LSA
```

Switch# show ipv6 ospf neighbor

```
OSPFv3 Process (200)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
1.1.1.1           1   Full/DR          00:00:35    eth-0-9    0
```

Switch# show ipv6 ospf route

```
OSPFv3 Process (200)
Codes: C - connected, D - Discard, O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2
   Destination                                  Metric
     Next-hop
C  2004:12:9::/96                                 1
     directly connected, eth-0-9, Area 0.0.0.0
```

11.2.5 配置优先级

这个例子主要讲述了如何配置接口优先级，优先级高的成为 DR。优先级为 0 的不参与 DR 选举。Switch C 的优先级是 10，这比 Switch A 和 Switch B 的默认优先级 1 要高，因此 Switch C 将成为这个网络内的 DR。

I. 拓扑

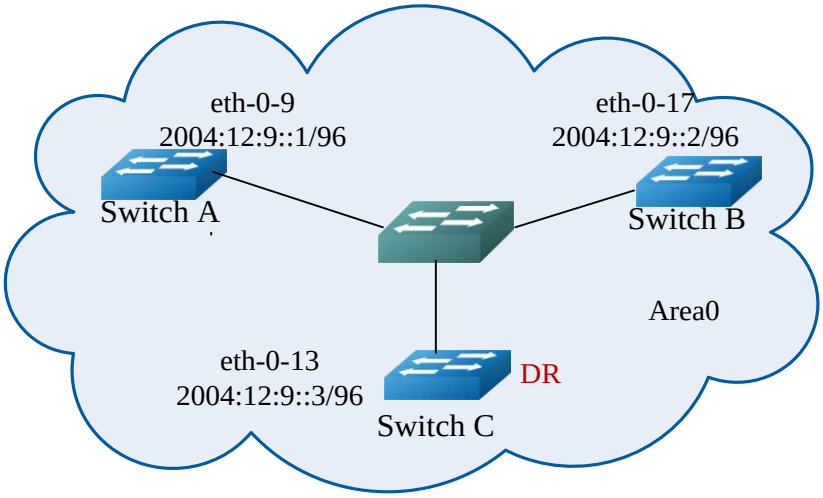


图11-3 OSPFv3 优先级

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 100	创建 OSPFv3 进程号 100
Switch(config-router)# router-id 1.1.1.1	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 100 area 0 instance 0	将该端口加入到 OSPFv3 process100，area0， instance0 中

Switch(config-if)# end	退出配置模式

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 200	创建 OSPFv3 进程号 200
Switch(config-router)# router-id 2.2.2.2	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-17	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 200 area 0 instance 0	将该端口加入到 OSPFv3 process200，area0， instance0 中
Switch(config-if)# end	退出配置模式

Switch C

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 300	创建 OSPFv3 进程号 300
Switch(config-router)# router-id 3.3.3.3	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-13	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::3/96	配置 IPv6 地址

Switch(config-if)# ipv6 router ospf 300 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# end	退出配置模式

III. 命令验证

使用如下命令，验证以上配置是否正确：

show ipv6 ospf neighbor

show ipv6 ospf interface

Switch C

Switch# show ipv6 ospf interface

```
eth-0-13 is up, line protocol is up
  Interface ID 13
  IPv6 Prefixes
    fe80::ee66:91ff:fe45:db00/10 (Link-Local Address)
    2004:12:9::3/96
  OSPFv3 Process (300), Area 0.0.0.0, Instance ID 0
    Router ID 3.3.3.3, Network Type BROADCAST, Cost: 1
    Transmit Delay is 1 sec, State DR, Priority 10
    Designated Router (ID) 3.3.3.3
      Interface Address fe80::ee66:91ff:fe45:db00
    Backup Designated Router (ID) 2.2.2.2
      Interface Address fe80::c629:f2ff:fe02:3600
    Timer interval configured, Hello 10, Dead 40, Wait 40, Retransmit 5
      Hello due in 00:00:01
    Neighbor Count is 2, Adjacent neighbor count is 2
```

Switch# show ipv6 ospf neighbor

```
OSPFv3 Process (300)
Neighbor ID     Pri   State           Dead Time   Interface  Instance ID
1.1.1.1           1   Full/DROther      00:00:32    eth-0-13   0
2.2.2.2           1   Full/Backup       00:00:36    eth-0-13   0
```

11.2.6 配置 OSPFv3 区域参数

您可以选择性地配置多个 OSPFv3 区域参数。这些参数将区域配置为末梢区域(Stub)。Stub 区域是一些特定的区域，Stub 区域的 ABR 不传播它们接收到的自治系统外部路由，在这些区域中路由器的路由表规模以及路由信息传递的数量都会大大减少。为保证到自治系统外的路由依旧可达，该区域的 ABR 将生成一条缺省路由，并发布给 Stub 区域中的其他非 ABR 路由器。

路由聚合是指 ABR 或 ASBR 将具有相同前缀的路由信息聚合，只发布一条路由到其它区域。AS 被划分成不同的区域后，区域间可以通过路由聚合来减少路由信息，减小路由表的规模，提高路由器的运算速度。如果网络号是连续的，你可以使用 area range 命

令将这些连续的网段聚合成一个网段。这样 ABR 只发送一条聚合后的 LSA，所有属于本命令指定的聚合网段范围的 LSA 将不再会被单独发送出去，这样可减少其它区域中 LSDB 的规模。

I. 拓扑

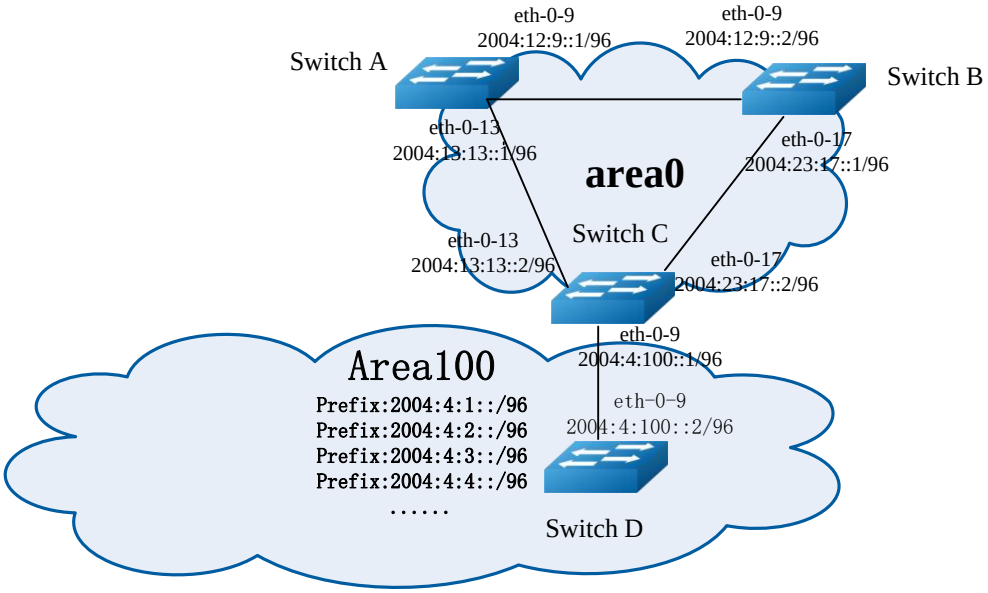


图11-4 OSPFv3 区域

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 100	创建 OSPFv3 进程号 100
Switch(config-router)# router-id 1.1.1.1	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 100 area 0 instance 0	将该端口加入到 OSPFv3 process100，area0， instance0 中

Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)#interface eth-0-13	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:13:13::2/96	设置端口的 IP 地址
Switch(config-if)# ipv6 router ospf 100 area 0 instance 0	将该端口加入到 OSPFv3 process100，area0， instance0 中
Switch(config-if)# end	退出配置模式

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 200	创建 OSPFv3 进程号 200
Switch(config-router)# router-id 2.2.2.2	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 200 area 0 instance 0	将该端口加入到 OSPFv3 process200，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)#interface eth-0-17	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#no shutdown	Up 端口

Switch(config-if)# ipv6 address 2004:23:17::1/96	设置端口的 IP 地址
Switch(config-if)# ipv6 router ospf 200 area 0 instance 0	将该端口加入到 OSPFv3 process200，area0， instance0 中
Switch(config-if)# end	退出配置模式

Switch C

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 300	创建 OSPFv3 进程号 300
Switch(config-router)# router-id 3.3.3.3	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-13	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:13:13::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 300 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)# interface eth-0-17	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:23:17::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 300 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.

Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:100::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 300 area 100 instance 0	将该端口加入到 OSPFv3 process300，area100， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)# router ipv6 ospf 300	进入 OSPF 进程号 300
Switch(config-router)# area 100 range 2004:4::/32	指定一段 prefix 发布到 OSPFv3 区域 0
Switch(config-router)# area 100 stub no-summary	区域 100 设置成 Stub 区域
Switch(config-if)# end	退出配置模式

Switch D

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 400	创建 OSPFv3 进程号 400
Switch(config-router)# router-id 4.4.4.4	指定 Router ID
Switch(config-router)# area 100 stub no-summary	区域 100 设置成 Stub 区域
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:100::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 400 area 100 instance 0	将该端口加入到 OSPFv3 process300，area100， instance0 中

Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:1::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 400 area 100 instance 0	将该端口加入到 OSPFv3 process300，area100， instance0 中
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:2::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 400 area 100 instance 0	将该端口加入到 OSPFv3 process300，area100， instance0 中
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-3	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:3::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 400 area 100 instance 0	将该端口加入到 OSPFv3 process300，area100， instance0 中
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-4	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:4::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 400 area 100 instance 0	将该端口加入到 OSPFv3 process300，area100， instance0 中

Switch(config-if)# end	退出配置模式

III. 命令验证

使用 **show ipv6 route** 命令验证上述配置。

Switch A

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O IA    2004:4::/32 [110/3]
       via fe80::c629:f2ff:fe02:3600, eth-0-13, 00:01:00
C       2004:12:9::/96
       via ::, eth-0-9, 00:15:56
C       2004:12:9::1/128
       via ::1, eth-0-9, 00:15:56
C       2004:13:13::/96
       via ::, eth-0-13, 00:15:55
C       2004:13:13::2/128
       via ::1, eth-0-13, 00:15:55
O       2004:23:17::/96 [110/2]
       via fe80::bc22:aeff:fe64:aa00, eth-0-9, 00:08:10
       via fe80::c629:f2ff:fe02:3600, eth-0-13, 00:08:10
C       fe80::/10
       via ::, Null0, 00:15:57
```

Switch B

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O IA    2004:4::/32 [110/3]
       via fe80::c629:f2ff:fe02:3600, eth-0-17, 00:00:57
C       2004:12:9::/96
       via ::, eth-0-9, 00:12:24
C       2004:12:9::2/128
```

```
       via ::1, eth-0-9, 00:12:24
O       2004:13:13::/96 [110/2]
       via fe80::b242:55ff:fe05:ff00, eth-0-9, 00:07:52
       via fe80::c629:f2ff:fe02:3600, eth-0-17, 00:07:52
C       2004:23:17::/96
       via ::, eth-0-17, 00:12:24
C       2004:23:17::1/128
       via ::1, eth-0-17, 00:12:24
C       fe80::/10
       via ::, Null0, 00:12:26
```

Switch C

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O       2004:4::/32 [110/0]
       via ::, Null0, 00:08:31
O       2004:4:1::/96 [110/2]
       via fe80::ee66:91ff:fe45:db00, eth-0-9, 00:01:08
O       2004:4:2::/96 [110/2]
       via fe80::ee66:91ff:fe45:db00, eth-0-9, 00:01:08
O       2004:4:3::/96 [110/2]
       via fe80::ee66:91ff:fe45:db00, eth-0-9, 00:01:08
O       2004:4:4::/96 [110/2]
       via fe80::ee66:91ff:fe45:db00, eth-0-9, 00:01:08
C       2004:4:100::/96
       via ::, eth-0-9, 00:08:32
C       2004:4:100::1/128
       via ::1, eth-0-9, 00:08:32
O       2004:12:9::/96 [110/2]
       via fe80::b242:55ff:fe05:ff00, eth-0-13, 00:08:03
       via fe80::bc22:aeff:fe64:aa00, eth-0-17, 00:08:03
O       2004:13:13::/96 [110/1]
       via fe80::b242:55ff:fe05:ff00, eth-0-13, 00:08:18
C       2004:23:17::/96
       via ::, eth-0-17, 00:08:32
C       2004:23:17::2/128
       via ::1, eth-0-17, 00:08:32
C       fe80::/10
       via ::, Null0, 00:08:34
```

Switch D

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O IA    ::/0 [110/2]
       via fe80::c629:f2ff:fe02:3600, eth-0-9, 00:00:53
C       2004:4:1::/96
       via ::, eth-0-1, 00:03:09
C       2004:4:1::1/128
       via ::1, eth-0-1, 00:03:09
C       2004:4:2::/96
       via ::, eth-0-2, 00:03:08
C       2004:4:2::1/128
       via ::1, eth-0-2, 00:03:08
C       2004:4:3::/96
       via ::, eth-0-3, 00:03:08
C       2004:4:3::1/128
       via ::1, eth-0-3, 00:03:08
C       2004:4:4::/96
       via ::, eth-0-4, 00:03:09
C       2004:4:4::1/128
       via ::1, eth-0-4, 00:03:09
C       2004:4:100::/96
       via ::, eth-0-9, 00:03:09
C       2004:4:100::2/128
       via ::1, eth-0-9, 00:03:09
C       fe80::/10
       via ::, Null0, 00:03:10
```

11.2.7 配置 OSPF 重分布路由

区域内和区域间路由描述的是 AS 内部的网络结构，外部路由则描述了应该如何选择到 AS 以外目的地址的路由。OSPF 将引入的 AS 外部路由分为两类：Type1 和 Type2。

第一类外部路由是指接收的是 IGP（Interior Gateway Protocol，内部网关协议）路由（例如静态路由和 RIPng 路由）。由于这类路由的可信程度较高，并且和 OSPFv3 自身路由的开销具有可比性，所以到第一类外部路由的开销等于本路由器到相应的 ASBR 的开销与 ASBR 到该路由目的地址的开销之和。

第二类外部路由是指接收的是 EGP（Exterior Gateway Protocol，外部网关协议）路由。由于这类路由的可信度比较低，所以 OSPFv3 协议认为从 ASBR 到自治系统之外的开销远远大于在自治系统之内到达 ASBR 的开销。所以计算路由开销时将主要考虑前者，即到第二类外部路由的开销等于 ASBR 到该路由目的地址的开销。如果计算出开销值相等的两条路由，再考虑本路由器到相应的 ASBR 的开销。下面例子 RIP 路由将作为外部路由被重分布到 OSPFv3 网络中。

I. 拓扑

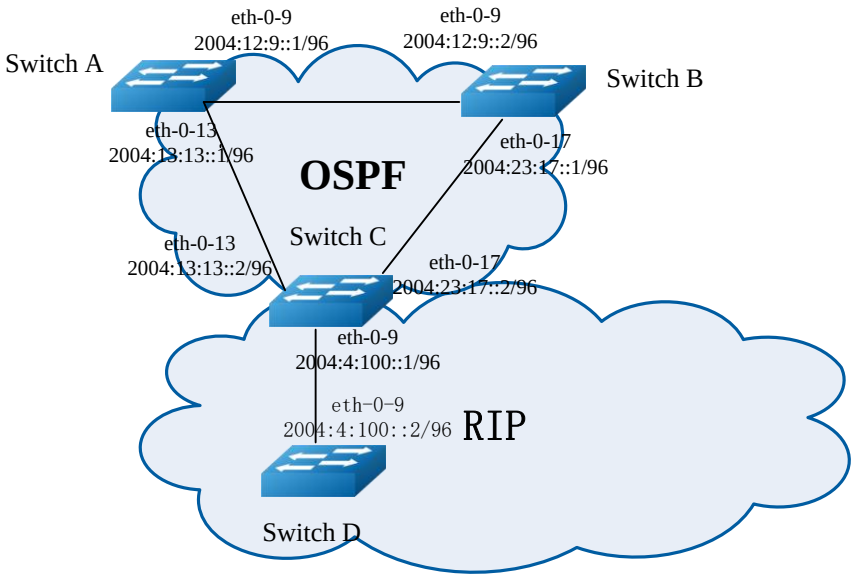


图11-5 OSPF v3 路由重分布

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 100	创建 OSPFv3 进程号 100
Switch(config-router)# router-id 1.1.1.1	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 100 area 0 instance 0	将该端口加入到 OSPFv3 process100，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.

Switch(config)#interface eth-0-13	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:13:13::2/96	设置端口的 IP 地址
Switch(config-if)# ipv6 router ospf 100 area 0 instance 0	将该端口加入到 OSPFv3 process100，area0， instance0 中
Switch(config-if)# end	退出配置模式

Switch B

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 200	创建 OSPFv3 进程号 200
Switch(config-router)# router-id 2.2.2.2	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 200 area 0 instance 0	将该端口加入到 OSPFv3 process200，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)#interface eth-0-17	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:23:17::1/96	设置端口的 IP 地址

Switch(config-if)# ipv6 router ospf 200 area 0 instance 0	将该端口加入到 OSPFv3 process200，area0， instance0 中
Switch(config-if)# end	退出配置模式

Switch C

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 300	创建 OSPFv3 进程号 300
Switch(config-router)# router-id 3.3.3.3	指定 Router ID
Switch(config-router)# redistribute ripng	重发布 ripng 到 OSPF 中
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-13	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:13:13::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 300 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)# interface eth-0-17	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:23:17::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 300 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)# router ipv6 rip	使能 ripng

Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:100::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router rip	将该端口加入到 RIPng 路由域中
Switch(config-if)# end	退出配置模式

Switch D

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 rip	使能 ripng
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:100::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router rip	将该端口加入到 RIPng 路由域中
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:4:1::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router rip	将该端口加入到 RIPng 路由域中
Switch(config-if)# end	退出配置模式

III. 命令验证

使用如下命令，验证上述配置：

show ipv6 ospf database external

show ipv6 route

Switch A

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O E2    2004:4:1::/96 [110/20]
       via fe80::c629:f2ff:fe02:3600, eth-0-13, 00:00:03
C       2004:12:9::/96
       via ::, eth-0-9, 00:34:20
C       2004:12:9::1/128
       via ::1, eth-0-9, 00:34:20
C       2004:13:13::/96
       via ::, eth-0-13, 00:34:19
C       2004:13:13::2/128
       via ::1, eth-0-13, 00:34:19
O       2004:23:17::/96 [110/2]
       via fe80::bc22:aeff:fe64:aa00, eth-0-9, 00:26:34
       via fe80::c629:f2ff:fe02:3600, eth-0-13, 00:26:34
C       fe80::/10
       via ::, Null0, 00:34:21
```

Switch# show ipv6 ospf database external

```
            OSPFv3 Router with ID (1.1.1.1) (Process 100)
                AS-external-LSA
  LS age: 140
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.1
  Advertising Router: 3.3.3.3
  LS Seq Number: 0x80000001
  Checksum: 0x66F7
  Length: 44
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2004:4:1::/96
    Prefix Options: 0 (-|-|-|-)
    External Route Tag: 0
```

Switch B

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O E2    2004:4:1::/96 [110/20]
       via fe80::c629:f2ff:fe02:3600, eth-0-17, 00:02:43
C       2004:12:9::/96
       via ::, eth-0-9, 00:33:31
C       2004:12:9::2/128
       via ::1, eth-0-9, 00:33:31
O       2004:13:13::/96 [110/2]
       via fe80::b242:55ff:fe05:ff00, eth-0-9, 00:28:59
       via fe80::c629:f2ff:fe02:3600, eth-0-17, 00:28:59
C       2004:23:17::/96
       via ::, eth-0-17, 00:33:31
C       2004:23:17::1/128
       via ::1, eth-0-17, 00:33:31
C       fe80::/10
       via ::, Null0, 00:33:33
```

Switch# show ipv6 ospf database external

```
show ipv6 ospf database external
            OSPFv3 Router with ID (2.2.2.2) (Process 200)
                AS-external-LSA
  LS age: 195
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.1
  Advertising Router: 3.3.3.3
  LS Seq Number: 0x80000001
  Checksum: 0x66F7
  Length: 44
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2004:4:1::/96
    Prefix Options: 0 (-|-|-|-)
    External Route Tag: 0
```

Switch C

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
```

```
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
R       2004:4:1::/96 [120/2]
       via fe80::ee66:91ff:fe45:db00, eth-0-9, 00:03:43
C       2004:4:100::/96
       via ::, eth-0-9, 00:07:01
C       2004:4:100::1/128
       via ::1, eth-0-9, 00:07:01
O       2004:12:9::/96 [110/2]
       via fe80::b242:55ff:fe05:ff00, eth-0-13, 00:29:57
       via fe80::bc22:aeff:fe64:aa00, eth-0-17, 00:29:57
O       2004:13:13::/96 [110/1]
       via fe80::b242:55ff:fe05:ff00, eth-0-13, 00:30:12
C       2004:23:17::/96
       via ::, eth-0-17, 00:30:26
C       2004:23:17::2/128
       via ::1, eth-0-17, 00:30:26
C       fe80::/10
       via ::, Null0, 00:30:28
```

Switch# show ipv6 ospf database external

```
show ipv6 ospf database external
            OSPFv3 Router with ID (3.3.3.3) (Process 300)
                AS-external-LSA
  LS age: 250
  LS Type: AS-External-LSA
  Link State ID: 0.0.0.1
  Advertising Router: 3.3.3.3
  LS Seq Number: 0x80000001
  Checksum: 0x66F7
  Length: 44
    Metric Type: 2 (Larger than any link state path)
    Metric: 20
    Prefix: 2004:4:1::/96
    Prefix Options: 0 (-|-|-|-)
    External Route Tag: 0
```

Switch D

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
C       2004:4:1::/96
```

```
        via ::, eth-0-1, 00:04:48
C        2004:4:1::1/128
        via ::1, eth-0-1, 00:04:48
C        2004:4:100::/96
        via ::, eth-0-9, 00:06:59
C        2004:4:100::2/128
        via ::1, eth-0-9, 00:06:59
C        fe80::/10
        via ::, Null0, 00:07:00
```

11.2.8 配置 OSPFv3 Cost

你可以通过修改接口的 COST 值来使路由成为最优路由。在下面的例子中，通过修改 COST 值可以使 Switch B 成为 Switch A 的下一跳。

默认接口的 COST 值是 1(1000M speed)。Switch B 的 eth-0-17 优先级 100，Switch D 的 eth-0-9 优先级 150. 那么到达 Switch C 的网络 2004:3:1::/96 的 Cost 值将不一样：

Switch B: 1+1+100 = 102

Switch C: 1+1+150 = 152

I. 拓扑

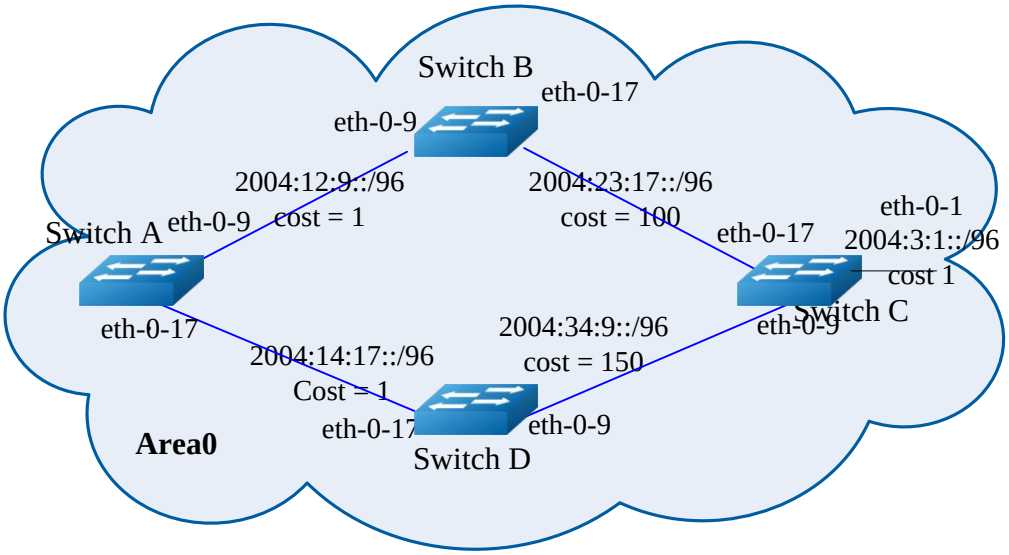


图11-6 OSPFv3 Cost

II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6

Switch(config)# router ipv6 ospf 100	创建 OSPFv3 进程号 100
Switch(config-router)# router-id 1.1.1.1	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 100 area 0 instance 0	将该端口加入到 OSPFv3 process100，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)#interface eth-0-17	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:14:17::1/96	设置端口的 IP 地址
Switch(config-if)# ipv6 router ospf 100 area 0 instance 0	将该端口加入到 OSPFv3 process100，area0， instance0 中
Switch(config-if)# end	退出配置模式

Switch B

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 200	创建 OSPFv3 进程号 200
Switch(config-router)# router-id 2.2.2.2	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性

Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:12:9::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 200 area 0 instance 0	将该端口加入到 OSPFv3 process200，area0， instance0 中
Switch(config-if)# end	退出配置模式
Switch# configure terminal	进入配置模式.
Switch(config)#interface eth-0-17	进入接口模式
Switch(config-if)#no switchport	设置接口为三层接口
Switch(config-if)#no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:23:17::1/96	设置端口的 IP 地址
Switch(config-if)# ipv6 router ospf 200 area 0 instance 0	将该端口加入到 OSPFv3 process200，area0， instance0 中
Switch(config-if)# ipv6 ospf cost 100	配置 OSPFv3 interface cost
Switch(config-if)# end	退出配置模式

Switch C

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 300	创建 OSPFv3 进程号 300
Switch(config-router)# router-id 3.3.3.3	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-17	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:23:17::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 300 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中

Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:34:9::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 300 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:3:1::1/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 300 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# end	退出配置模式

Switch D

Switch# configure terminal	进入配置模式
Switch (config)# ipv6 enable	使能 IPv6
Switch(config)# router ipv6 ospf 400	创建 OSPFv3 进程号 400
Switch(config-router)# router-id 4.4.4.4	指定 Router ID
Switch(config-router)# exit	退出 OSPFv3 配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:34:9::2/96	配置 IPv6 地址

Switch(config-if)# ipv6 router ospf 400 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# ipv6 ospf cost 150	配置 OSPFv3 interface cost
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-17	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2004:14:17::2/96	配置 IPv6 地址
Switch(config-if)# ipv6 router ospf 400 area 0 instance 0	将该端口加入到 OSPFv3 process300，area0， instance0 中
Switch(config-if)# end	退出配置模式

III. 命令验证

使用命令 **show ipv6 ospf route** 验证以上配置。

Switch A

Switch# show ipv6 ospf route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O       2004:3:1::/96 [110/102]
       via fe80::bc22:aeff:fe64:aa00, eth-0-9, 00:08:06
C       2004:12:9::/96
       via ::, eth-0-9, 01:15:43
C       2004:12:9::1/128
       via ::1, eth-0-9, 01:15:43
C       2004:14:17::/96
       via ::, eth-0-17, 00:18:38
C       2004:14:17::1/128
       via ::1, eth-0-17, 00:18:38
O       2004:23:17::/96 [110/101]
       via fe80::bc22:aeff:fe64:aa00, eth-0-9, 00:08:06
O       2004:34:9::/96 [110/102]
       via fe80::bc22:aeff:fe64:aa00, eth-0-9, 00:03:56
```

```
C       fe80::/10
        via ::, Null0, 01:15:44
```

Switch B

Switch# show ipv6 ospf route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O       2004:3:1::/96 [110/101]
       via fe80::c629:f2ff:fe02:3600, eth-0-17, 00:08:33
C       2004:12:9::/96
       via ::, eth-0-9, 01:12:40
C       2004:12:9::2/128
       via ::1, eth-0-9, 01:12:40
O       2004:14:17::/96 [110/2]
       via fe80::b242:55ff:fe05:ff00, eth-0-9, 00:18:43
C       2004:23:17::/96
       via ::, eth-0-17, 01:12:40
C       2004:23:17::1/128
       via ::1, eth-0-17, 01:12:40
O       2004:34:9::/96 [110/101]
       via fe80::c629:f2ff:fe02:3600, eth-0-17, 00:04:23
C       fe80::/10
       via ::, Null0, 01:12:42
```

Switch C

Switch# show ipv6 ospf route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
C       2004:3:1::/96
       via ::, eth-0-1, 00:13:54
C       2004:3:1::1/128
       via ::1, eth-0-1, 00:13:54
O       2004:12:9::/96 [110/2]
       via fe80::bc22:aeff:fe64:aa00, eth-0-17, 00:19:47
O       2004:14:17::/96 [110/2]
       via fe80::ee66:91ff:fe45:db00, eth-0-9, 00:02:27
C       2004:23:17::/96
```

```
       via ::, eth-0-17, 01:09:02
C       2004:23:17::2/128
       via ::1, eth-0-17, 01:09:02
C       2004:34:9::/96
       via ::, eth-0-9, 00:04:52
C       2004:34:9::1/128
       via ::1, eth-0-9, 00:04:52
C       fe80::/10
       via ::, Null0, 01:09:04
```

Switch D

Switch# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
O       2004:3:1::/96 [110/103]
       via fe80::b242:55ff:fe05:ff00, eth-0-17, 00:02:35
O       2004:12:9::/96 [110/2]
       via fe80::b242:55ff:fe05:ff00, eth-0-17, 00:02:35
C       2004:14:17::/96
       via ::, eth-0-17, 00:04:09
C       2004:14:17::2/128
       via ::1, eth-0-17, 00:04:09
O       2004:23:17::/96 [110/102]
       via fe80::b242:55ff:fe05:ff00, eth-0-17, 00:02:35
C       2004:34:9::/96
       via ::, eth-0-9, 00:06:06
C       2004:34:9::2/128
       via ::1, eth-0-9, 00:06:06
C       fe80::/10
       via ::, Null0, 00:44:59
```

11.2.9 配置监听 OSPFv3

您可以通过命令显示具体的统计数据，如 IPv6 路由表的内容，缓存和数据库。

Switch# show ipv6 ospf	显示 OSPF 进程信息

Switch # show ipv6 ospf database database-summary Switch # show ipv6 ospf database router Switch # show ipv6 ospf database network self-originate Switch # show ipv6 ospf database inter-router Switch # show ipv6 ospf database intra-prefix Switch # show ipv6 ospf database inter-prefix Switch # show ipv6 ospf database link Switch # show ipv6 ospf database external	显示 OSPF 链路状态信息库
Switch # show ipv6 ospf interface	显示 OSPFv3 接口信息
Switch # show ipv6 ospf neighbor	显示 OSPFv3 邻居信息

11.3 RIPng 配置

11.3.1 简介

RIPng（Routing Information Protocol Next Generation）是对原来的 IPv4 网络中 RIP-2 协议的扩展，大多数概念都可以用于 RIPng。

RIPng 是一种较为简单的内部网关协议（Interior Gateway Protocol，IGP），主要用于规模较小的网络中。

RIPng 是一种基于距离矢量（Distance-Vector）算法的协议，它通过 UDP 报文进行路由信息的交换。RIPng 使用跳数（Hop Count）来衡量到达目的地址的距离，称为路由权（RoutingCost）。在 RIPng 中，路由器到与它直接相连网络的跳数为 0，通过一个路由器可达的网络的跳数为 1，其余依此类推。为限制收敛时间，RIP 规定 cost 的取值为 0～15 之间的整数，cost 取值大于或等于 16 的跳数被定义为无穷大，即目的网络或主机不可达。

为提高性能，防止产生路由环，RIPng 支持水平分割（Split Horizon）。RIPng 还可引入其它路由协议所得到的路由。

为了在 IPv6 网络中应用，RIPng 对原有的 RIP 协议进行了修改：

- UDP 端口号：使用 UDP 的 521 端口发送和接收路由信息。
- 组播地址：使用 FF02::9 作为链路本地范围内的 RIPng 路由器组播地址
- 下一跳地址：使用 128 比特的 IPv6 地址

- 源地址：使用链路本地地址 FE80::/10 作为源地址发送 RIPng 路由信息更新报文。

11.3.2 参考文献

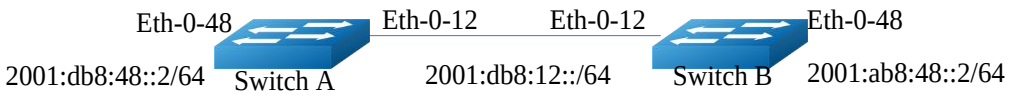
RIPng 模块是基于以下 RFC：

RFC 2080 – RIPng for IPv6

11.3.3 配置启用 RIPng

在两个交换机上启用 RIPng 路由协议的配置步骤如所示

I. 拓扑



II. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# router ipv6 rip	启用 RIPng
Switch(config-router)# exit	退出路由模式
Switch(config)# interface eth-0-12	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2001:db8:12::1/64	配置 IPv6 地址
Switch(config-if)# ipv6 router rip	该端口使能 RIPng
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-48	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2001:db8:48::2/64	配置 IPv6 地址
Switch(config-if)# ipv6 router rip	该端口使能 RIPng

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# router ipv6 rip	启用 RIPng
Switch(config-router)# exit	退出路由模式
Switch(config)# interface eth-0-12	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2001:db8:12::2/64	配置 IPv6 地址
Switch(config-if)# ipv6 router rip	该端口使能 RIPng
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-48	进入接口模式
Switch(config-if)# no switchport	使能三层接口属性
Switch(config-if)# no shutdown	Up 端口
Switch(config-if)# ipv6 address 2001:ab8:49::2/64	配置 IPv6 地址
Switch(config-if)# ipv6 router rip	该端口使能 RIPng

III. 命令验证

使用如下命令，验证上述配置：

show ipv6 rip database

show ipv6 rip interface

show ipv6 protocols rip

show ipv6 route rip

Switch A output

Switch# show ipv6 rip database

```
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, Ra - RIP aggregated,
       Rcx - RIP connect suppressed, Rsx - RIP static suppressed,
       K - Kernel, C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
   Network                     Next Hop                  If        Met Tag Time
R  2001:ab8:49::/64            fe80::1271:d1ff:fec8:3300 eth-0-12 5   0   00:02:34
Rc 2001:db8:12::/64            ::                        eth-0-12 1   0
```

```
Rc 2001:db8:48::/64            ::                          eth-0-48 1    0
```

Switch# show ipv6 rip interface

```
eth-0-12 is up, line protocol is up
  Routing Protocol: RIPng
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IPv6 interface address:
      2001:db8:12::1/64
      fe80::7e14:63ff:fe76:8900/10
eth-0-48 is up, line protocol is up
  Routing Protocol: RIPng
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IPv6 interface address:
      2001:db8:48::2/64
      fe80::7e14:63ff:fe76:8900/10
```

Switch# show ipv6 protocols rip

```
Routing Protocol is "ripng"
  Sending updates every 30 seconds with +/-5 seconds, next due in 7 seconds
  Timeout after 180 seconds, garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Default redistribute metric is 1
  Redistributing:
  Interface
    eth-0-12
    eth-0-48
  Routing for Networks:
  Number of routes (including connected): 3
  Distance: (default is 120)
```

Switch# show ipv6 route rip

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
R       2001:ab8:49::/64 [120/5]
       via fe80::1271:d1ff:fec8:3300, eth-0-12, 00:26:05
```

Switch B

Switch# show ipv6 rip database

```
Codes: R - RIP, Rc - RIP connected, Rs - RIP static, Ra - RIP aggregated,
       Rcx - RIP connect suppressed, Rsx - RIP static suppressed,
       K - Kernel, C - Connected, S - Static, O - OSPF, I - IS-IS, B - BGP
```

```
   Network                     Next Hop                  If        Met Tag Time
Rc 2001:ab8:49::/64            ::                        eth-0-48  1   0
Rc 2001:db8:12::/64            ::                        eth-0-12  1   0
R  2001:db8:48::/64            fe80::7e14:63ff:fe76:8900 eth-0-12  2   0   00:02:33
```

Switch# show ipv6 rip interface

```
eth-0-12 is up, line protocol is up
  Routing Protocol: RIPng
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IPv6 interface address:
      2001:db8:12::2/64
      fe80::1271:d1ff:fec8:3300/10
eth-0-48 is up, line protocol is up
  Routing Protocol: RIPng
    Passive interface: Disabled
    Split horizon: Enabled with Poisoned Reversed
    IPv6 interface address:
      2001:ab8:49::2/64
      fe80::1271:d1ff:fec8:3300/10
```

Switch# show ipv6 protocols rip

```
Routing Protocol is "ripng"
  Sending updates every 30 seconds with +/-5 seconds, next due in 13 seconds
  Timeout after 180 seconds, garbage collect after 120 seconds
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Outgoing routes will have 3 added to metric if on list ripng_acl
  Default redistribute metric is 1
  Redistributing:
  Interface
    eth-0-12
    eth-0-48
  Routing for Networks:
  Number of routes (including connected): 3
  Distance: (default is 120)
```

Switch# show ipv6 route rip

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, I - IS-IS, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       Dr - DHCPV6 Relay
       [*] - [AD/Metric]
Timers: Uptime
R       2001:db8:48::/64 [120/2]
       via fe80::7e14:63ff:fe76:8900, eth-0-12, 00:23:31
```

11.3.4 配置 Metric 参数

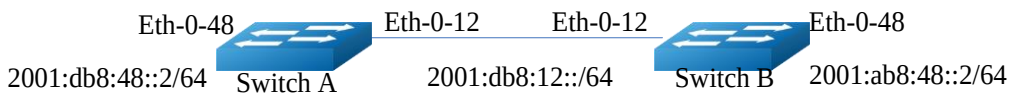
偏移度量值是附加在 RIPng 路由上的输入输出度量值，包括发送偏移度量值和接收偏移度量值。发送偏移度量值不会改变路由表中的路由度量值，仅当接口发送 RIP 路由信息时才会添加到发送路由上；接收偏移度量值会影响接收到的路由度量值，接口接收到一条合法的 RIP 路由时，在将其加入路由表前会把度量值附加到该路由上。偏移度量值一般包括如下的参数：

- 指定增加路由 Metric 的 ACL 参数说明如下。
  - **In**：应用在从邻居路由器学习到的 RIPng 的路由上
  - **Out**：应用在发布给邻居路由器 RIPng 通告上
- 匹配 ACL 路由的偏移值 Metric
- 应用偏移列表的接口

  如果一个路由匹配全局偏移表（不指定接口）和一个基于接口的偏移列表，此时基于接口的偏移列表优先。在这种情况下，基于接口的偏移列表的度量值是被加到路由上。

下面例子讲述如何在 SwitchA 上将 2001:db8:48::2/64 在 Eth-0-12 接口上增加 metric 3

I. 拓扑



II. 配置

Switch A configuration

Switch# show run

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::1/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:db8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
```

Switch B configuration

Switch# show run

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::2/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:ab8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
```

Validation route table on Switch B

Switch# show ipv6 route rip

```
R       2001:db8:48::/64 [120/2]
        via fe80::7e14:63ff:fe76:8900, eth-0-12, 00:44:47
```

Switch A

Switch# configure terminal	进入配置模式
Switch(config)#ipv6 access-list ripngoffset	创建 ACL.
Switch(config-ipv6-acl)# permit any 2001:db8:48::/64 any	匹配相应的网段
Switch(config-ipv6-acl)# router ipv6 rip	启用 RIPng 路由协议
Switch(config-router)# offset-list ripngoffset out 3 eth-0-12	设置偏移列表的 Metric 值

III. 命令验证

Switch B output

Switch# show ipv6 route rip

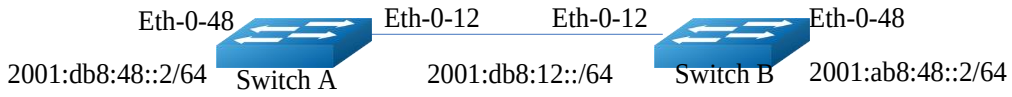
```
R       2001:db8:48::/64 [120/5]
        via fe80::7e14:63ff:fe76:8900, eth-0-12, 00:00:07
```

11.3.5 配置管理距离

默认情况下，RIPng 的管理距离是 120。比较路由时，管理距离越低，路由越容易被选中。

下面例子讲述了如何修改 RIPng 的管理距离。

I. 拓扑



II. 配置

Switch A configuration

Switch# show running-config

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::1/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:db8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
```

Switch B configuration

Switch# show running-config

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::2/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:ab8:48::2/64
 ipv6 router rip
!
```

```
router ipv6 rip
!
```

Validation route table on Switch B

Switch# show ipv6 route rip

```
R       2001:db8:48::/64 [120/2]
        via fe80::7e14:63ff:fe76:8900, eth-0-12, 00:44:47
```

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# router ipv6 rip	启用 RIPng 路由协议
Switch(config-router)# distance 100	设置 RIPng 路由的管理距离为 100

III. 命令验证

Switch B output

Switch# show ipv6 route rip

```
R       2001:db8:48::/64 [100/5]
        via fe80::7e14:63ff:fe76:8900, eth-0-12, 00:00:09
```

11.3.6 配置重分布

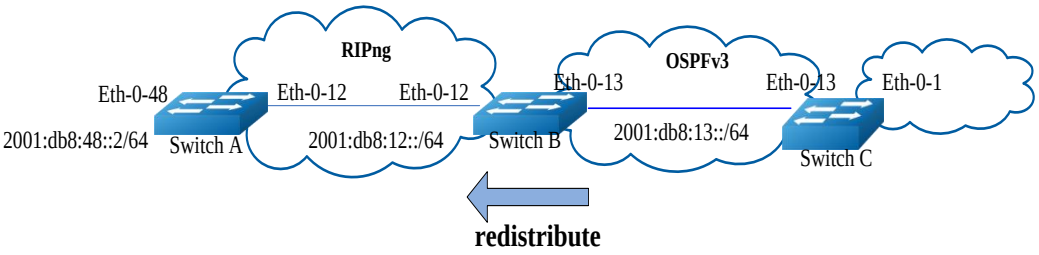
你可以将静态路由，直连路由以及其他路由协议比如 OSPFv3 的路由重分布到 RIP 中并被 RIPng 发送给它的邻居。

默认 RIPng 的重发布 Metric 为 1，最大 16。

将特定的路由重发布到 RIPng 上，其度量值可以是默认的，也可以是修改后的。

下面例子讲述如何重分布其他的路由信息到 RIPng。

I. 拓扑



II. 配置

Switch A configuration

Switch# show running-config

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::1/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:db8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
```

Switch B configuration

Switch# show running-config

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::2/64
 ipv6 router rip
!
interface eth-0-13
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:13::1/64
 ipv6 router ospf area 0
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
```

```
 ipv6 address 2001:ab8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
router ipv6 ospf
 router-id 1.1.1.1
```

Switch C configuration

Switch# show running-config

```
interface eth-0-1
 no switchport
  ipv6 address auto link-local
 ipv6 address 2001:db8:1::1/64
 ipv6 router ospf area 0
!
interface eth-0-13
 no switchport
 ipv6 address 2001:db8:13::2/64
 ipv6 router ospf area 0
!
router ipv6 ospf
 router-id 2.2.2.2
!
```

Validation route table on Switch A

Switch# show ipv6 route rip

```
R        2001:ab8:48::/64 [120/5]
         via fe80::1271:d1ff:fec8:3300, eth-0-12, 01:43:37
```

Validation route table on Switch B

Switch# show ipv6 route

```
O        2001:db8:1::/64 [110/2]
         via fe80::5c37:1dff:febe:2d00, eth-0-13, 00:31:17
R        2001:db8:48::/64 [100/5]
         via fe80::7e14:63ff:fe76:8900, eth-0-12, 00:49:57
```

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# router ipv6 rip	启用 RIPng 路由协议
Switch(config-router)#default-metric 2	指定默认的 Metric
Switch(config-router)#redistribute ospfv3 metric 5	重分布 OSPFv3 路由到 RIPng 中

III. 命令验证

Switch A output

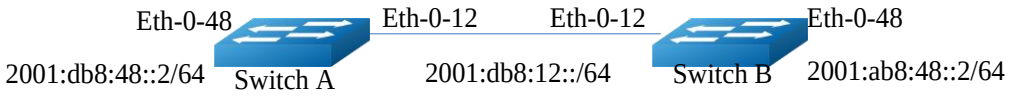
Switch# show ipv6 route rip

```
R       2001:ab8:48::/64 [120/5]
        via fe80::1271:d1ff:fec8:3300, eth-0-12, 01:48:23
R       2001:db8:1::/64 [120/6]
        via fe80::1271:d1ff:fec8:3300, eth-0-12, 00:00:19
```

11.3.7 配置水平分割参数

通常情况下，连接到组播网络并且使用距离矢量路由协议的路由器，使用水平分割机制来避免环路。配置水平分割可以使得从一个接口学到的路由不能通过此接口向外发布，这通常优化了多个路由器之间的通信，尤其在链路中断时。配置毒性逆转可以使得从一个接口学到的路由还可以从这个接口向外发布，但这些路由的度量值已设置为16，即不可达。

I. 拓扑



II. 配置

Switch A configuration

Switch# show running-config

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::1/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:db8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
```

Switch B configuration

Switch# show running-config

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::2/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:ab8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
```

Switch B debug configuration

Switch# debug ipv6 rip packet send detail

Switch# terminal monitor

Disable Split-horizon on Switch B

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-12	配置接口 eth-0-12
Switch(config-if)# no ipv6 rip split-horizon	禁用水平分割

Oct 24 10:00:06 Switch RIPNG6-7: SEND[eth-0-12]: Send to [ff02::9]:521

Oct 24 10:00:06 Switch RIPNG6-7: SEND[eth-0-12]: RESPONSE version 1 packet size 64

Oct 24 10:00:06 Switch RIPNG6-7: 2001:ab8:49::/64 metric 4 tag 0

Oct 24 10:00:06 Switch RIPNG6-7: 2001:db8:12::/64 metric 1 tag 0

Oct 24 10:00:06 Switch RIPNG6-7: 2001:db8:48::/64 metric 5 tag 0

Enable Split-horizon on Switch B

Switch# configure terminal	进入配置模式
Switch(config)#interface eth-0-12	配置接口 eth-0-12
Switch(config-if)# ipv6 rip split-horizon	启用水平分割

Oct 24 10:05:16 Switch RIPNG6-7: SEND[eth-0-12]: Send to [ff02::9]:521

Oct 24 10:05:16 Switch RIPNG6-7: SEND[eth-0-12]: RESPONSE version 1 packet size 44

Oct 24 10:05:16 Switch RIPNG6-7: 2001:ab8:49::/64 metric 4 tag 0

Oct 24 10:05:16 Switch RIPNG6-7: 2001:db8:12::/64 metric 1 tag 0

III. 命令验证

使用如下命令，验证上述配置：

show running-config

show ipv6 rip interface

11.3.8 配置 Timer

RIPng 受多个定时器的控制，比如路由更新的频率，路由失效的时间等等。您可以调整这些计时器以调整 RIPng 的性能，以更好地满足您的互联网工作的需要。如下参数可供调整：

- Update 定时器，定义了发送更新报文的时间间隔。
- Timeout 定时器，定义了路由老化时间。如果在老化时间内没有收到关于某条路由的更新报文，则该条路由在路由表中的度量值将会被设置为 16。
- Garbage-Collect 定时器，定义了一条路由从度量值变为 16 开始，直到它从路由表里被删除所经过的时间。

I. 配置

使用如下表所示的命令配置 Timer

Switch# configure terminal	进入配置模式
Switch(config)# router ipv6 rip	启用 RIPng 路由协议
Switch(config-router)# timers basic 10 180 120	指定路由表 update timer 10 秒，指定路由信息超时 180 秒，垃圾信息收集时间 120 秒

II. 命令验证

使用如下命令，验证上述配置：

show running-config

show ipv6 protocols rip

Switch# show ipv6 protocols rip

```
Routing Protocol is "ripng"
  Sending updates every 10 seconds with +/-5 seconds, next due in 5 seconds
  Timeout after 180 seconds, garbage collect after 120 seconds
```

```
  Outgoing update filter list for all interface is not set
  Incoming update filter list for all interface is not set
  Outgoing routes will have 3 added to metric if on list ripng_acl
  Default redistribute metric is 2
  Redistributing:
  Interface
    eth-0-12
    eth-0-48
  Routing for Networks:
  Number of routes (including connected): 3
  Distance: (default is 100)
```

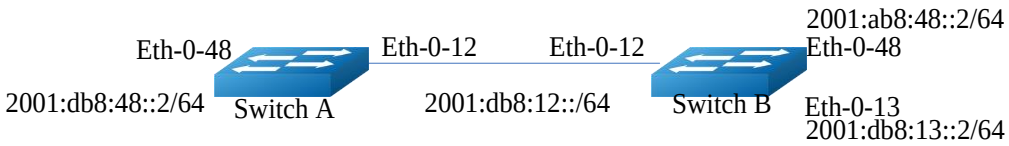
11.3.9 配置 RIPng 路由过滤列表

路由器提供路由信息过滤功能，通过指定访问控制列表和地址前缀列表，可以配置入口或出口过滤策略，对接收或发布的路由进行过滤。一个路由过滤列表通常包括如下参数：

- 一个被用作过滤器的 ACL 或 prefix list。
- **In 方向**：过滤器被应用在学习到的路由上；**Out 方向**：过滤器被应用在发布的路由上。

应用过滤器的接口（可选）。

I. 拓扑



II. 配置

Switch A configuration

Switch# show running-config

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::1/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:db8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
```

Switch B configuration

Switch# show running-config

```
interface eth-0-12
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:12::2/64
 ipv6 router rip
!
interface eth-0-13
 no switchport
 ipv6 address auto link-local
 ipv6 address 2001:db8:13::1/64
 ipv6 router rip
!
interface eth-0-48
 no switchport
 ipv6 nd ra mtu suppress
 ipv6 address auto link-local
 ipv6 address 2001:ab8:48::2/64
 ipv6 router rip
!
router ipv6 rip
!
```

Switch A output

Switch# show ipv6 route rip

```
R       2001:ab8:48::/64 [120/5]
        via fe80::1271:d1ff:fec8:3300, eth-0-12, 00:18:29
R       2001:db8:13::/64 [120/2]
        via fe80::1271:d1ff:fec8:3300, eth-0-12, 00:03:37
```

参照如下表中的命令，配置交换机 B。

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 prefix-list ripngfilter seq 5 deny 2001:db8:48::/64 Switch(config)# ipv6 prefix-list ripngfilter seq 10 permit any	建立列表
Switch(config)# router ipv6 rip	启用 RIPng 路由协议
Switch(config-router)# distribute-list prefix ripngfilter out eth-0-12	应用策略

III. 命令验证

Switch A output

Switch# show ipv6 route rip

```
R       2001:db8:13::/64 [120/2]
        via fe80::1271:d1ff:fec8:3300, eth-0-12, 00:03:37
```

11.4 Ipv6 Prefix-list 配置

11.4.1 简介

路由策略（Routing Policy）是为了改变网络流量所经过的途径而修改路由信息的技术，主要通过改变路由属性（包括可达性）来实现。地址前缀列表是路由策略的一种，作用比较灵活。一个地址前缀列表由前缀列表名标识。每个前缀列表可以包含多个表项，每个表项可以独立指定一个网络前缀形式的匹配范围，并用一个索引号来标识，索引号指明了进行匹配检查的顺序。在匹配的过程中，交换机按升序依次检查由索引号标识的各个表项。只要有某一表项满足条件，就意味着本次匹配过程结束，而不再进行下一个表项的匹配。

11.4.2 基础配置

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 prefix-list test seq 1 deny 2001:db8::1/32 le 48	创建地址前缀列表 test，并创建一条表项，指定序号为 1
Switch(config)# ipv6 prefix-list test permit any	创建一个表项为了防止不匹配条目出现时候遭遇拒绝
Switch(config)# ipv6 prefix-list test description this ipv6 prefix list is fot test	添加地址前缀列表描述
Switch(config)# ipv6 prefix-list test permit 2001:abc::1/32 le 48	创建一条表项，使用默认序号
Switch(config)# exit	退出全局模式

II. 命令验证

Switch# show ipv6 prefix-list detail

```
Prefix-list list number: 1
Prefix-list entry number: 3
Prefix-list with the last deletion/insertion: test
```

```
ipv6 prefix-list test:
   Description: this ipv6 prefix list is fot test
   count: 3, range entries: 0, sequences: 1 - 10
     seq 1 deny 2001:db8::1/32 le 48 (hit count: 0, refcount: 0)
     seq 5 permit any (hit count: 0, refcount: 0)
     seq 10 permit 2001:abc::1/32 le 48 (hit count: 0, refcount: 0)
```

11.4.3 配置 RIPng 简单应用

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 prefix-list aa seq 11 deny 2001:db8::1/32 le 48	创建地址前缀列表 aa，并创建一条表项
Switch(config)# ipv6 prefix-list aa permit any	创建一个表项为了防止不匹配条目出现时候遭遇拒绝
Switch(config)# router ipv6 rip	进入 Ripng 路由模式
Switch(config-router)# distribute-list prefix aa out	应用策略
Switch(config-router)# end	退出 Ripng 路由模式

II. 命令验证

Switch# show ipv6 prefix-list

```
ipv6 prefix-list aa: 2 entries
     seq 11 deny 1:db8::1/32 le 48
     seq 15 permit any
```

Switch# show running-config

```
Building configuration...
…
ipv6 prefix-list aa seq 11 deny 1:db8::1/32 le 48
ipv6 prefix-list aa seq 15 permit any
…
router ipv6 rip
distribute-list prefix aa out
```

11.4.4 配置 Route-map 简单应用

I. 配置 ipv6 prefix-list 应用到 route-map

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 prefix-list ripng_pre_1 seq 11 permit fe80::a8f0:d8ff:fe7d:c501/128	创建地址前缀列表 ripng_pre_1，并创建一条表项

Switch(config)# ipv6 prefix-list ripng_pre_1 permit any	创建一个表项为了防止不匹配条目出现时候遭遇拒绝
Switch(config)# route-map ripng_rmap permit	创建 route-map
Switch(config-route-map)# match ipv6 address prefix-list ripng_pre_1	匹配地址前缀列表 ripng_rmap
Switch(config-route-map)# set local-preference 200	设置行为
Switch(config-route-map)# exit	退出路由模式
Switch(config)# router ipv6 rip	进入 RIPng 路由模式
Switch(config-router)# redistribute static route-map ripng_rmap	配置重发布 static 路由
Switch(config-router)# end	退出 RIPng 模式

II. 命令验证

Switch # show route-map

```
route-map ripng_rmap, permit, sequence 10
  Match clauses:
    ipv6 next-hop prefix-list ripng_pre_1
  Set clauses:
    ipv6 next-hop local fe80::1
```

Switch # show running-config

```
Building configuration...
…
ipv6 prefix-list ripng_pre_1 seq 11 permit fe80::a8f0:d8ff:fe7d:c501/128
ipv6 prefix-list ripng_pre_1 seq 15 permit any
!
!
route-map ripng_rmap permit 10
 match ipv6 next-hop prefix-list ripng_pre_1
 set ipv6 next-hop local fe80::1
!

router ipv6 rip

 redistribute static route-map ripng_rmap

!

ipv6 route 2001:dbc::/64 fe80::a8f0:d8ff:fe7d:c501 eth-0-9

!
```

Switch# show ipv6 rip database

```
S  2001:dbc::/64              fe80::1                   eth-0-9  1   0
```

12 IPv6 业务配置指导

12.1 IPv6 over IPv4 隧道配置

12.1.1 简介

隧道技术是一种封装技术，它利用一种网络协议来传输另一种网络协议，即一种网络协议将其他网络协议的数据报文封装在自己的报文中，然后在网络中传输。封装后的数据报文在网络中传输的路径，称为隧道。隧道是一条虚拟的点对点连接，隧道的两端需要对数据报文进行封装及解封装。隧道技术就是指包括数据封装、传输和解封装在内的全过程。

在 IPv4 Internet 向 IPv6 Internet 过渡的初期，IPv4 网络已被大量部署，而 IPv6 网络只是散布在世界各地的一些孤岛。在 IPv4 网络上用于连接 IPv6 孤岛的隧道，称为 IPv6 over IPv4 隧道， 即 IPv6 报文被封装在 IPv4 报文中，实现 IPv6 报文的透明传输。为了实现 IPv6 over IPv4 隧道，需要在 IPv4 网络与 IPv6 网络交界的边界交换机上启动 IPv4/IPv6 双协议栈。

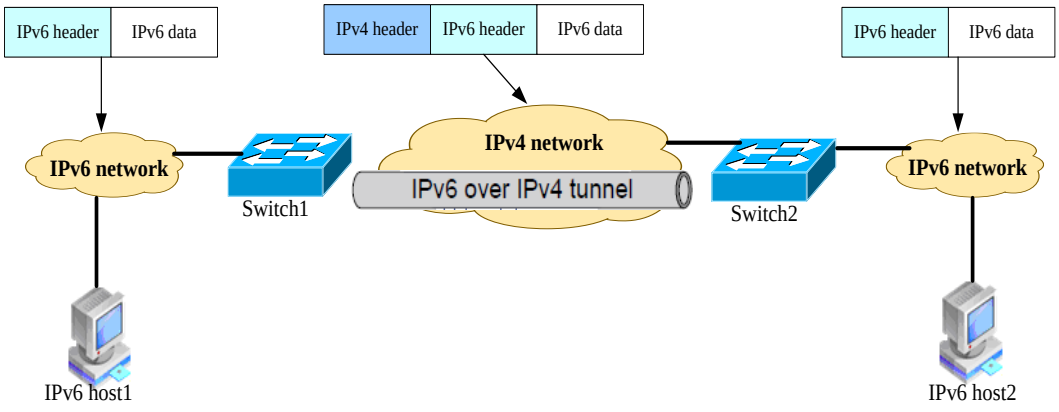


图12-1 ： IPv6 over IPv4 隧道原理图

IPv6 over IPv4 隧道对报文的处理过程如下：

- IPv6 网络中的设备发送 IPv6 报文，该报文到达隧道的源端设备 Switch1。
- Switch1 根据路由表判定该报文要通过隧道进行转发后，在 IPv6 报文前封装上 IPv4 的报文头，通过隧道的实际物理接口将报文转发出去。

- 封装报文通过隧道到达隧道目的端设备 Switch2，Switch2 判断该封装报文的目的地是本设备后，将对报文进行解封装。
- Switch2 根据解封装后的 IPv6 报文的目的地址转发该 IPv6 报文。如果目的地就是本设备，则将 IPv6 报文转给上层协议处理。

  这种技术的优点是，不用把所有的设备都升级为双栈，只要求 IPv4/IPv6 网络的边缘设备实现双栈和隧道功能。除边缘节点外，其它节点不需要支持双协议栈。可以大大利用现有的 IPv4 网络投资。

根据隧道终点的 IPv4 地址的获取方式不同，隧道分为“配置隧道”和“自动隧道”。

- 如果 IPv6 over IPv4 隧道的终点地址不能从 IPv6 报文的目的地址中自动获取，需要进行手工配置，这样的隧道称为“配置隧道”。
- 如果 IPv6 over IPv4 隧道的终点地址采用内嵌 IPv4 地址的特殊 IPv6 地址形式，则可以从 IPv6 报文的目的地址中自动获取隧道终点的 IPv4 地址，这样的隧道称为“自动隧道”。

  目前，常用的 IPv6 over IPv4 隧道模式有以下几种。

  - IPv6 over IPv4 手动隧道
  - 6to4 隧道
  - ISATAP 隧道

II. IPv6 over IPv4 **手动隧道**

IPv6 手工配置隧道的源和目的地址是手工指定的，它提供了一个点到点的连接。IPv6 手工配置隧道可以建立在两个边界路由器之间为被 IPv4 网络分离的 IPv6 网络提供稳定的连接，或建立在终端系统与边界路由器之间为终端系统访问 IPv6 网络提供连接。隧道的端点设备必须支持 IPv6/IPv4 双协议栈。其它设备只需实现单协议栈即可。

IPv6 手工配置隧道要求在设备上手工配置隧道的源地址和目的地址，如果一个边界设备要与多个设备建立手工隧道，就需要在设备上配置多个隧道。所以手工隧道通常用于两个边界路由器之间，为两个 IPv6 网络提供连接。

III. 6to4 **隧道**

- 普通 6to4 隧道

  6to4 隧道是点到多点的自动隧道，主要用于将多个 IPv6 孤岛通过 IPv4 网络连接到 IPv6 网络。6to4 隧道通过在 IPv6 报文的目的地址中嵌入 IPv4 地址，来实现自动获取隧道终点的 IPv4 地址。

  6to4 隧道使用了一种特殊的 IPv6 地址，即 6to4 地址，其格式为：

  2002:IPv4 地址:子网 ID:接口 ID

  6to4 地址的前缀是 2002:IPv4 地址，前缀长度为 48bits。其中 IPv4 地址是为 IPv6 孤岛申请的一个全球唯一的 IPv4 地址。在 IPv6/IPv4 边界交换机与 IPv4 网络链接的物理接口上必须配置该 IPv4 地址。子网 ID 的长度为 16bits，接口 ID 的长度为 64bits，均由用户在 IPv6 孤岛内分配。
- 6to4 中继

6to4 隧道只能用于前缀为 2002::/16 的 6to4 网络之间的通信，但在 IPv6 网络中也会使用像 2001::/16 这样的 IPv6 网络地址。为了实现 6to4 网络和其它 IPv6 网络的通信，必须有一台 6to4 路由器作为网关转发到 IPv6 网络的报文，这台路由器就叫做 6to4 中继（6to4 relay）路由器。（如果 IPv6 报文的目的地址不是 6to4 地址，但下一跳是 6to4 地址，则从下一跳地址中取出 IPv4 地址做为隧道的目的地址。）

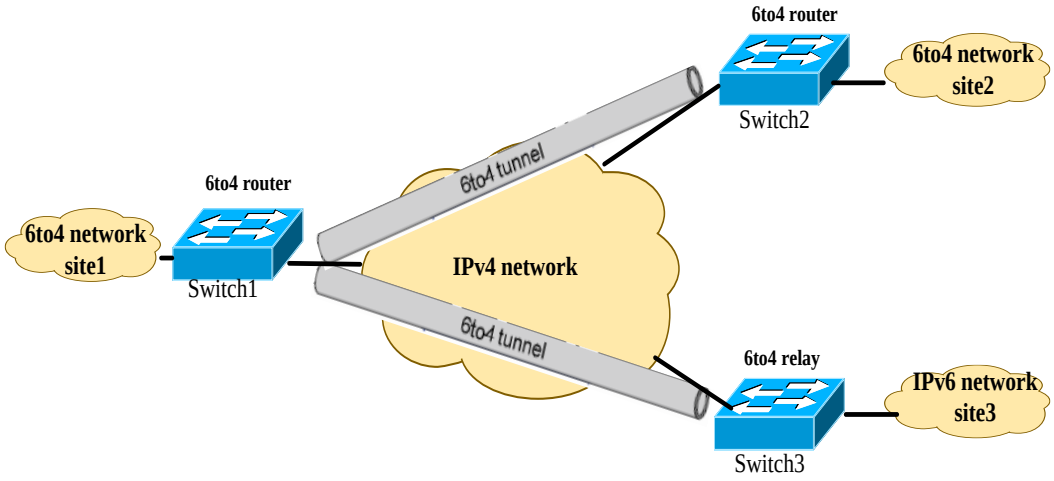


图12-2 ：6to4 隧道示意图

如上图所示，IPv6 报文在到达边界路由器后，根据报文的 IPv6 目的地址查找转发表，如果出接口是 6to4 自动隧道的 Tunnel 虚接口，且报文的目的地址是 6to4 地址或下一跳是 6to4 地址，则从 6to4 地址中取出 IPv4 地址做为隧道报文的目的地址，隧道报文的源地址是 Tunnel 接口上配置的。

IV. ISATAP 隧道

随着 IPv6 技术的推广，现有的 IPv4 网络中将会出现越来越多的 IPv6 主机，ISATAP 隧道技术为这种应用提供了一个较好的解决方案。ISATAP 隧道是点到多点的自动隧道技术，通过在 IPv6 报文的目的地址中嵌入的 IPv4 地址，可以自动获取隧道的终点。

使用 ISATAP 隧道时，IPv6 报文的目的地址和隧道接口的 IPv6 地址都要采用特殊的 ISATAP 地址。

Prefix(64bit)::5EFE:IPv4-Address

在创建 ISATAP 隧道时，由于 IPv4/IPv6 主机和 ISATAP 交换机在同一个 IPv4 网络里，ISATAP 地址中嵌入的 IPv4 地址可以是公网地址，也可以是私网地址。ISATAP 隧道主要用于在 IPv4 网络中 IPv6 路由器—IPv6 路由器、IPv6 主机—IPv6 路由器的连接。

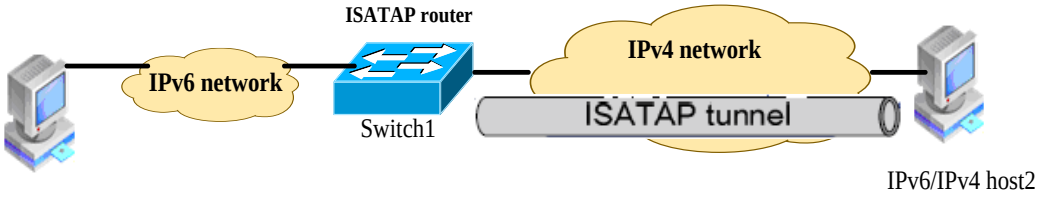


图12-3 ：ISATAP 隧道示意图

如上图所示，IPv4/IPv6 主机获得 IPv6 地址的过程如下：

步骤 1 IPv4/IPv6 主机发送交换机请求消息 IPv4/IPv6 主机使用 ISATAP 格式的链路本地地址向 ISATAP 交换机发送交换机请求消息，该交换机请求消息被封装在 IPv4 报文中。

步骤 2 ISATAP 交换机响应请求 ISATAP 交换机使用交换机通告消息响应主机的交换机请求。交换机通告消息中包含 ISATAP 前缀（ISATAP 前缀在交换机上通过人工配置）。

步骤 3 IPv4/IPv6 主机得到自己的 IPv6 地址 IPv4/IPv6 主机将 ISATAP 前缀与 5EFE:IPv4-Address 组合得到自己的 IPv6 地址，并用此地址访问 IPv6 主机。

12.1.2 配置手工隧道

I. 拓扑

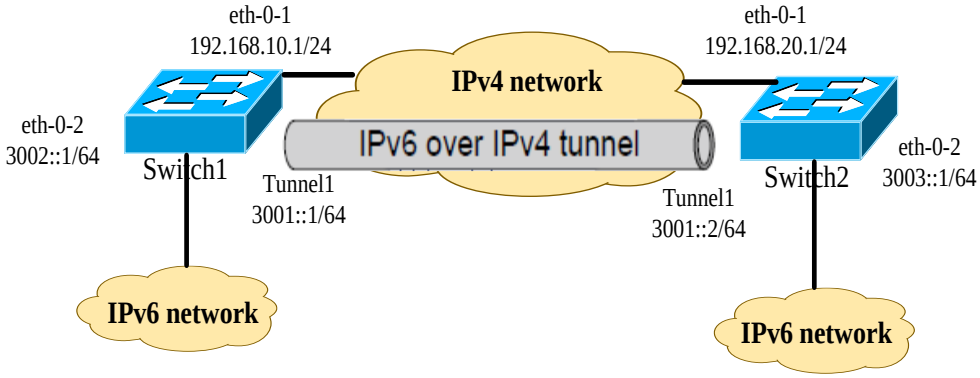


图12-4 ：配置手工隧道

如上图所示，两个 IPv6 网络分别通过 Switch1 和 Switch2 与 IPv4 网络连接，要求在 Switch1 和

Switch2 之间建立 IPv6 手动隧道，使两个 IPv6 网络可以互通。

II. 配置

Switch1

1. 使能 IPv6 功能

Switch# configure terminal	进入全局配置模式

Switch(config)# ipv6 enable	全局使能 IPv6

2. 配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口
Switch(config-if)# ip address 192.168.10.1/24	配置接口的 IPv4 地址
Switch(config)# ip route 192.168.20.0/24 192.168.10.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 192.168.10.2 0.0.2222	配置静态 ARP, 0.0.2222 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

3. 配置 eth-0-2 的 IPv6 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ipv6 address 3002::1/64	配置接口的 IPv6 地址

4. 配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel destination 192.168.20.1	配置 tunnel 的目的地
Switch(config-if)# tunnel mode ipv6ip	配置 tunnel 模式为手工隧道
Switch(config-if)# ipv6 address 3001::1/64	配置 tunnel 接口的 IPv6 地址

5. 配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

6. 配置到达对端的静态 IPv6 路由

Switch# configure terminal	进入全局配置模式

Switch(config)# ipv6 route 3003::/16 tunnel1	配置到达隧道对端的静态路由

Switch2

1. 使能 IPv6 功能

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 enable	全局使能 IPv6

2. 配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口
Switch(config-if)# ip address 192.168.20.1/24	配置接口的 IPv4 地址
Switch(config)# ip route 192.168.10.0/24 192.168.20.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 192.168.20.2 0.0.1111	配置静态 ARP, 0.0.1111 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

3. 配置 eth-0-2 的 IPv6 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ipv6 address 3003::1/64	配置接口的 IPv6 地址

4. 配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel destination 192.168.10.1	配置 tunnel 的目的地
Switch(config-if)# tunnel mode ipv6ip	配置 tunnel 模式为手工隧道
Switch(config-if)# ipv6 address 3001::2/64	配置 tunnel 接口的 IPv6 地址

5. 配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

6. 配置到达对端的静态 IPv6 路由

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 route 3002::/16 tunnel1	配置到达隧道对端的静态路由

II. 检查配置结果

Switch1

```
Switch1# show interface tunnel1
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Tunnel protocol/transport IPv6/IP, Status Valid
  Tunnel source 192.168.10.1(eth-0-1), destination 192.168.20.1
  Tunnel DSCP inherit,  Tunnel TTL 64
  Tunnel transport MTU 1480 bytes
Switch1# show ipv6 interface tunnel1
  Interface current state: UP
  The maximum transmit unit is 1480 bytes
  IPv6 is enabled, link-local address is fe80::c0a8:a01
  Global unicast address(es):
    3001::1, subnet is 3001::/64
  ICMP error messages limited to one every 1000 milliseconds
  ICMP redirects are always sent
  ND DAD is enabled, number of DAD attempts: 1
  ND router advertisement is disabled
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND retransmit interval is 1000 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements max interval: 600 secs
  ND router advertisements min interval: 198 secs
  ND router advertisements live for 1800 seconds
  ND router advertisements hop-limit is 0
  Hosts use stateless autoconfig for addresses.
```

Switch2

Switch1# show interface tunnel1

```
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
```

```
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Tunnel protocol/transport IPv6/IP, Status Valid
  Tunnel source 192.168.20.1(eth-0-1), destination 192.168.10.1
  Tunnel DSCP inherit,  Tunnel TTL 64
  Tunnel transport MTU 1480 bytes
```

Switch1# show ipv6 interface tunnel1

```
  Interface current state: UP
  The maximum transmit unit is 1480 bytes
  IPv6 is enabled, link-local address is fe80::c0a8:1401
  Global unicast address(es):
    3001::2, subnet is 3001::/64
  ICMP error messages limited to one every 1000 milliseconds
  ICMP redirects are always sent
  ND DAD is enabled, number of DAD attempts: 1
  ND router advertisement is disabled
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND retransmit interval is 1000 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements max interval: 600 secs
  ND router advertisements min interval: 198 secs
  ND router advertisements live for 1800 seconds
  ND router advertisements hop-limit is 0
  Hosts use stateless autoconfig for addresses.
```

说明

1. 在配置之前，必须先全局使能 IPv6 功能
2. 必须使 IPv4 报文 3 层路由可达，否则会造成 tunnel 报文转发失败。
3. tunnel 接口上必须配置 IPv6 地址，否则配置在该接口上的路由无效。

12.1.3 配置 6to4 隧道

I. 拓扑

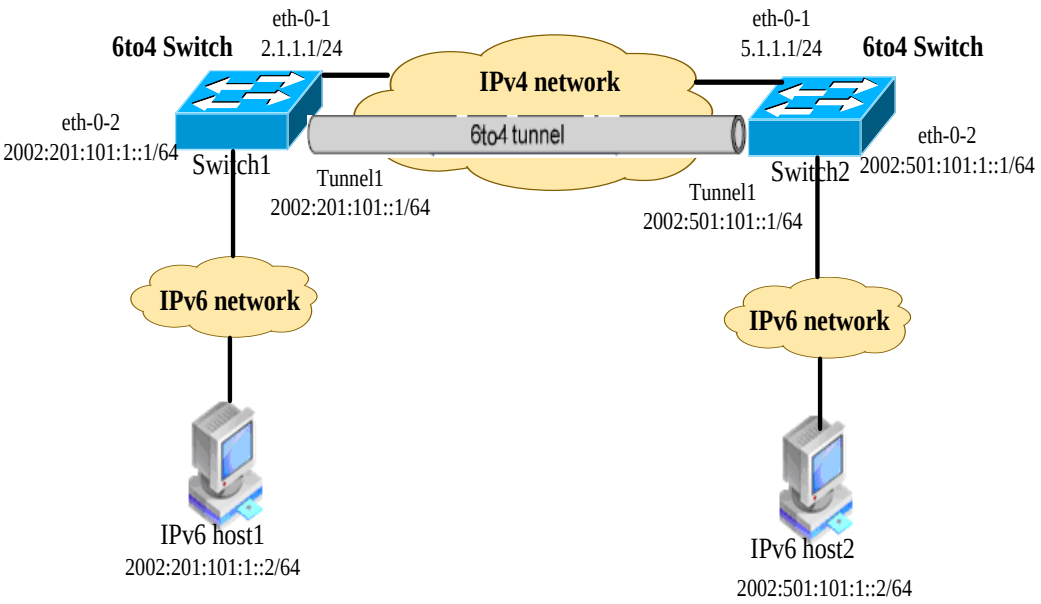


图12-5 ： 6to4 隧道配置

如上图所示，两个 6to4 网络通过网络边缘 6to4 switch（Switch1 和 Switch2）与 IPv4 网络相连。在 Switch1 和 Switch2 之间建立 6to4 隧道，实现 6to4 网络中的主机 Host1 和 Host2 之间的互通。

为了实现 6to4 网络之间的互通，除了配置 6to4 隧道外，还需要为 6to4 网络内的主机及 6to4 router 配置 6to4 地址。

- Switch1 上接口 eth-0-1 的 IPv4 地址为 2.1.1.1/24，转换成 IPv6 地址后使用 6to4 前缀 2002:0201:0101::/48。对此前缀进行子网划分，Tunnel1 使用 2002:0201:0101::/64 子网，eth-0-2 使用 2002:0201:0101:1::/64 子网。
- Switch2 上接口 eth-0-1 的 IPv4 地址为 5.1.1.1/24，转换成 IPv6 地址后使用 6to4 前缀 2002:0501:0101::/48。对此前缀进行子网划分，Tunnel1 使用 2002:0501:0101::/64 子网，eth-0-2 使用 2002:0501:0101:1::/64 子网。

II. 配置

Switch1

1) 使能 IPv6 功能

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 enable	全局使能 IPv6

2) 配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口
Switch(config-if)# ip address 2.1.1.1/24	配置接口的 IPv4 地址
Switch(config)# ip route 5.1.1.0/24 2.1.1.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 2.1.1.2 0.0.2222	配置静态 ARP, 0.0.2222 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

3) 配置 eth-0-2 的 IPv6 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ipv6 address 2002:201:101:1::1/64	配置接口的 IPv6 地址

4) 配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel mode ipv6ip 6to4	配置 tunnel 模式为 6to4 隧道
Switch(config-if)# ipv6 address 2002:201:101::1/64	配置 tunnel 接口的 IPv6 地址

5) 配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

6) 配置到达对端的静态 IPv6 路由

Switch# configure terminal	进入全局配置模式

Switch(config)# ipv6 route 2002::/16 tunnel1	配置到达隧道对端的静态路由

Switch2

1）使能 IPv6 功能

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 enable	全局使能 IPv6

2）配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口
Switch(config-if)# ip address 5.1.1.1/24	配置接口的 IPv4 地址
Switch(config)# ip route 2.1.1.0/24 5.1.1.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 5.1.1.2 0.0.1111	配置静态 ARP, 0.0.1111 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

3）配置 eth-0-2 的 IPv6 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ipv6 address 2002:501:101:1::1/64	配置接口的 IPv6 地址

4）配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel mode ipv6ip 6to4	配置 tunnel 模式为 6to4 隧道
Switch(config-if)# ipv6 address 2002:501:101::1/64	配置 tunnel 接口的 IPv6 地址

5）配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

6）配置到达对端的静态 IPv6 路由

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 route 2002::/16 tunnel1	配置到达隧道对端的静态路由

I. 检查配置结果

Switch1

Switch1# show interface tunnel1

```
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Tunnel protocol/transport IPv6/IP 6to4, Status Valid
  Tunnel source 2.1.1.1(eth-0-1), destination UNKNOWN
  Tunnel DSCP inherit,  Tunnel TTL 64
  Tunnel transport MTU 1480 bytes
```

Switch2

Switch1# show interface tunnel1

```
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Tunnel protocol/transport IPv6/IP 6to4, Status Valid
  Tunnel source 5.1.1.1(eth-0-1), destination UNKNOWN
  Tunnel DSCP inherit,  Tunnel TTL 64
  Tunnel transport MTU 1480 bytes
```

说明

1. 6to4 隧道无需配置目的地址。
2. 对于自动隧道，使用同种封装协议的 Tunnel 接口不能同时配置完全相同的源地址。
3. 如果封装前 IPv6 报文的目的 IPv6 地址与 Tunnel 接口的 IPv6 地址不在同一个网段，则必须配置通过 Tunnel 接口到达目的 IPv6 地址的转发路由，以便需要进行封装的报文能正常转发。对于自动隧道，用户只能配置静态路由，指定到达目的 IPv6 地址的路由出接口为本端 Tunnel 接口或下一跳为对端 Tunnel 接口地址，不支持动态路由。
4. 一台交换机上只允许存在一条 6to4 隧道。

12.1.4 配置 6to4 中继

I. 拓扑

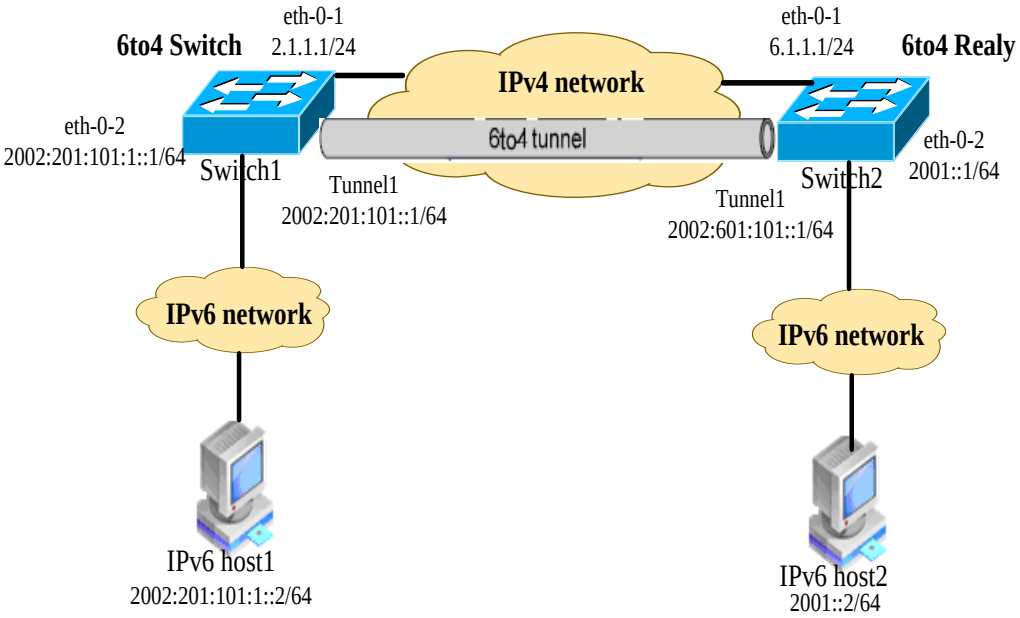


图12-6 ： 12-1 配置 6to4 中继

如上图所示，Switch1为6to4 交换机，其IPv6 侧的网络使用6to4 地址。Switch2作为6to4 中继交换机，它和IPv6 网络（2001::/16）相连。要求在Switch1和Switch2之间配置6to4 隧道，使得6to4 网络中的主机与IPv6 网络中的主机互通。

II. 配置

Switch1

1) 使能 IPv6 功能

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 enable	全局使能 IPv6

2) 配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口
Switch(config-if)# ip address 2.1.1.1/24	配置接口的 IPv4 地址

Switch(config)# ip route 6.1.1.0/24 2.1.1.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 2.1.1.2 0.0.2222	配置静态 ARP, 0.0.2222 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

3) 配置 eth-0-2 的 IPv6 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ipv6 address 2002:201:101:1::1/64	配置接口的 IPv6 地址

4) 配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel mode ipv6ip 6to4	配置 tunnel 模式为 6to4 隧道
Switch(config-if)# ipv6 address 2002:201:101::1/64	配置 tunnel 接口的 IPv6 地址

5) 配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

6) 配置到达对端的静态 IPv6 路由

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 route 2001::/16 2002:601:101::1	配置到纯 IPv6 网络的静态路由
Switch(config)# ipv6 route 2002:601:101::/48 tunnel1	配置到 6to4 中继的静态路由

Switch2

1）使能 IPv6 功能

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 enable	全局使能 IPv6

2）配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口
Switch(config-if)# ip address 6.1.1.1/24	配置接口的 IPv4 地址
Switch(config)# ip route 2.1.1.0/24 6.1.1.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 6.1.1.2 0.0.1111	配置静态 ARP, 0.0.1111 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

3）配置 eth-0-2 的 IPv6 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ipv6 address 2001::1/64	配置接口的 IPv6 地址

4）配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel mode ipv6ip 6to4	配置 tunnel 模式为 6to4 隧道
Switch(config-if)# ipv6 address 2002:601:101::1/64	配置 tunnel 接口的 IPv6 地址

5) 配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

6）配置到达对端的静态 IPv6 路由

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 route 2002::/16 tunnel1	配置到达隧道对端的静态路由

I. 检查配置结果

Switch1

Switch1# show interface tunnel1

```
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Tunnel protocol/transport IPv6/IP 6to4, Status Valid
  Tunnel source 2.1.1.1(eth-0-1), destination UNKNOWN
  Tunnel DSCP inherit,  Tunnel TTL 64
  Tunnel transport MTU 1480 bytes
```

Switch1# show ipv6 route

```
IPv6 Routing Table
Codes: C - connected, S - static, R - RIP, O - OSPF, I - IS-IS, B - BGP
       [*] - [AD/Metric]
Timers: Uptime
S        2001::/16 [1/0]
       via 2002:601:101::1 (recursive via ::, tunnel1), 00:00:32
C        2002:201:101::/64
       via ::, tunnel1, 00:00:04
C        2002:201:101::1/128
       via ::1, tunnel1, 00:00:04
S        2002:601:101::/48 [1/0]
       via ::, tunnel1, 00:00:22
```

Switch1# show ipv6 interface tunnel1

```
Interface tunnel1
  Interface current state: UP
  The maximum transmit unit is 1480 bytes
  IPv6 is enabled, link-local address is fe80::201:101
  Global unicast address(es):
    2002:201:101::1, subnet is 2002:201:101::/64
  ICMP error messages limited to one every 1000 milliseconds
  ICMP redirects are always sent
  ND DAD is enabled, number of DAD attempts: 1
  ND router advertisement is disabled
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND retransmit interval is 1000 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements max interval: 600 secs
```

```
  ND router advertisements min interval: 198 secs
  ND router advertisements live for 1800 seconds
  ND router advertisements hop-limit is 0
  Hosts use stateless autoconfig for addresses.
```

Switch2

```
Switch1# show interface tunnel1
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Tunnel protocol/transport IPv6/IP 6to4, Status Valid
  Tunnel source 6.1.1.1(eth-0-1), destination UNKNOWN
  Tunnel DSCP inherit,  Tunnel TTL 64
  Tunnel transport MTU 1480 bytes
```

说明

1. 6to4 中继交换机的配置与 6to4 交换机的配置相同，但为实现 6to4 网络与 IPv6 网络的互通，需要在 6to4 交换机上配置到 IPv6 网络的路由。
2. 当交换机上存在到达 6to4 中继的路由时，不能切换隧道模式。

12.1.5 配置 ISATAP 隧道

I. 拓扑

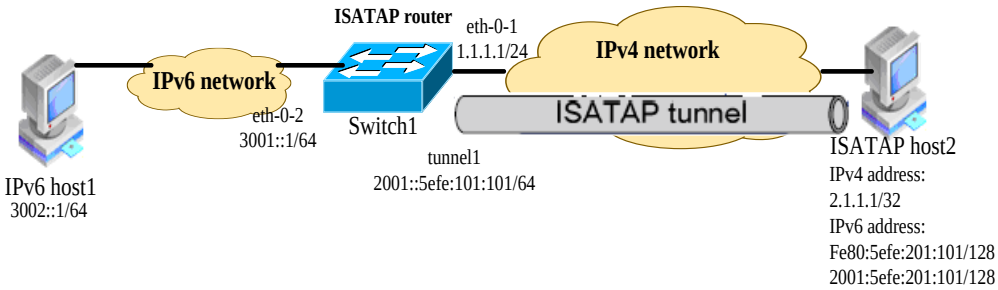


图12-7 ：配置 ISATAP 隧道

如上图所示，IPv6 网络和IPv4 网络通过ISATAP交换机相连，在IPv4 网络侧分布着一些IPv6 主机。要求将IPv4 网络中的IPv6 主机通过ISATAP隧道接入到IPv6 网络。

II. 配置

Switch1

1) 使能 IPv6 功能

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 enable	全局使能 IPv6

2) 配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口
Switch(config-if)# ip address 1.1.1.1/24	配置接口的 IPv4 地址
Switch(config)# ip route 2.1.1.0/24 1.1.1.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 1.1.1.2 0.0.2222	配置静态 ARP, 0.0.2222 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

3) 配置 eth-0-2 的 IPv6 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ipv6 address 3001::1/64	配置接口的 IPv6 地址

4) 配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel mode ipv6ip isatap	配置 tunnel 模式为 ISATAP 隧道
Switch(config-if)# ipv6 address 2001::/64 eui-64	配置 tunnel 接口的 IPv6 地址
Switch(config-if)# no ipv6 nd ra suppress	取消对 RA 消息发布的抑制，使主机可以通过交换机发布的 RA 消息获取地址前缀等信息

5) 配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

6) 配置到达对端的静态 IPv6 路由

Switch# configure terminal	进入全局配置模式
Switch(config)# ipv6 route 2001::/16 tunnel1	配置到 ISATAP 主机的静态路由

配置 ISATAP 主机

ISATAP 主机上的具体配置与主机的操作系统有关，下面仅以 Windows XP 操作系统为例进行说明。

在主机上安装 IPv6 协议。

C:\>ipv6 install

在 Windows XP 上，ISATAP 接口通常为接口 2，只要在该接口上配置 ISATAP 交换机的 IPv4 地址即可完成主机侧的配置。先看看这个 ISATAP 接口的信息：

```
Interface 2: Automatic Tunneling Pseudo-Interface
  Guid {48FCE3FC-EC30-E50E-F1A7-71172AEEE3AE}
  does not use Neighbor Discovery
  does not use Router Discovery
  routing preference 1
  EUI-64 embedded IPv4 address: 0.0.0.0
  router link-layer address: 0.0.0.0
    preferred link-local fe80::5efe:2.1.1.1, life infinite
  link MTU 1280 (true link MTU 65515)
  current hop limit 128
  reachable time 25000ms (base 30000ms)
  retransmission interval 1000ms
  DAD transmits 0
  default site prefix length 48
```

C:\>ipv6 rlu 2 1.1.1.1

只需要这么一个命令，这就完成了主机的配置，我们再来看看这个 ISATAP 接口的信息：

```
Interface 2: Automatic Tunneling Pseudo-Interface
  Guid {48FCE3FC-EC30-E50E-F1A7-71172AEEE3AE}
  does not use Neighbor Discovery
  does not use Router Discovery
  routing preference 1
  EUI-64 embedded IPv4 address: 2.1.1.1
  router link-layer address: 1.1.1.1
preferred global 2001::5efe:2.1.1.1, life 29d23h59m46s/6d23h59m46s (public)
    preferred link-local fe80::5efe:2.1.1.1, life infinite
  link MTU 1280 (true link MTU 65515)
  current hop limit 128
  reachable time 25000ms (base 30000ms)
  retransmission interval 1000ms
  DAD transmits 0
  default site prefix length 48
```

I. 检查配置结果

Switch1

Switch# show interface tunnel1

```
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Tunnel protocol/transport IPv6/IP ISATAP, Status Valid
  Tunnel source 1.1.1.1(eth-0-1), destination UNKNOWN
  Tunnel DSCP inherit,  Tunnel TTL 64
  Tunnel transport MTU 1480 bytes
```

Switch# show ipv6 interface tunnel1

```
Interface tunnel1
  Interface current state: UP
  The maximum transmit unit is 1480 bytes
  IPv6 is enabled, link-local address is fe80::101:101
  Global unicast address(es):
    2001::101:101, subnet is 2001::/64 [EUI]
  ICMP error messages limited to one every 1000 milliseconds
  ICMP redirects are always sent
  ND DAD is enabled, number of DAD attempts: 1
  ND router advertisement is enabled
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND retransmit interval is 1000 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements max interval: 600 secs
  ND router advertisements min interval: 198 secs
  ND next router advertisement due in 359 secs.
  ND router advertisements live for 1800 seconds
  ND router advertisements hop-limit is 0
  Hosts use stateless autoconfig for addresses.
```

说明

1. ISATAP 隧道无需配置目的地址。
2. 对于自动隧道，使用同种封装协议的 Tunnel 接口不能同时配置完全相同的源地址。
3. 如果封装前 IPv6 报文的目的 IPv6 地址与 Tunnel 接口的 IPv6 地址不在同一个网段，则必须配置通过 Tunnel 接口到达目的 IPv6 地址的转发路由，以便需要进行封装的报文能正常转发。对于自动隧道，用户只能配置静态路由，指定到达目的 IPv6 地址的路由出接口为本端 Tunnel 接口或下一跳为对端 Tunnel 接口地址，不支持动态路由。

12.2 NDP 配置

12.2.1 简介

网络节点（主机和路由器）使用邻居发现协议（ND）来探测直连邻居的链路层地址。并且提供一种机制，快速验证一个已经缓存在表项中的邻居的有效性。

主机还能使用 ND 来找到邻居的路由器。

网络节点之间利用该协议作为一种保活机制，定期探测邻居的有效性，探测邻居链路层地址的改变或邻居失效事件。

12.2.2 拓扑

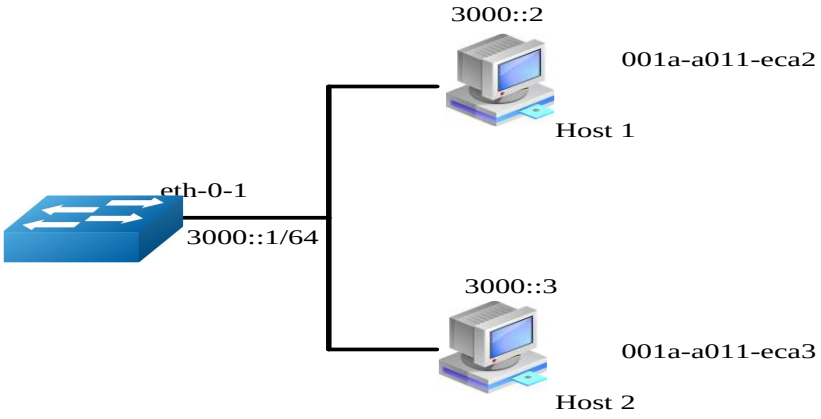


图12-8 ： NDP Topology

12.2.3 配置

在这个例子中，接口 eth-0-1 的地址是 3000::1/64。

在 3000::/64 网段中有两台主机，地址分别是 3000::2 和 3000::2. MAC 地址分别是 001a-a011-eca2 和 001a-a011-eca3。其中 3000::2 配置了静态邻居，3000::3 通过动态协议学习。

将 eth-0-1 的老化时间配置成 10 分钟。NS 报文间隔配置成 2 秒。

Switch# configure terminal	进入全局配置模式
Switch (config)# interface eth-0-1	进入接口模式
Switch (config-if)# no switchport	将接口配置为 3 层路由口

Switch (config-if)# no shutdown	打开接口
Switch (config-if)# ipv6 address 3000::1/64	配置 IPv6 地址
Switch (config-if)# ipv6 nd reachable-time 600	配置邻居老化时间
Switch (config-if)# ipv6 nd ns-interval 2000	配置 NS 报文间隔
Switch (config-if)# exit	退出接口模式
Switch (config)# ipv6 neighbor 3000::2 001a.a011.eca2	配置静态邻居表项
Switch(config)# end	退出全局配置模式。

12.2.4 命令验证

Switch # show ipv6 neighbors

```
IPv6 address                              Age       Link-Layer Addr State Interface
3000::2                                    -         001a-a011-eca2  REACH eth-0-1
3000::3                                    6         001a-a011-eca3  REACH eth-0-1
fe80::6d8:e8ff:fe4c:e700                   6         001a-a011-eca3  STALE eth-0-1
```

12.3 DHCPv6 Relay 配置

12.3.1 简介

DHCPv6 服务器和客户端都在一个子网内，则客户端和服务器之间可以直接进行 DHCPv6 协议的交互，这时不需要启动 DHCPv6 中继功能。如果 DHCPv6 服务器和客户端不在一个子网内，则需要启动 DHCPv6 中继功能将 DHCPv6 报文转发到外部的 DHCPv6 服务器。

DHCPv6 中继转发同正常的 IPv6 路由转发不同，IPv6 路由转发的 IPv6 数据包在网络之间透明交换，而 DHCPv6 中继接收 DHCPv6 消息同时产生一个新的 DHCPv6 消息发送到另一个接口。DHCPv6 中继在报文中设置中继地址，同时可以添加中继信息（Remote-id），转发到 DHCPv6 服务器端。

12.3.2 拓扑图

下图为测试 DHCPv6 中继代理功能的网络拓扑，需要两台 PC 机和一台交换机构建测试环境。

- 计算机 A 作为 DHCPv6 服务器
- 计算机 B 作为 DHCPv6 客户端
- 交换机作为 DHCPv6 中继

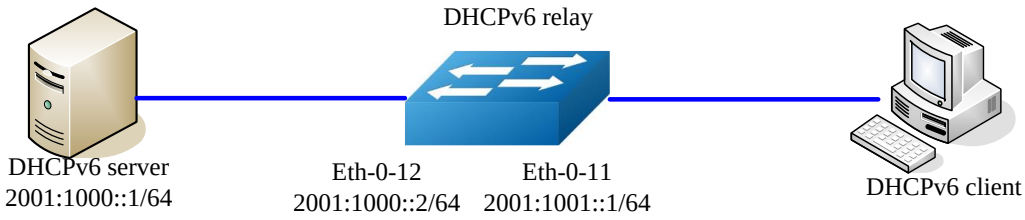


图12-9 ： DHCPv6 中继拓扑图

12.3.3 配置

使能 DHCPv6 中继全局服务

Switch(config)# service dhcpv6 enable	使能 DHCPv6 服务器
Switch(config)# dhcpv6 relay	使能 DHCPv6 Relay 功能
Switch(config)# dhcpv6 relay remote-id option	使能 DHCPv6 Remote-id 选项
Switch(config)# dhcpv6 relay pd route	使能 DHCPv6 prefix-delegation 路由学习

配置 DHCPv6 服务器组

Switch(config)# dhcpv6-server 1 2001:1000::1	创建 DHCPv6 服务器组

配置接口 eth-0-12

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-12	进入接口配置模式
Switch(config-if)# no switchport	将接口设置三层接口
Switch(config-if)# ipv6 address 2001:1000::2/64	设置 IPv6 地址

Switch(config-if)# no shutdown	使能接口
Switch(config-if)# exit	退出接口配置模式

配置接口 eth-0-11

Switch(config)# interface eth-0-11	进入接口配置模式
Switch(config-if)# no switchport	将接口设置三层接口.
Switch(config-if)# ipv6 address 2001:1001::1/64	设置 IPv6 地址
Switch(config-if)# no shutdown	使能接口
Switch(config-if)# dhcpv6-server 1	设置 DHCPv6 服务器组
Switch(config-if)# exit	退出接口配置模式

12.3.4 命令验证

步骤 1 检查接口配置。

```
Switch# show running-config interface eth-0-12
!
interface eth-0-12
 no switchport
 ipv6 address 2001:1000::1/64

!

Switch # show running-config interface eth-0-11
!
interface eth-0-11
 no switchport
 ipv6 address 2001:1001::1/64
 dhcpv6-server 1
!
```

步骤 2 检查 DHCPv6 服务器状态。

```
Switch# show services
Networking services configuration:
Service Name         Status
============================================================
dhcp                   disable
dhcpv6                 enable
```

步骤 3 检查 DHCPv6 服务器组配置。

```
Switch# show dhcpv6-server
DHCPv6 server group information:
============================================================
group 1 ipv6 address list:
[1] 2001:1000::1
```

步骤 4 显示 DHCPv6 中继统计。

```
Switch# show dhcpv6 relay statistics
DHCPv6 relay packet statistics:
============================================================
Client relayed packets :  8
Server relayed packets :  8
Client error packets   :  0
Server error packets   :  0
```

步骤 5 显示记录的 Prefix-delegation 客户端信息。

```
Switch# show dhcpv6 relay pd client
DHCPv6 prefix-delegation client information:
============================================================
Interface  : eth-0-11
Client DUID  : 000100011804ff38c2428f04970
Client IPv6 address  : fe80::beac:d8ff:fedf:c600
    IA ID  : d8dfc60
        IA Prefix :  2002:2:9:eebe::/64
            prefered/max lifetime : 280/300
            expired time :  2001-1-1 09:10:58
============================================================
```

13 IPv6 组播配置指导

13.1 IPv6 Multicast-Routing 配置

13.1.1 简介

随着 Internet 网络的不断发展，网络数据、语音、视频信息等多种交互业务与日俱增。另外，新兴的电子商务、网上会议、网上拍卖、视频点播、远程教学等对带宽和实时数据交互要求较高的服务逐渐兴起，这些服务对信息安全性、可计费性、网络带宽提出了更高的要求。

当网络中需要某信息的用户量不确定时，单播和广播方式的效率会很低，IPv6 组播技术的出现改变了这一现状。当网络中的某些用户需要特定信息时，组播信息发送者（即组播源）仅发送一次信息，借助组播路由协议为组播数据包建立树型路由，被传递的信息在距离用户端尽可能近的节点才开始复制和分发。

通过组播路由协议，多个接收者能跨越不同网络接收到组播数据。

- MLD(Multicast Listener Discovery，组播侦听发现协议)是 IPv6 协议族中负责 IPv6 组播成员管理的协议。它用来在 IPv6 主机和与其直接相邻的组播路由器之间建立、维护组播组成员关系。
- PIMv6（Protocol Independent Multicast，协议无关组播），用于 IPv6 组播路由器或多层交换机之间。为 IPv6 组播提供路由的单播路由协议可以是静态路由、RIPng、OSPFv3 等，组播路由和单播路由协议无关，只要单播路由协议能产生路由表项即可。借助 RPF（Reverse Path Forwarding，逆向路径转发）机制，PIMv6 实现了在网络中传递组播信息。为了描述上的方便，我们把由支持 PIMv6 协议的组播路由器所组成的网络称为 PIMv6 组播域，PIMv6 有两种模式：密集模式和稀疏模式，我们目前只支持稀疏模式。

13.1.2 配置

我们默认能支持限制 2048 条组播路由表。

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 multicast route-limit 1000	配置最大组播限制条目

13.1.3 检查配置

```
Switch# show ipv6 mroute 2001:1::1234

IPv6 Multicast Routing Table
Flags: I - Immediate Stat, T - Timed Stat, F - Forwarder installed
Timers: Uptime/Stat Expiry
Interface State: Interface
2001:1::1234, ff0e::1234:5678
uptime 00:00:31, stat expires 00:03:08
Owner PIM-SMv6, Flags: TF
  Incoming interface: eth-0-1
  Outgoing interface list:
    Register
    eth-0-2
2001:1::1234, ff0e::6666:6666
uptime 00:00:00, stat expires 00:03:30
Owner PIM-SMv6, Flags: TF
  Incoming interface: eth-0-1
  Outgoing interface list:
    Register
```

13.2 MLD 配置

13.2.1 简介

参与 IPv6 组播的主机、路由器、多层交换机必须具备 MLD 功能。该协议定义了查询器和主机角色：

- 网络设备的查询器发送查询消息给网络中特定组来发现组播中的成员。
- 主机发送 MLD 报告报文(响应查询报文)来通知查询者主机要加入相应的组播组列表中。
- 一个组播组的成员是动态的，主机可以随时加入和离开。在一个多播组成员的位置或数量上没有限制。

一个主机可作为不止一个组播组的成员，在同一时间，成员在组播组内活跃，它可以改变从组到组、时间到时间。一个组播组，可以持续很长一段时间，也可以非常短暂。

MLD 报文使用下面的组播地址：

- MLD 普通组查询以 ff02::1 为目的地址(在一个子网中的所有系统)。
- MLD 特定组的查询以特定组 IPv6 地址为目的查询。
- MLD 组成员发送 Report 报文给特定的组播 IPv6 地址。
- MLD 版本 1(MLDv1)离开组播组时，发送离开消息给 ff02::2。

13.2.2 参考

MLD 模块是基于以下 RFC

- RFC 2710
- RFC 3810

13.2.3 配置

MLD 的使能是依赖于组播路由协议的使能，当接口上使能 PIMv6 或者其他组播路由协议，MLD 将会在接口上自动启用，反之亦然。但是请注意，MLD 在工作之前，IPv6 组播路由必须在全局模式启用。系统支持动态学习 MLD 组记录，也可以配置静态 MLD 组记录。

启用 MLD

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 multicast-routing	全局模式下启用组播路由
Switch(config)# interface eth-0-1	进入接口 Eth-0-1
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:1::1/64	设置 IPv6 地址
Switch(config-if)# ipv6 pim sparse-mode	接口上启用 PIMv6-SM

配置 MLD 接口参数

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	接入接口模式
Switch(config-if)# ipv6 mld version 2	设置 MLD 版本
Switch(config-if)# ipv6 mld query-interval 120	设置 MLD 查询时间间隔
Switch(config-if)# ipv6 mld query-max-response-time 12	设置 MLD 查询最大响应时间
Switch(config-if)# ipv6 mld robustness-variable 3	设置 MLD 的鲁棒参数
Switch(config-if)# ipv6 mld last-member-query-count 3	设置 MLD 的最后一个成员查询计数
Switch(config-if)# ipv6 mld last-member-query-interval 2000	设置 MLD 的最后一个成员查询间隔

配置最大 MLD 组数目

可以全局配置最大 MLD 组数目或者接口模式下最大 MLD 组数目。

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 mld limit 2000	设置全局最大 MLD 组数目
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ipv6 mld limit 1000	设置接口下最大 MLD 组数目

配置静态 MLD 组

可以在接口模式下配置静态 MLD 组。

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ipv6 mld static-group ff0e::1234	配置静态 MLD 组

配置 MLD 代理

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 pim sparse-mode	在接口上启用 PIMv6-SM
Switch(config-if)# ipv6 mld proxy-service	设置接口为 MLD 代理上游口
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 pim sparse-mode	接口上启用 PIMv6-SM
Switch(config-if)# ipv6 mld mroute-proxy eth-0-1	设置 eth-0-2 为 MLD 代理下游口，MLD 代理上游口为 eth-0-1

13.2.4 检查配置

显示 MLD 接口信息

```
Switch# show ipv6 mld interface
Interface eth-0-1 (Index 1)
 MLD Active, Querier, Version 1 (default)
 Internet address is fe80::8c8e:dbff:feef:1900
 MLD interface has 0 group-record states
 MLD activity: 0 joins, 0 leaves
 MLD query interval is 125 seconds
 MLD querier timeout is 255 seconds
 MLD max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
Interface eth-0-9 (Index 9)
 MLD Active, Querier, Version 1 (default)
 Internet address is fe80::8c8e:dbff:feef:1900
 MLD interface has 0 group-record states
 MLD activity: 0 joins, 0 leaves
 MLD query interval is 125 seconds
 MLD querier timeout is 255 seconds
 MLD max query response time is 10 seconds
 Last member query response interval is 1000 milliseconds
 Group Membership interval is 260 seconds
```

显示 MLD 组信息

```
Switch# show ipv6 mld groups
MLD Connected Group Membership
Group Address                           Interface            Expires
ff0e::1234:5678                         eth-0-2              00:03:01
```

13.3 PIMv6-SM 配置

13.3.1 简介

协议无关组播稀疏模式(PIMv6-SM)是一个组播路由协议，用来将稀疏分散的组播设备联系起来协同工作。它将有助于分散的网络节点节约带宽和通过发送单一流量到多个接受着来降低网络流量。

PIMv6-SM 使用接收者发起成员的 IPv6 组播模型，支持共享和最短路径树，并使用软状态机制，以适应不断变化的网络条件。它依赖于单播路由协议来建立和维护路由器间的组播路由。

13.3.2 参考

在 PIMv6-SM 模块是基于以下的 IETF 标准：

RFC 4601

13.3.3 术语

以下是 PIMv6-SM 协议概念的简要描述：

汇聚点(RP)

RP（Rendezvous Point）在 SM 模式中作为组播的汇聚点，发送者和接收者在 RP 处进行汇聚。对于所有的组播路由器，必须知道某个组播组对应哪个 RP。

所有的组播数据需要在 RP 上注册，然后所有需要组播数据的接收者通过向 RP 发送 JOIN 报文来请求数据。源的注册机制就是让 RP 知道现在网络内有什么源的数据。

组播路由信息库(MRIB)

组播路由表是从单播路由表的获得的。在 PIMv6-SM 中，MRIB 是用来决定向何处发送加入/剪枝消息。它还提供了目的网络的路由度量。发送和处理的 Assert 消息时将使用这些度量。

反向路径转发（RPF）

反向路径转发是指路由器在接受数据包从源 A 通过接口 IF1 时，只有 IF1 是到达源 A 的出接口时才会接受这个包。反向路径转发通过使用单播路由表来决定入端口是否正确。这个数据包将被转发是由于单播路由表表明了接口 IF1 是到达源 A 的最短路径。单播路由表为组播数据选择最短路径。

组播树状态信息库(TIB)

组播树状态信息库是组播路由器上保存所有组播转发树信息的一个信息库，通过收到 PIMv6 加入/剪枝消息，Assert 消息和 MLD 消息建立起来。

上游 Upstream

朝向树根，树根可能是源或 RP。

下游 Downstream

远离树根，树根可能是源或 RP。

基于源的树

基于源的树的转发路径是到达源的最短转发路径，如果单播路由度量是跳数，基于源的树的转发路径的跳数最小，如果单播路由度量是延迟，基于源的树的转发路径的延迟最小。

对于每个组播源，有一个对应的组播转发树直接将源和接收者连接起来。所有发往指定组的流量沿着对应的转发树进行转发。

共享树

共享树依赖于汇聚点(RP)，所有流量从源发往那个汇聚点，然后汇聚点再将流量发送给接收者。对于每一个组播组来说，不管有多少个源，只有一个转发树。共享树是单向的，流量只会从 RP 流向接收者。如果一个源要发送组播数据，首先组播数据要被发到 RP，然后在从 RP 发送到接收者。

自举路由器(BSR)

当一个组播源开始发送组播数据或者一个接收者开始发送加入信息到 RP，组播路由器必须知道汇聚点的信息。自举路由器负责在 PIMv6-SM 网络启动后，收集网络内的 RP 信息，为每个组选举出 RP，然后将 RP 集（即组-RP 映射数据库）发布到整个 PIMv6-SM 网络。

数据流从源到接收者

发送 Hello 消息

PIMv6 路由器定期的发送 Hello 消息来发现 PIMv6 路由器邻居。Hello 消息是组播报文，使用 ff02::d 这个地址。PIMv6 路由器对 Hello 消息进行响应，Hello 消息中的 Hold 时间来决定信息的有效时间。

选举指定路由器

在一个多路访问的网络中如果有多个组播路由器，只能有一个组播路由器被选为指定路由器，负责为本地网络的组播接收者往 RP 发送加入/剪枝消息。

RP 发现

PIMv6-SM 通过自举路由器来产生自举消息，然后发布 RP 信息给所有的组播路由器。组播路由器接收和保存自举消息，当 DR 从直连 host 收到一个 MLD 报文或组播数据，DR 计算出该组播组的 RP，然后发送加入/剪枝到 RP 或者封装 reigster 报文到 RP。在小网络环境下可以静态指定 RP。

加入共享树

要加入一个多播组，主机发送一个 MLD 消息给上游路由器，组播路由器向 RP 方向的上游的 PIMv6 邻居发送加入报文。当组播路由器接收到下游设备的加入请求后，检查本地的组播组是否存在。如果存在，说明加入消息被送到共享树，收到消息的接口被成为 outgoing 的接口。如果不存在，条目将被创建，收到消息接口的被加入到 outgoing 中并再次向 RP 方向的上游的 PIMv6 邻居发送加入报文。

组播源注册

与组播源 S 直接相连的路由器接收到该组播报文后，就将该报文封装成 Register 注册报文，并单播发送给对应的 RP。当 RP 接收到来自组播源 S 的注册消息后，一方面解封装注册消息并将组播信息沿着 RPT 树转发到接收者，另一方面朝组播源 S 逐跳发送（S，G）加入消息，从而让 RP 和组播源 S 之间的所有路由器上都生成了（S，G）表项，这些沿途经过的路由器就形成了 SPT 树的一个分支。SPT 源树以组播源 S 为根，以 RP 为目的地组播源 S 发出的组播信息沿着已经建立好的 SPT 树到达 RP，然后由 RP 将信息沿着 RPT 共享树进行转发。

发送注册停止消息

当 RP 从组播源接收到注册报文后也收到未封装的组播报文，将发送注册停止消息给组播源一侧的 DR，当 DR 收到注册停止消息后将不再发送注册消息给 RP 了。

剪枝端口

接收者侧的组播路由器向 RP 方向的上游的 PIMv6 邻居发送剪枝报文，当上联组播路由器收到剪枝报文后，将收到剪枝报文的端口从转发端口中删除，当本路由器上没有其他接收者后会继续向 RP 方向的上游的 PIMv6 邻居发送剪枝报文。

转发组播数据

PIMv6-SM 路由器将组播数据发往那些已经明确表示加入组播组的接收者。

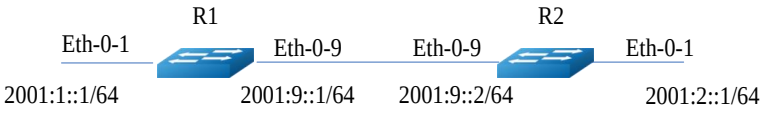
组播路由器将进行 RPF 检查，只有检查通过的组播数据包才将通过出端口发送出去。

13.3.4 配置通用 PIMv6 Sparse-mode

I. 配置

PIMv6-SM 是一个软状态协议。主要要求是，所需的接口上启用 PIMv6-SM 协议，并正确配置的 RP 信息，通过静态或动态的方法。所有组播组的 MLD 报告/离开和 PIMv6 加入/剪枝消息保持动态。目前，我们只支持一个 RP 的所有组播组（224.0.0.0 / 4）。

本节提供了两个相关的场景，PIMv6-SM 配置的例子。下面的例子中使用的网络拓扑如下；



配置静态 RP

以上例子中 R1 是 RP，所有的路由器都配置静态 RP：

- 每个路由器配置静态 RP 地址 2001:1::1。
- 所有接口上必须启用 PIMv6-SM 功能。

R1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:1::1/64	配置 IPv6 地址

Switch(config-if)# ipv6 pim sparse-mode	在接口上启用 PIMv6-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:9::1/64	配置 IPv6 地址
Switch(config-if)# ipv6 pim sparse-mode	在接口上启用 PIMv6-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# ipv6 route 2001:2::/64 2001:9::2	配置静态单播路由
Switch(config)# ipv6 pim rp-address 2001:1::1	配置静态 RP 地址

R2

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:2::1/64	配置 IPv6 地址
Switch(config-if)# ipv6 pim sparse-mode	在接口上启用 PIMv6-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:9::2/64	配置 IPv6 地址
Switch(config-if)# ipv6 pim sparse-mode	在接口上启用 PIMv6-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# ipv6 route 2001:1::/64 2001:9::1	配置静态单播路由

Switch(config)# ipv6 pim rp-address 2001:1::1	配置静态 RP 地址

I. 检查配置

所有的路由器配置使用相同的 RP 地址 2001:1::1，使用以下命令来验证 RP 的配置，接口的详细信息和组播路由表。

RP 详细说明

在 R1 上，显示 PIMv6 稀疏模式 RP 映射的命令表明 11.1.1.1 是对所有组播组 224.0.0.0 / 4 静态配置的 RP。所有其他路由器都会有类似的输出：

R1# show ipv6 pim sparse-mode rp mapping

```
PIM Group-to-RP Mappings
Group(s): ff00::/8, Static
    RP: 2001:1::1
         Uptime: 00:00:04
Embedded RP Groups:
```

接口的详细信息

显示 R1 接口的组播信息。

R1# show ipv6 pim sparse-mode interface

```
Interface    VIFindex Ver/   Nbr    DR
                      Mode   Count  Prior
eth-0-1      2        v2/S   0      1
    Address       : fe80::fc94:efff:fe96:2600
    Global Address: 2001:1::1
    DR            : this system
eth-0-9      0        v2/S   0      1
    Address       : fe80::fc94:efff:fe96:2600
    Global Address: 2001:9::1
    DR            : this system
```

IPv6 组播路由表

显示 PIMv6-SM 的组播路由表。

R1# show ipv6 pim sparse-mode mroute detail

```
IPv6 Multicast Routing Table
(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
FCR Entries: 0
*, ff0e::1234:5678
Type: (*,G)
Uptime: 00:01:37
```

```
  RP: 2001:1::1, RPF nbr: None, RPF idx: None
  Upstream:
   State: JOINED, SPT Switch: Enabled, JT: off
   Macro state: Join Desired,
  Downstream:
   eth-0-1:
     State: NO INFO, ET: off, PPT: off
     Assert State: NO INFO, AT: off
      Winner: ::, Metric: 4294967295, Pref: 4294967295, RPT bit: on
     Macro state: Could Assert, Assert Track
  Local Olist:
   eth-0-1
```

R2# show ipv6 pim sparse-mode mroute detail

```
IPv6 Multicast Routing Table
(*,*,RP) Entries: 0
(*,G) Entries: 1
(S,G) Entries: 0
(S,G,rpt) Entries: 0
FCR Entries: 0
*, ff0e::1234:5678
Type: (*,G)
Uptime: 00:00:06
  RP: 2001:1::1, RPF nbr: None, RPF idx: None
  Upstream:
   State: JOINED, SPT Switch: Enabled, JT: off
   Macro state: Join Desired,
  Downstream:
   eth-0-1:
     State: NO INFO, ET: off, PPT: off
     Assert State: NO INFO, AT: off
      Winner: ::, Metric: 4294967295, Pref: 4294967295, RPT bit: on
     Macro state: Could Assert, Assert Track
  Local Olist:
   eth-0-1
```

13.3.5 配置动态 RP

在小型并且简单的网络中，组播信息量少，全网络仅依靠一个 RP 进行信息转发即可，此时可以在 SM 域中各路由器上静态指定 RP 位置。但是更多的情况下，PIMv6-SM 网络规模都很大，通过 RP 转发的组播信息量巨大，为了缓解 RP 的负担同时优化共享树的拓扑结构，不同组播组应该对应不同的 RP，此时就需要自举机制来动态选举 RP。

I. 配置

以下是动态 RP 的详细配置：

R1

Switch# configure terminal	进入配置模式

Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:1::1/64	配置 IPv6 地址
Switch(config-if)# ipv6 pim sparse-mode	在接口上启用 PIMv6-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:9::1/64	配置 IPv6 地址
Switch(config-if)# ipv6 pim sparse-mode	进入配置模式
Switch(config-if)# exit	进入接口模式
Switch(config)# ipv6 route 2001:1::/64 2001:9::1	配置静态单播路由
Switch(config)# ipv6 pim rp-candidate eth-0-1	配置候选 RP 接口

R2

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:2::1/64	配置 IPv6 地址
Switch(config-if)# ipv6 pim sparse-mode	在接口上启用 PIMv6-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开接口
Switch(config-if)# no switchport	设置接口为三层接口

Switch(config-if)# ipv6 address 2001:9::2/64	配置 IPv6 地址
Switch(config-if)# ipv6 pim sparse-mode	在接口上启用 PIMv6-SM
Switch(config-if)# exit	退出接口模式
Switch(config)# ipv6 route 2001:1::/64 2001:9::1	配置静态单播路由
Switch(config)# ipv6 pim rp-candidate eth-0-9	配置候选 RP 接口
Switch(config)# ipv6 pim bsr-candidate eth-0-9	配置候选 BSR 接口

选择最高优先级的路由器为 RP。如果有两个或多个路由器的优先级相同，在 BSR 机制的一个哈希函数是用来选择的 RP，以确保在 PIMv6 域的所有路由器对同一组相同的 RP。使用 **ipv6 pim rp-candidate IFNAME PRIORITY** 命令来改变候选 RP 的默认的优先级。

I. 检查配置

PIMv6-SM 的组-RP 的 Mapping 关系

使用 **show ipv6 pim sparse-mode rp mapping** 命令，来显示组-RP 的映射的详细信息，输出内容是候选 RP 信息。对组的范围 ff00::/8 的组有两个候选 RP。候选 RP 2001:1::1 默认的优先级 192，而候选 RP 2001:9::2 的优先级被配置为 2。由于候选 RP 2001:1::1 由于具有更高的优先权，它被选中作为组播组 ff00::/8 的 RP。

R2# show ipv6 pim sparse-mode rp mapping

```
PIM Group-to-RP Mappings
This system is the Bootstrap Router (v2)
Group(s): ff00::/8
  RP: 2001:9::2
    Info source: 2001:9::2, via bootstrap, priority 2
         Uptime: 00:00:32, expires: 00:02:02
  RP: 2001:1::1
    Info source: 2001:1::1, via bootstrap, priority 192
         Uptime: 00:00:31, expires: 00:02:03
Embedded RP Groups:
```

RP 详细显示

要显示特定组的 RP 路由器的信息，使用下面的命令。此输出显示，2001:9::2 已经选择 ff02::1234 的组播组的 RP。

R2# show ipv6 pim sparse-mode rp-hash ff02::1234

```
    RP: 2001:9::2
    Info source: 2001:9::2, via bootstrap
```

RP 信息后达到域中的所有 PIMv6 路由器，各种状态机保持所有路由从组成员的加入/剪枝的结果。要显示接口的详细信息和组播路由表的信息，请参见以上配置 RP 的静态部分。

13.3.6 配置自举路由器

每个组播组需要有一个为它服务的 RP，这个 RP 作为基于组播组的分发树的根。为了组播数据能从发送者到达接收者，在一个组播域内的组播路由器需要使用同样的组播组-RP 的映射。为了选择指定组播组的 RP，组播路由器需要维护一系列的组播组-RP 的映射关系，这被称为 RP 集。自举路由器的机制就是用来让在同一个组播域内的组播路由器能够学习到这个 RP 集。

BSR 是 PIMv6-SM 网络里的管理核心，主要负责：

- 负责收集网络中 Candidate-RP（C-RP）发来的 Advertisement 宣告信息。
- 为每个组播组选择部分 C-RP 信息以组成 RP-Set 集（即组播组和 RP 的映射数据库）。
- 发布到整个 PIMv6-SM 网络，从而使网络内的所有路由器（包括 DR）都会知道 RP 的位置。

在一个 PIMv6 域中，需要配置一个或多个候选 BSR，候选 BSR 之间通过自动选举，产生自举路由器 BSR，负责收集并发布 RP 信息。下面简单描述一下候选 BSR 之间的自动选举：

- 在将路由器配置为候选 BSR 时，必须同时指定一个启动了 PIMv6-SM 的接口。
- 每个候选 BSR 开始都认为自己是本 PIMv6-SM 的 BSR，并使用这个接口的 IPv6 地址作为 BSR 地址，发送自举报文（Bootstrap message）。
- 当候选 BSR 收到其它路由器发来的自举报文时，它将新收到的自举报文的 BSR 地址与自己的 BSR 地址进行比较，比较标准包括优先级和 IPv6 地址，优先级相同的情况下，较大的 IPv6 地址被认为是更好的。如果前者更好，则将这个新的 BSR 地址替换自己的 BSR 地址，并且不再认为自己是 BSR。否则，保留自己的 BSR 地址，继续将自己视为 BSR。
- 备选 RP 将自己的 RP 信息报告给自举路由器，然后自举路由器将汇聚的 RP 集通过自举报文发布到整个组播域的所有路由器。

I. 拓扑

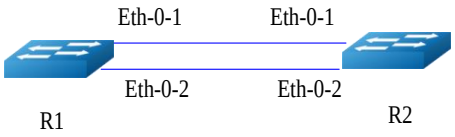


图13-1 BSR 拓扑

II. 配置

Router 1

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 pim bsr-candidate eth-0-1	指定 BSR 的候选接口，默认优先级 64

Router 2

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 pim bsr-candidate eth-0-1 10 25	配置 HASH 掩码长度为 10 优先级 25 的 BSR 候选接口
Switch(config)# ipv6 pim rp-candidate eth-0-1 priority 0	配置优先级为 0 的 RP 候选接口

通过命令 **ipv6 pim unicast-bsm** 配置接口以单播方式发送和接收 BSM 消息。

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ipv6 pim dr-priority 10	配置接口 DR 的优先级
Switch(config-if)# ipv6 pim unicast-bsm	配置接口以单播方式发送和接收 BSM 消息

I. 检查配置

检查候选 BSR 路由器

```
Switch# show ipv6 pim sparse-mode bsr-router

PIM6v2 Bootstrap information
This system is the Bootstrap Router (BSR)
  BSR address: 2001:9::1 (?)
  Uptime:      00:01:27, BSR Priority: 64, Hash mask length: 126
  Next bootstrap message in 00:00:16
  Role: Candidate BSR
  State: Elected BSR
```

检查候选 BSR 路由器

```
Switch# show ipv6 pim sparse-mode bsr-router
PIM6v2 Bootstrap information
  BSR address: 2001:9::1 (?)
  Uptime:      00:01:34, BSR Priority: 64, Hash mask length: 126
  Expires:     00:01:51
  Role: Candidate BSR
  State: Candidate BSR
  Candidate RP: 2001:9::2(eth-0-9)
    Advertisement interval 60 seconds
    Next C-RP advertisement in 00:00:35
```

在 E-BSR 上检查 RP

```
Switch# show ipv6 pim sparse-mode rp mapping
PIM Group-to-RP Mappings
This system is the Bootstrap Router (v2)
Group(s): ff00::/8
  RP: 2001:9::2
    Info source: 2001:9::2, via bootstrap, priority 0
         Uptime: 00:45:37, expires: 00:02:29
Embedded RP Groups:
```

在 C-BSR 上检查 RP

```
Switch# show ipv6 pim sparse-mode rp mapping

PIM Group-to-RP Mappings
Group(s): ff00::/8
  RP: 2001:9::2
    Info source: 2001:9::1, via bootstrap, priority 0
         Uptime: 00:03:14, expires: 00:01:51
Embedded RP Groups:
```

13.3.7 配置 PIMv6-SSM

PIMv6-SSM 可以跟 PIMv6-SM 在组播路由器上一起工作。PIMv6-SSM 默认是 disable 的。

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 pim ssm default	使能 PIMv6-SSM
Switch(config)# ipv6 pim ssm range ipv6acl	根据指定的 ipv6acl 来设置 PIMv6-SSM 的组范围

13.4 PIMv6-DM 配置

13.4.1 简介

协议无关组播密集模式(PIMv6-DM)是一个组播路由协议，用来将密集分布的组播设备联系起来协同工作。它将有助于分散的网络节点节约带宽和通过发送单一流量到多个接收者来降低网络流量。

PIMv6-DM 设想当一个组播源开始发送组播流的时候，所有的下游系统都期望接受这个组播流。刚开始组播流被泛洪到整个网络。当泛洪的时候，PIMv6-DM 使用 RPF 来防止组播流的环路。如果某些网络区域没有该组播组的接收成员，PIMv6-DM 会把转发分支通过剪枝来删除掉。

剪枝状态有一个生命周期，当生命周期超时后，组播数据将再一次开始转发，每个(S,G)对应的组播组都有自己的剪枝状态。当某个组播组有新的接收者出现在已经被剪枝的区域里，路由器会通过朝组播源发送”graft”消息来把剪枝状态转换成转发路径。

13.4.2 参考

在 PIMv6-DM 模块是基于以下的 IETF 标准：

RFC 3973

13.4.3 配置通用 PIMv6 dense-mode

I. 拓扑

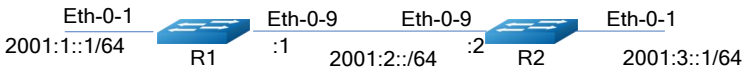


图13-2 配置 PIMv6 dense-mode

II. 配置

PIMv6-DM 是一个软状态协议。主要要求是在所需的接口上启用 PIMv6-DM 协议。所有组播组的状态通过 MLD 报告/离开和 PIMv6 消息来动态的维护。

本节提供了一个 PIMv6-DM 配置的相关的场景。下面的例子中使用的网络拓扑如上图；

组播流从 R1 的 eth-0-1 口进来，接收者来与 R2 的 eth-0-1 相连。

下面是配置的举例：

Configuring R1

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 enable	使能 ipv6
Switch(config)# interface eth-0-1	进入 eth-0-1 的接口模式
Switch(config-if)# no shutdown	启用端口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:1::1/64	配置接口的 ipv6 地址
Switch(config-if)# ipv6 pim dense-mode	使能接口的 pimv6 dm 功能
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入 eth-0-9 的接口模式
Switch(config-if)# no shutdown	启用端口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:2::1/64	配置接口的 ipv6 地址
Switch(config-if)# ipv6 pim dense-mode	使能接口的 pimv6 dm 功能
Switch(config-if)# exit	退出接口模式

Switch(config)# ipv6 route 2001:3::/64 2001:2::2	配置一条静态路由

Configuring R2

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 enable	使能 ipv6
Switch(config)# interface eth-0-1	进入 eth-0-1 的接口模式
Switch(config-if)# no shutdown	启用端口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:3::1/64	配置接口的 ipv6 地址
Switch(config-if)# ipv6 pim dense-mode	使能接口的 pimv6 dm 功能
Switch(config-if)# exit	退出接口模式
Switch(config)# interface eth-0-9	进入 eth-0-9 的接口模式
Switch(config-if)# no shutdown	启用端口
Switch(config-if)# no switchport	设置接口为三层接口
Switch(config-if)# ipv6 address 2001:2::2/64	配置接口的 ipv6 地址
Switch(config-if)# ipv6 pim dense-mode	使能接口的 pimv6 dm 功能
Switch(config-if)# exit	退出接口模式
Switch(config)# ipv6 route 2001:1::/64 2001:2::1	配置一条静态路由

I. 检查配置

使用下面的命令来检查接口配置和路由表信息。

接口的详细信息

用 show ipv6 pim dense-mode interface 来显示 R1 上接口的详细信息.

```
R1# show ipv6 pim dense-mode interface
Neighbor Address                          Interface   VIFIndex Ver/   Nbr
                                                               Mode   Count
fe80::326f:c9ff:fef2:8200                 eth-0-1     0        v2/D   0
fe80::326f:c9ff:fef2:8200                 eth-0-9     2        v2/D   1
```

邻居的详细信息

用 show ipv6 pim dense-mode neighbor 来显示 R1 上邻居的详细信息

```
R1# show ipv6 pim sparse-mode neighbor
Neighbor Address                          Interface   Uptime/Expires    Ver
fe80::ce47:6eff:feb7:1400                 eth-0-9     00:51:51/00:01:24 v2
```

组播路由表的信息

用 show ipv6 pim dense-mode mroute 来显示 PIM-DM 组播路由表的信息

R1# show ipv6 pim dense-mode mroute

```
PIM-DM Multicast Routing Table
(2001:1::2, ff0e::1)
  Source directly connected on eth-0-1
  State-Refresh Originator State: Originator
  Upstream IF: eth-0-1
    Upstream State: Forwarding
    Assert State: NoInfo
  Downstream IF List:
    eth-0-9, in 'olist':
      Downstream State: NoInfo
      Assert State: NoInfo
```

R2# show ipv6 pim dense-mode mroute

```
PIM-DM Multicast Routing Table
(2001:1::2, ff0e::1)
  RPF Neighbor: none
  Upstream IF: eth-0-9
    Upstream State: AckPending
    Assert State: Loser
  Downstream IF List:
    eth-0-1, in 'olist':
      Downstream State: NoInfo
      Assert State: NoInfo
```

13.5 配置 MLD Snooping

13.5.1 简介

MLD Snooping（Multicast Listener Discovery，MLD 侦听）是运行在二层以太网交换机上的 IPv6 组播约束机制，用于管理和控制 IPv6 组播组。

二层交换机通过 MLD Snooping 来控制 IPv6 组播流量的泛洪。当二层以太网交换收到主机和路由器之间传递的 MLD 报文时，MLD Snooping 将对 MLD 报文所带的信息进行分析，将端口和 MAC 组播地址建立起映射关系，并根据这样的映射关系转发 IPv6 组播数据。IPv6 组播路由器定期发送通用组查询来维护 IPv6 组播组成员关系。所有接

收者将发送 MLD 报告报文来响应这个查询，交换机通过这个监听 MLD 报告报文来建立转发表项。

二层的组播组可以通过 MLD 报文动态建立，也可以静态配置。静态配置的组播组将覆盖动态学的组播组。

13.5.2 配置启用 MLD Snooping

MLD Snooping 可以在全局模式下启用或者每个 VLAN 下启用。假如 MLD Snooping 在全局模式下关闭，即使你在每个 VLAN 下启用 MLD Snooping 也是无效的。假如 MLD Snooping 在全局模式下开启。可以在某个 VLAN 下关闭 MLD Snooping，另一方面，全局配置可以覆盖每个 VLAN 配置。默认情况下，MLDSnooping 在全局模式下和每个 VLAN 上没有使能。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)# ipv6 mld snooping	全局模式下启用 MLD Snooping
Switch(config)#ipv6 mld snooping vlan 1	在单 VLAN 模式下启用 MLD Snooping
Switch # show ipv6 mld snooping vlan 1	检查配置

II. 命令验证

Switch # show ipv6 mld snooping vlan 1

```
Global Mld Snooping Configuration
------------------------------------------------
Mld Snooping                            :Enabled
Mld Snooping Fast-Leave                 :Disabled
Mld Snooping Version                    :1
Mld Snooping Max-Member-Number          :4096
Mld Snooping Unknown Multicast Behavior  :Flood
Mld Snooping Report-Suppression         :Enabled
Vlan 1
-----------
Mld Snooping                            :Enabled
Mld Snooping Fast-Leave                 :Disabled
Mld Snooping Report-Suppression         :Enabled
Mld Snooping Version                    :1
Mld Snooping Max-Member-Number          :4096
Mld Snooping Unknown Multicast Behavior  :Flood
Mld Snooping Group Access-list          :N/A
Mld Snooping Mrouter Port               :
Mld Snooping Mrouter Port Aging Interval(sec) :255
```

13.5.3 配置 MLD Snooping 快速离开

正常情况下，MLD Snooping 在接收到 MLD 离开报文后不会直接将端口从组播组中删除，而是发送 MLD 特定组查询报文，如果等待一段时间后没有得到响应，才将该端口从组播组中删除。启动快速删除功能后，MLD Snooping 收到 MLD 离开报文时，直接将端口从组播组中删除。当端口下只有一个用户时，快速删除可以节省带宽。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)#ipv6 mld snooping fast-leave	全局模式下启用快速离开功能
Switch(config)#ipv6 mld snooping vlan 1 fast-leave	在 VLAN 模式下启用快速离开功能
Switch# show ipv6 mld snooping vlan 1	检查配置

II. 命令验证

```
Switch # show ipv6 mld snooping vlan 1
Global Mld Snooping Configuration
------------------------------------------------
Mld Snooping                              :Enabled
Mld Snooping Fast-Leave                   :Enabled
Mld Snooping Version                      :1
Mld Snooping Max-Member-Number            :4096
Mld Snooping Unknown Multicast Behavior   :Flood
Mld Snooping Report-Suppression           :Enabled
Vlan 1
-----------
Mld Snooping                              :Enabled
Mld Snooping Fast-Leave                   :Enabled
Mld Snooping Report-Suppression           :Enabled
Mld Snooping Version                      :1
Mld Snooping Max-Member-Number            :4096
Mld Snooping Unknown Multicast Behavior   :Flood
Mld Snooping Group Access-list            :N/A
Mld Snooping Mrouter Port                 :
Mld Snooping Mrouter Port Aging Interval(sec) :255
```

13.5.4 配置 MLD Snooping 查询参数

三层交换机在所连接的网段上周期性的发送 MLD 通用查询报文，通过解析返回的 MLD 主机报告报文，获知该网段内哪些组播组有成员。组播路由器周期性地发送查询报文，当得到某一组成员的 MLD 主机报告报文的时候，刷新该网段相应的组成员关系信息。

I. 配置

Switch #configure terminal	进入配置模式

Switch(config)# ipv6 mld snooping query-interval 100	设置查询时间间隔是 100 秒
Switch(config)# ipv6 mld snooping query-max-response-time 5	设置查询的最大响应时间 5 秒
Switch(config)#ipv6 mld snooping last-member-query-interval 2000	设置当仅存最后一个成员时的查询间隔
Switch(config)#ipv6 mld snooping vlan 1 querier address fe80::1	在 VLAN1 上配置 MLD Snooping 的查询地址
Switch(config)#ipv6 mld snooping vlan 1 querier	在 VLAN1 上启用 MLD Snooping 的查询功能
Switch(config)#ipv6 mld snooping vlan 1 query-interval 200	在 VLAN1 上设置查询时间间隔是 200 秒
Switch(config)#ipv6 mld snooping vlan 1 query-max-response-time 5	在 VLAN1 上设置查询的最大响应时间 5 秒
Switch(config)#ipv6 mld snooping vlan 1 querier-timeout 100	在 VLAN1 上设置查询超时时间 100 秒
Switch(config)#ipv6 mld snooping vlan 1 last-member-query-interval 2000	在 VLAN1 上设置特定组的查询间隔 2000 秒
Switch(config)# ipv6 mld snooping vlan 1 discard-unknown	在 VLAN1 上丢弃未知组播报文
Switch(config)# ipv6 mld snooping discard-unknown	在全局模式下设置丢弃未知组播报文

II. 命令验证

```
Switch # show ipv6 mld snooping querier
Global Mld Snooping Querier Configuration
-------------------------------------------------
Version                          :1
Last-Member-Query-Interval (msec) :2000
Max-Query-Response-Time (sec)     :5
Query-Interval (sec)              :100
Global Source-Address             :::
TCN Query Count                   :2
TCN Query Interval (sec)          :10
Vlan 1:   MLD snooping querier status
-----------------------------------------
Elected querier is : fe80::1
-----------------------------------------
Admin state                       :Enabled
Admin version                     :1
```

```
Operational state                 :Querier
Querier operational address       :fe80::1
Querier configure address         :fe80::1
Last-Member-Query-Interval (msec) :2000
Max-Query-Response-Time (sec)     :5
Query-Interval (sec)              :200
Querier-Timeout (sec)             :100
```

13.5.5 配置 MLD Snooping 组播路由端口

组播路由端口是交换机上连接到组播路由器的端口，可以动态学习或者静态配置。当某个 VLAN 的端口上收到 MLD 通用组查询报文或者是 PIMv6 Hello 报文，该端口成为这个 VLAN 的组播路由端口。所有从组播路由端口上收到的的 MLD 查询报文要在所属 VLAN 内广播。所有 VLAN 上收到 MLD 报告/离开报文也将从组播路由端口转发(报文抑制关闭的情况下)，另外所有从该 VLAN 上收到的组播流量将从组播路由端口转发。

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 mld snooping report-suppression	启用 MLD Snooping 的报告抑制功能
Switch(config)# ipv6 mld snooping vlan 1 mrouter interface eth-0-1	配置静态组播路由端口
Switch(config)# ipv6 mld snooping vlan 1 report-suppression	在 VLAN1 上启用报告抑制功能
Switch(config)# ipv6 mld snooping vlan 1 mrouter-aging-interval 200	配置动态组播路由端口老化时间

II. 命令验证

```
Switch# show ipv6 mld snooping vlan 1
Global Mld Snooping Configuration
------------------------------------------------
Mld Snooping                              :Enabled
Mld Snooping Fast-Leave                   :Enabled
Mld Snooping Version                      :1
Mld Snooping Max-Member-Number            :4096
Mld Snooping Unknown Multicast Behavior   :Discard
Mld Snooping Report-Suppression           :Enabled
Vlan 1
-----------
Mld Snooping                              :Enabled
Mld Snooping Fast-Leave                   :Enabled
Mld Snooping Report-Suppression           :Enabled
Mld Snooping Version                      :1
Mld Snooping Max-Member-Number            :4096
```

```
Mld Snooping Unknown Multicast Behavior  :Discard
Mld Snooping Group Access-list           :N/A
Mld Snooping Mrouter Port                :eth-0-1(static)
Mld Snooping Mrouter Port Aging Interval(sec) :200
```

13.5.6 配置 MLD Snooping 查询 TCN

可以通过配置 TCN 的时间间隔以及查询次数来适应 STP 收敛拓扑后的组播组学习以及更新。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)# ipv6 mld snooping querier tcn query-count 5	设置 TCN 的查询次数
Switch(config)# ipv6 mld snooping querier tcn query-interval 20	设置 TCN 的查询时间间隔 20 秒

II. 命令验证

```
Switch # show ipv6 mld snooping querier
Global Mld Snooping Querier Configuration
-------------------------------------------------
Version                          :1
Last-Member-Query-Interval (msec) :2000
Max-Query-Response-Time (sec)     :5
Query-Interval (sec)              :100
Global Source-Address             :::
TCN Query Count                   :5
TCN Query Interval (sec)          :20
Vlan 1:   MLD snooping querier status
-----------------------------------------
Elected querier is : fe80::1
-----------------------------------------
Admin state                       :Enabled
Admin version                     :1
Operational state                 :Querier
Querier operational address       :fe80::1
Querier configure address         :fe80::1
Last-Member-Query-Interval (msec) :2000
Max-Query-Response-Time (sec)     :5
Query-Interval (sec)              :200
Querier-Timeout (sec)             :100
```

13.5.7 配置 MLD Snooping 报告抑制

交换机使用 MLD 报告抑制来同一个 MLD 报文重复发送给组播路由器。当 MLD 路由器抑制使能时(默认)，交换机将第一个 MLD 报告报文发送给组播路由器，其余同样的

MLD 报告报文将不再发送给组播路由器。这样就阻止了重复 MLD 报告报文发送给组播路由器了。

I. 配置

Switch# configure terminal	进入配置模式
Switch(config)# ipv6 mld snooping report-suppression	在全局模式下启用报告抑制
Switch(config)# ipv6 mld snooping vlan 1 report-suppression	在 VLAN1 模式下启用报告抑制

II. 命令验证

Switch # show ipv6 mld snooping

```
Global Mld Snooping Configuration
------------------------------------------------
Mld Snooping                                :Enabled
Mld Snooping Fast-Leave                     :Disabled
Mld Snooping Version                        :2
Mld Snooping Max-Member-Number              :4096
Mld Snooping Unknown Multicast Behavior     :Flood
Mld Snooping Report-Suppression             :Enabled
Vlan 1
-----------
Mld Snooping                                :Enabled
Mld Snooping Fast-Leave                     :Disabled
Mld Snooping Report-Suppression             :Enabled
Mld Snooping Version                        :2
Mld Snooping Max-Member-Number              :4096
Mld Snooping Unknown Multicast Behavior     :Flood
Mld Snooping Group Access-list              :N/A
Mld Snooping Mrouter Port                   :
Mld Snooping Mrouter Port Aging Interval(sec) :255
```

13.5.8 配置静态组播组

交换机在二层端口上收到 MLD 报文时会建立 MLD Snooping 的组记录。目前系统中也支持静态配置 MLD Snooping 的组记录，在静态配置时需要指定组地址，二层端口，以及二层端口所属的 VLAN。

I. 配置

Switch#configure terminal	进入配置模式
Switch(config)# ipv6 mld snooping vlan 1 static-group ff0e::1234 interface eth-0-2	配置静态组播组 ff0e::1234，成员端口是 vlan1 的 eth-0-2

II. 命令验证

Switch# show ipv6 mld snooping groups

```
VLAN  Interface  Group Address                              Uptime     Expire-time
1     eth-0-2    ff0e::1234                                 00:00:02   stopped
```

13.5.9 限制和配置指导

VRRP，RIPng，OSPFv3 等协议使用了组播 IPv6，因此在使能了 MLD Snooping 的网络中，要避免使用这样的组播 IPv6，这些组播 IPv6 是，它们映射出来的 MAC 和被协议模块使用的组播 IPv6 映射出来的 MAC 一致。

VRRP 使用了 ff02::12，因此组播 MAC 3333.0000.0012 映射出的组播 IPv6 在 MLD Snooping 和 VRRP 的网络中避免使用。

RIPv6 使用了 ff02::9，因此组播 MAC 3333.0000.0009 映射出的组播 IPv6 在 MLD Snooping 和 RIPng 的网络中避免使用。

OSPF 使用了 ff02::5，因此组播 MAC 3333.0000.0005 映射出的组播 IPv6 在 MLD Snooping 和 OSPFv3 的网络中避免使用。

13.6 配置 MVR6

13.6.1 简介

在传统的 IPv6 组播点播方式下，汇聚组播路由器下连一些接入交换机，接入交换机上连接了分布在不同 VLAN 中的用户。当这些不同 VLAN 的用户点播相同 Group 的节目时，汇聚的组播路由器需要为每个 VLAN 内的用户复制一份数据，每个 VLAN 的组播流量都要占用接入交换机的带宽。这样即增加了汇聚路由器的负担，也浪费接入设备的带宽。

MVR6(IPv6 组播 VLAN 注册)功能能够很好的解决这个问题。在靠近用户侧的接入交换机上启用组播 VLAN，汇聚路由器只需把组播数据在源 VLAN 内发送给接入交换机，而不必在每个用户 VLAN 内都复制一份，接入交换机收到组播数据后再根据用户请求进行复制，给每个 VLAN 内的用户发送一份组播数据。从而节省了网络带宽，也减轻了三层设备的负担。

MVR6 依赖于 MLD Snooping 进行工作，而且只有 MVR6 全局配置的 Group 才会生效。如果在 MVR6 的下游口上接收的 MLD 报文中组播组不在 MVR6 全局 Group 中，该报文将被忽略。通过在 MVR6 的下游口上接收的 MLD 报告/离开报文来维护接收者信息，MVR6 上游口收到 IPv6 组播数据后根据下游口的 IPv6 组播组信息来决定将 IPv6 组播数据从哪些 VLAN 的端口转发出去。

13.6.2 术语

MVR6：IPv6 组播 VLAN 注册

Source vlan：组播 VLAN 的源 VLAN

Source port：MVR6 网络中的上游口，连接组播路由器的端口

Receiver port：MVR6 网络中的下游口，连接接收者的端口

13.6.3 拓扑

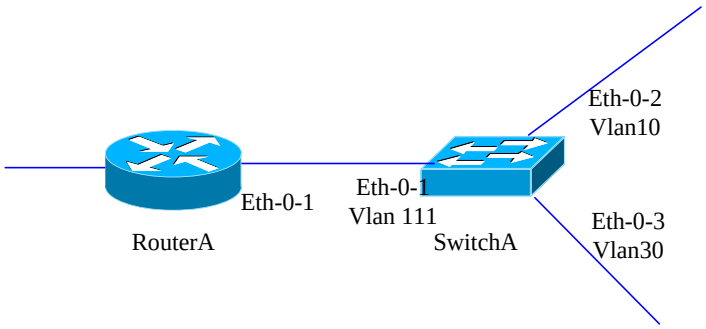


图13-3 IPv6 组播 VLAN 拓扑

13.6.4 配置

目的

在 Router A 的 eth-0-1 上启用 MLD&PIMv6-SM。

配置 Switch A：eth-0-1 属于 vlan111，eth-0-2 属于 vlan10，eth-0-3 属于 vlan30。

在 Switch A 启用 MVR6，从 Router A 到 Switch A 上拷贝一份组播流，在 Switch A 上再将这个组播流进行复制，从 eth-0-2 和 eth-0-3 发送出去。

Router A

在配置接口上启用 MLD&PIMv6-SM。

RouterA# configure terminal	进入配置模式
RouterA(config)# interface eth-0-1	进入接口模式
RouterA(config-if)# no switchport	设置端口为三层端口
RouterA(config-if)# no shutdown	使能端口
RouterA(config-if)# ipv6 address 2001:1::1/64	配置 IPv6 地址
RouterA(config-if)# ipv6 pim sparse-mode	启用 PIMv6-SM 协议
RouterA(config-if)# end	返回全局模式

Switch A

配置 eth-0-1 属于 vlan111，eth-0-2 属于 vlan10，eth-0-3 属于 vlan30。

SwitchA# configure terminal	进入配置模式
SwitchA(config)# vlan database	进入 VLAN 模式
SwitchA(config-vlan)# vlan 111,10,30	创建 vlan 111，10，30
SwitchA(config-vlan)# quit	退出 VLAN 模式
SwitchA(config)# interface vlan 111	进入 VLAN 接口模式
SwitchA(config-if)# exit	退出 VLAN 接口模式
SwitchA(config)# interface vlan 10	进入 VLAN 接口模式
SwitchA(config-if)# exit	退出 VLAN 接口模式
SwitchA(config)# interface vlan 30	进入 VLAN 接口模式
SwitchA(config-if)# exit	退出 VLAN 接口模式
SwitchA(config)# interface eth-0-1	进入接口模式
SwitchA(config-if)# switchport access vlan111	设置端口属于 VLAN111
SwitchA(config)# interface eth-0-2	进入接口模式
SwitchA(config-if)# switchport access vlan10	设置端口属于 VLAN10
SwitchA(config)# interface eth-0-3	进入接口模式
SwitchA(config-if)# switchport access vlan30	设置端口属于 VLAN30
SwitchA(config-if)# end	退出接口模式

在 switch A 启用 MVR6，这样从 Router A 到 Switch A 只会拷贝一份组播流，在 Switch A 上再将这个组播流从 eth-0-2 和 eth-0-3 发送出去。

SwitchA # configure terminal	进入配置模式
SwitchA(config)# no ipv6 multicast-routing	关闭 IPv6 组播路由
SwitchA(config)# mvr6	启用 MVR6
SwitchA(config)# mvr6 vlan 111	创建 MVR6 的 VLAN
SwitchA(config)# mvr6 group ff0e::1234 64	创建 IPv6 组播组
SwitchA(config)# mvr6 source-address fe80::1111	配置 MVR6 源地址

SwitchA(config)# interface eth-0-1	进入接口模式
SwitchA(config-if)# mvr6 type source	配置接口为 MVR6 的源端口
SwitchA(config)# interface eth-0-2	进入接口模式
SwitchA(config-if)# mvr6 type receiver vlan 10	设置接口为 MVR6 的接收端口
SwitchA(config)# interface eth-0-3	进入接口模式
SwitchA(config-if)# mvr6 type receiver vlan 30	设置接口为 MVR6 的接收端口
SwitchA(config-if)# end	退出接口模式

13.6.5 命令验证

Router A

```
RouterA # show ipv6 mld groups
MLD Connected Group Membership
Group Address                           Interface            Expires
ff0e::1234                              eth-0-2              00:03:01
ff0e::1235                              eth-0-2              00:03:01
ff0e::1236                              eth-0-2              00:03:01
ff0e::1237                              eth-0-2              00:03:01
ff0e::1238                              eth-0-2              00:03:01
.............
ff0e::1273                              eth-0-2              00:03:01
```

Switch A

```
SwitchA# show mvr6
MVR6 Running: TRUE
MVR6 Multicast VLAN: 111
MVR6 Source-address: fe80::111
MVR6 Max Multicast Groups: 1024
MVR6 Hw Rt Limit: 224
MVR6 Current Multicast Groups: 64
SwitchA# show mvr6 groups
VLAN  Interface  Group Address                          Uptime     Expire-time
10     eth-0-2   ff0e::1234                            00:03:23   00:02:03
10     eth-0-2   ff0e::1235                            00:03:23   00:02:03
10     eth-0-2   ff0e::1236                            00:03:23   00:02:03
10     eth-0-2   ff0e::1237                            00:03:23   00:02:03
10     eth-0-2   ff0e::1238                            00:03:23   00:02:03
10     eth-0-2   ff0e::1239                            00:03:23   00:02:03
......
10     eth-0-2   ff0e::1273                            00:03:23   00:02:03
```

14 RPC API 配置指导

14.1 管理配置

14.1.1 简介

RPC API 服务给用户通过软件远程控制交换机的能力, 目前只支持 JSON-RPC over HTTP 和 HTTP 基本认证。

14.1.2 配置 RPC API 服务

用户可以按照下列步骤启用 RPC API 服务, 默认使用 80 端口。

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# service rpc-api enable port 80	启动 RPC API 服务, 使用 TCP 80 (HTTP) 端口
Switch(config)# exit	退出配置模式

用户可以按照下列步骤, 关闭 RPC API 服务器:

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# service rpc-api disable	关闭 RPC API 服务
Switch(config)# exit	退出配置模式

14.1.3 配置 RPC API 服务的 HTTP 认证

用户可以配置 RPC API 服务器的 HTTP 认证。

目前只支持 HTTP Basic 基本认证, 用户认证失败将会返回 401 状态码。

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# username centec password centec	配置交换机的用户名(centec)和密码(centec), 可以用于 HTTP 认证
Switch(config)# service rpc-api auth-mode basic	启用 RPC API 服务的 HTTP 基本认证
Switch(config)# exit	退出配置模式

取消 HTTP 认证:

Switch1

Switch# configure terminal	进入配置模式
Switch(config)# no service rpc-api auth-mode	关闭 RPC API 服务的 HTTP 认证
Switch(config)# exit	退出配置模式

NOTE

启用或关闭 HTTP 认证后， 用户需要手动重启 RPC API 服务， 或者重启交换机，配置才能生效。

启用 RPC API 后会占用 2 个 imish 资源下发命令，当是有 show users 时 RPC API 占用 imish 的 idle time 一直是 0。

14.1.4 显示 RPC API 服务信息

显示当前 RPC API 的配置

Switch1

Switch# show services rpc-api	显示系统当前的 RPC API 服务配置

```
RPC API services configuration:
HTTP server:    running, port: 80, authentication mode: none
```

14.2 RPC API 规范

14.2.1 概述

RPC API 服务, 使用了标准的 JSON-RPC 规范。可以通过 JSON RPC method: ‘executeCmds’执行交换机的 CLI 命令行。默认初始模式为 EXEC(#)特权模式。

用户需要通过发送 JSON-RPC (over HTTP)请求 URL: ***http://<交换机管理口 IP 地址>:<端口号>/command-api***, 请求和返回的 JSON-RPC 格式如下:

14.2.2 JSON-RPC Request

```
{
    "params":[                                                  命令参数
        {
            "format":"text",                                        期望命令返回格式,
可以是’text’或者’json’, 默认’text’
            "version":1,                                        命令版本号
            "cmds":[                                             命令列表
                "show run",                                     命令行 1
                "config t",                                       命令行 2
                "vlan database",                                命令行 3
                "vlan 1-8",                                       命令行 4
                "interface eth-0-1",                           命令行 5
                "switchport mode trunk",                 命令行 6
                "switchport trunk allowed vlan add 2",      命令行 7
                "shutdown",                                      命令行 8
                "end",                                                 命令行 9
                "show interface switchport"          命令行 10
            ]
        }
    ],
    "jsonrpc":"2.0",                                                    JSON RPC 协
议版本号
    "method":"executeCmds",                                     运行交换机 CLI 命令的方
法
    "id":"70853aff-af77-420e-8f3c-fa9430733a19"   JSON RPC 协议中的 UID
}
```

14.2.3 JSON-RPC Response

```
{
    "jsonrpc":"2.0",                                                        JSON
RPC 协议版本号
    "id":"70853aff-af77-420e-8f3c-fa9430733a19",      JSON RPC 协议中的 UID

"result":[                                                              JSON
RPC 返回值列表
        {
            "sourceDetails":"version 5.1.6.fcs\n!\n …",      命令行 1 的返回值。
```

```
如果运行成功，原始文本输出在”sourceDetails” 属性中。
            "errorCode":-1003,                                                    如果有错，会
输出在
            "errorDesc":"unsupported command…",
warnings/errorCode/errorDesc 属性中。
"warnings":"% Invalid …",

JSON 格式化对象也会输出到这里。
        },
        { },
命令行 2 的返回值。
        { },
命令行 3 的返回值。
        { },
命令行 4 的返回值。
        { },
命令行 5 的返回值。
        { },
命令行 6 的返回值。
        { },
命令行 7 的返回值。
        { },
命令行 8 的返回值。
        { },
命令行 9 的返回值。
        {
            "sourceDetails":" Interface name          : eth-0-1\n Switchport
mode          : trunk\n …\n"
        }
命令行 10 的返回值。
    ]
}
```

14.2.4 Python Client 代码示例

以 pyjsonrpc 库为例，示例代码如下：

```
import pyjsonrpc
import json
http_client = pyjsonrpc.HttpClient(
    url = "http://10.10.39.64:80/command-api",
    username = "centec",
    password = "centec"
)
cmds = {}
cmd_list = ["show run", "config t", "vlan database", "vlan 1-8", "interface eth-0-
1", "switchport mode trunk", "switchport trunk allowed vlan add 2", "shutdown",
"end", "show interface switchport"]
cmds['cmds'] = cmd_list
cmds['format'] = 'text'
cmds['version'] = 1
try:
```

```
    response = http_client.call("executeCmds", cmds)
    print("json response:");
    json_result = json.dumps(response, indent=4)
    print(json_result)
except Exception, e:
    if e.code == 401:
        print "Unauthorized user"
    else:
        print e.message
        print e.data
```

14.2.5 JSON-RPC 错误码

下面列出了 JSON-RPC 2.0 的错误码：

错误码	描述
-32700	JSON 解析错误
-32600	无效请求
-32601	方法无效
-32602	无效参数
-32603	内部错误

14.2.6 RPC-API 错误码

下面列出了 RPC-API 的错误码：

错误码	描述
-1000	普通错误
-2001	不支持的 JSON RPC API 版本
-2002	JSON RPC 中必须指定’params’ 和 ‘cmds’
-2003	不支持的 JSON 返回格式
-3001	命令执行错误: 超时
-3002	命令执行错误: 不支持该命令
-3003	命令执行错误: 该命令未授权

-3004	命令执行错误: 找不到该命令
-3005	命令执行错误: 不能够转换为 JSON 格式
-3006	命令执行错误: 命令行数目太少
-3007	命令执行错误: 命令行数目太多

15 VPN 配置指导

15.1 VRF 配置

15.1.1 简介

VRF 简称 VPN 路由转发表，也称 VPN-instance(VPN 实例)，是 PE 为直接相连的站点建立并维护的一个专门实体，每个站点在 PE 上都有自己的 VPN-instance，每个 VPN-instance 包含到一个或多个与该 PE 直接相连的 CE 的路由和转发表。VRF 可以把一台路由器在逻辑上划分为多台虚拟的路由器，每台虚拟的路由器就像单独的一台路由器一样工作，有自己独立的路由表和相应的参与数据转发的接口，并且彼此业务隔离。这从根本上解决了多种业务并存于一台物理设备且又需要隔离的问题，能够节省用户在设备及通信资源方面的投资。

15.1.2 配置

Switch# configure terminal	进入配置模式
Switch(config)# ip vrf vpn1	创建 VRF，并且进入 VRF 配置模式
Switch(config-vrf)# rd 100:1	创建 RD。RD 可以是一个 AS 号和一个数字的组合(xxx:y)，也可以是一个 IP 地址和一个数字的组合(A.B.C.D:y)
Switch(config-vrf)# router-id 1.1.1.1	设置路由器标志
Switch(config-vrf)# route-target both 100:1	创建 RT。RT 可以是一个 AS 号和一个数字的组合(xxx:y)，也可以是一个 IP 地址和一个数字的组合(A.B.C.D:y)
Switch(config-vrf)# import map route-map	给 VRF 设置路由策略
Switch(config-vrf)# interface eth-0-1	进入接口模式
Switch(config-if)# no shutdown	将端口 up 起来
Switch(config-if)# no switch	设置端口为 3 层口
Switch(config-if)# ip vrf forwarding vpn1	将接口加入 VRF vpn1 下
Switch(config-if)# ip add 1.1.1.1/24	设置 IP 地址

Switch(config-if)# end	退出接口模式
Switch# show ip vrf vpn1	显示配置信息

15.1.3 命令验证

使用命令 **show ip vrf** 来验证配置会得到类似如下的屏幕回显信息。

```
Switch# show ip vrf
VRF vpn1, FIB ID 1
Router ID: 1.1.1.1 (config)
Interfaces:
  eth-0-1
DUT1# show ip vrf interfaces vpn1
Interface             IP-Address      VRF                    Protocol
eth-0-1               1.1.1.1         vpn1                   up
Switch# show ip vrf bgp brief
Name                    Default RD          Interfaces
vpn1                    100:1               eth-0-1
Switch# show ip vrf bgp detail
VRF vpn1; default RD 100:1
Interfaces:
  eth-0-1
VRF Table ID = 1
Export VPN route-target communities
  RT:100:1
Import VPN route-target communities
  RT:100:1
import-map: route-map
No export route-map
```

15.2 IPv4 over IPv4 GRE 隧道配置

15.2.1 简介

隧道技术是一种封装技术，它利用一种网络协议来传输另一种网络协议，即一种网络协议将其他网络协议的数据报文封装在自己的报文中，然后在网络中传输。封装后的数据报文在网络中传输的路径，称为隧道。隧道是一条虚拟的点对点连接，隧道的两端需要对数据报文进行封装及解封装。隧道技术就是指包括数据封装、传输和解封装在内的全过程。

当两个相隔离的 IPv4 网络需要相互通信，此时就需要在两个网络之间创建一个隧道机制。在 IPv4 网络上用于连接两个相隔离 IPv4 孤岛的 gre 隧道，称为 IPv4 gre 隧道， 即 IPv4 报文通过 gre 协议被封装在 IPv4 报文中，实现 IPv4 报文的透明传输。Gre 隧道协议在封装 IPv4 报文时会添加 gre 头，gre 头中包含 key，sequence，checksum 等可选信息。为了实现 gre 隧道，需要在 IPv4 网络与 IPv4 网络交界的边界交换机上启动 IPv4 双协议栈。

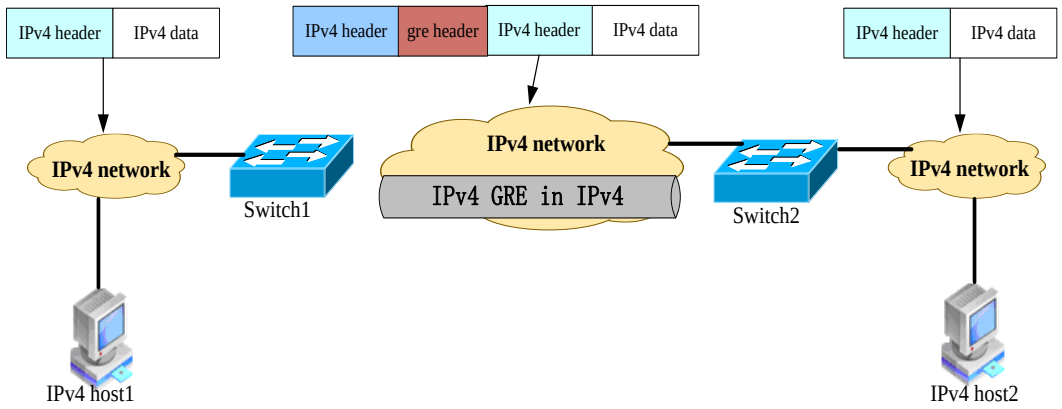


图15-1 ：IPv4 gre over IPv4 隧道原理图

IPv4 gre over IPv4 隧道对报文的处理过程如下：

IPv4 网络中的设备发送 IPv4 报文，该报文到达隧道的源端设备 Switch1。

Switch1 根据路由表判定该报文要通过隧道进行转发后，在 IPv4 报文前先封装上 gre 头然后再封装外层 IPv4 的报文头，通过隧道的实际物理接口将报文转发出去。

封装报文通过隧道到达隧道目的端设备 Switch2，Switch2 判断该封装报文的目的地是本设备后，将对报文进行解封装。

Switch2 根据解封装后的 IPv4 报文的目的地址转发该 IPv4 报文。如果目的地就是本设备，则将 IPv4 报文转给上层协议处理。在解封装过程中，会校验 gre 头中的 key 选项，只有当 key 相匹配时才会对该 IP4 报文作处理，否则丢弃。

这种技术的优点是，当 IPv4/IPv4 网络的边缘设备实现隧道功能，便可以将报文从一端透传到另外一端并可以进行报文校验，可以大大利用现有的 IPv4 网络投资。

II. IPv4 GRE 隧道

GRE 隧道的源和目的地址是手工指定的，它提供了一个点到点的连接。GRE 隧道可以建立在两个边界路由器之间为被 IPv4 网络分离的 IPv4 网络提供稳定的连接，或建立在终端系统与边界路由器之间为终端系统访问 IPv4 网络提供连接。

GRE 隧道要求在设备上手工配置隧道的源地址和目的地址，此外 gre key 配置是可选配置。

15.2.2 配置 IPv4 GRE 隧道

I. 拓扑

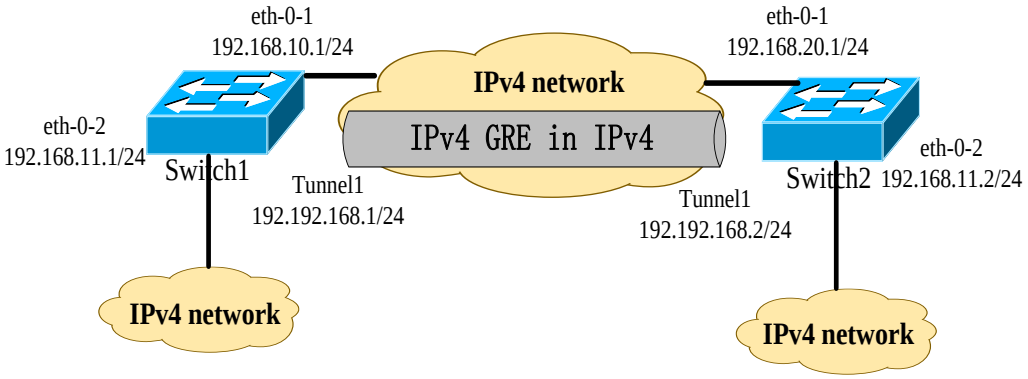


图15-2 ：配置 IPv4 GRE 隧道

如上图所示，两个 IPv4 网络分别通过 Switch1 和 Switch2 与 IPv4 网络连接，要求在 Switch1 和 Switch2 之间建立 IPv4 gre 隧道，使两个 IPv4 网络可以互通。

II. 配置

Switch1

配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口
Switch(config-if)# ip address 192.168.10.1/24	配置接口的 IPv4 地址
Switch(config)# ip route 192.168.20.0/24 192.168.10.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 192.168.10.2 0.0.2222	配置静态 ARP, 0.0.2222 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

配置 eth-0-2 的 IPv4 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ip address 192.168.11.1/24	配置接口的 IPv4 地址

配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel mode gre	配置 tunnel 模式为 gre
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel destination 192.168.20.1	配置 tunnel 的目的地
Switch(config-if)# tunnel gre key 100	配置 tunnel 的 gre key 为 100
Switch(config-if)# ip address192.192.168.1/24	配置 tunnel 接口的 IPv4 地址

配置 tunnel 的 keepalive 功能

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	进入 tunnel1 的接口模式
Switch(config-if)# keepalive 5 3	使能 tunnel1 的 keepalive 功能

配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

配置到达对端的静态 IPv4 路由

Switch# configure terminal	进入全局配置模式
Switch(config)# ip route 3.3.3.3/24 tunnel1	配置到达隧道对端的静态路由

Switch2

配置 IPv4 地址，使报文路由 3 层可达

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将 eth-0-1 配置为 3 层路由口

Switch(config-if)# ip address 192.168.20.1/24	配置接口的 IPv4 地址
Switch(config)# ip route 192.168.10.0/24 192.168.20.2	配置到达对端的 IPv4 静态路由
Switch(config)# arp 192.168.20.2 0.0.1111	配置静态 ARP, 0.0.1111 为下一跳的系统 MAC 地址。（该 ARP 条目也可以通过动态学习得到）

配置 eth-0-2 的 IPv4 地址

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	将 eth-0-2 配置为 3 层路由口
Switch(config-if)# ip address 192.168.11.2/24	配置接口的 IPv4 地址

配置 tunnel 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	创建 tunnel 虚接口
Switch(config-if)# tunnel mode gre	配置 tunnel 模式为 gre
Switch(config-if)# tunnel source eth-0-1	将 eth-0-1 口作为 tunnel 的源
Switch(config-if)# tunnel destination 192.168.10.1	配置 tunnel 的目的地
Switch(config-if)# tunnel gre key 100	配置 tunnel 的 gre key 为 100
Switch(config-if)# ip address192.192.168.2/24	配置 tunnel 接口的 IPv4 地址

配置 tunnel 的 keepalive 功能

Switch# configure terminal	进入全局配置模式
Switch(config)# interface tunnel1	进入 tunnel1 的接口模式
Switch(config-if)# keepalive 5 3	使能 tunnel1 的 keepalive 功能

配置 tunnel decap 接口

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口模式

Switch(config-if)# tunnel enable	使能 eth-0-1 口作 tunnel decap

6）配置到达对端的静态 IPv4 路由

Switch# configure terminal	进入全局配置模式
Switch(config)# ip route 4.4.4.4/24 tunnel1	配置到达隧道对端的静态路由

I. 检查配置结果

Switch1

```
Switch1# show interface tunnel1
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Internet primary address:
    192.192.168.1/24 pointopoint 192.192.168.255
  Tunnel protocol/transport GRE/IP, Status Valid
  Tunnel source 192.168.10.1(eth-0-1), destination 192.168.20.1
  Tunnel DSCP inherit,  Tunnel TTL 255
  Tunnel GRE key enable: 100
  Tunnel GRE keepalive enable, Send period: 5, Retry times: 3
  0 packets input, 0 bytes
  0 packets output, 0 bytes
```

Switch2

```
Switch2# show interface tunnel1
Interface tunnel1
  Interface current state: UP
  Hardware is Tunnel
  Index 8193 , Metric 1 , Encapsulation TUNNEL
  VRF binding: not bound
  Internet primary address:
    192.192.168.2/24 pointopoint 192.192.168.255
  Tunnel protocol/transport GRE/IP, Status Valid
  Tunnel source 192.168.20.1(eth-0-1), destination 192.168.10.1
  Tunnel DSCP inherit,  Tunnel TTL 255
  Tunnel GRE key enable: 100
  Tunnel GRE keepalive enable, Send period: 5, Retry times: 3
  0 packets input, 0 bytes
  0 packets output, 0 bytes
```

说明

必须使 IPv4 报文 3 层路由可达，否则会造成 tunnel 报文转发失败。

tunnel 接口上必须配置 IPv4 地址，否则配置在该接口上的路由无效。

16 可靠性配置指导

16.1 BHM 配置

16.1.1 简介

BHM 是用于监控其他协议进程的一个模块，当某个受监控进程长时间（30 秒）无响应时，BHM 模块会采取措施恢复系统，或者提示用户恢复系统。这些措施包括在终端打印警告信息、关闭所有端口，或者重启系统。系统使用何种措施是可配置的，默认是重启系统。

BHM 监控的协议模块有：RIP，RIPNG，OSPF，OSPF6，BGP，LDP，RSVP，PIM，PIM6，802.1X，LACP，MSTP，DHCP-RELAY，DHCP-RELAY6，RMON，OAM，ONM，SSH，SNMP，PTP，SSM，以及一些系统进程：NSM,，MI，CHSM，HSRVD。

16.1.2 术语

BHM：Beat heart Monitor

16.1.3 配置

Switch1# configure terminal	进入全局配置模式
Switch1(config)# sysmon enable	使能系统监控功能
Switch1(config)# heart-beat-monitor enable	使能 BHM 模块功能
Switch1(config)# heart-beat-monitor reactivate reload system	配置监控措施为“重启系统”

16.1.4 命令验证

使用如下命令可以显示配置结果。

Switch1# show heart-beat-monitor

```
heart-beat-monitor enable.
heart-beat-monitor reactivation: restart system.
```

16.2 CFM 配置

本章包括一个连接错误管理（CFM）配置的完整用例。如果想了解用例中使用命令的详细信息，请参考 CFM 的命令手册。为防止重复，常用命令将不列在 CFM 命令手册中。

16.2.1 简介

CFM 能够根据 802.1ag 协议去检测，验证，定位和通知以太网中的连接故障。CFM 也能够发现和验证以太网中的路径。CFM 是操作维护管理模块 OAM 的一部份。CFM 对用户数据报文是透明的并且能最大限度地发现连接错误。

CFM 使用标准的以太网报文，仅以太网协议类型不同。支持的 CFM 消息如下：

- 连续性检查(CC)消息

  连续性检查消息周期性发送，使维护域端点 MEP 和维护域中间节点 MIP 可以发现其它 MEP。它用于检测任何一对 MEP 间的连续性丢失(LOC)。

- 以太网环回(LB)消息

  MEP 发送一个 LB 消息来验证另一个 MEP 或 MIP 是否可达。LB 消息和网间控制消息协议 ICMP ping 消息类似。

- 以太网链路追踪(LT)消息

  MEP 发送 LT 消息以追踪到一个目的 MEP/MIP 中间每一跳的 MIP。LT 消息类似 UDP 链路追踪消息。

- 以太网帧延时测量(DM)消息

ETH-DM 可用于按需的 OAM，测量帧时延和帧时延变化。帧时延和帧时延变化的测量是通过向对等 MEP 周期地发送带有 ETH-DM 信息的帧，并在诊断间隔内从对等 MEP 接收带有 ETH-DM 信息的帧来完成的。每一个 MEP 都可以进行帧时延和帧时延变化的测量。

当一个 MEP 能产生带有 ETH-DM 信息的帧时，它向同一 ME 内它对等的 MEP 周期地发送带有 ETH-DM 信息的帧。当一个 MEP 产生带有 ETH-DM 信息的帧时，它也预期在同一 ME 中从对等的 MEP 接收带有 ETH-DM 信息的帧。

- 以太网锁定信号(LCK)消息

  以太网锁定信号功能（ETH-LCK）用于通告服务器层（子层）MEP 的管理性锁定以及随后的数据业务流中断，该业务流是送往期待接收这业务流的 MEP 的。它使得接收带有 ETH-LCK 信息的帧的 MEP 能区分是故障情况，还是服务器层（子层）MEP 的管理性锁定动作。

- 以太网客户端信号失败功能(CSF)

  以太网客户端信号失败功能是由服务器 MEP 将客户端检查到的失败或错误通知到对应服务器 MEP，再由对端服务器 MEP 通知对端的客户端 MEP。通常应用于客户端 MEP 无法直接通知到对端 MEP 的情况，此时 CC 或 AIS 是均无法使用的。CSF 消息是由服务器 MEP 发给对端服务器的 MEP 的消息。CSF 仅应用于点对点的以太网传输中。

- 以太网帧丢失测量(LM)消息

ETH-LM 用于收集计数器的数值，应用于入口和出口处的服务帧，在此计数器在一对 MEP 之间保持着发送和接收的数据帧的计数。

ETH-LM 是通过向其对等 MEP 发送带有 ETH-LM 信息的帧，并类似地从对等 MEP 接收带有 ETH-LM 信息的帧实现的。

16.2.2 参考

IEEE 802.1ag/D8.1

16.2.3 限制

CFM 和 802.1x 及 mirror destination 配在同一端口上时功能冲突，因为 CFM 不应该与这些功能配在同一端口上；

16.2.4 配置 CC/LB/LT/AIS/DM

I. 拓扑

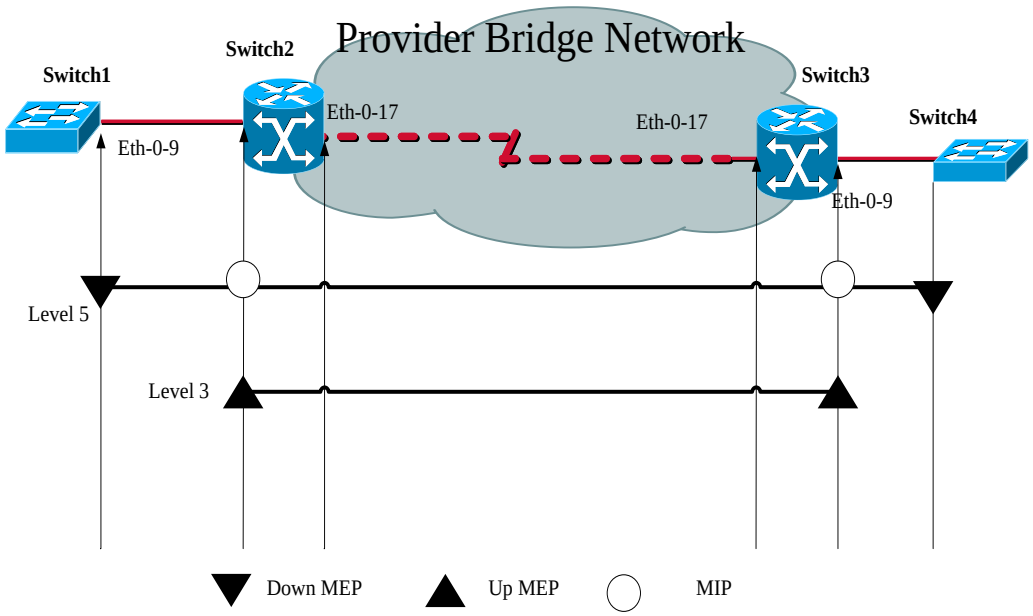


图16-1 CFM 拓扑

II. 配置

Switch 1

Switch1# configure terminal	进入全局配置模式
Switch1(config)# vlan database	进入 VLAN 配置模式
Switch1(config vlan)# vlan 30	创建 VLAN 30

Switch1(config vlan)# exit	退出 VLAN 配置模式
Switch1(config)# ethernet cfm enable	全局使能 CFM
Switch1(config)# ethernet cfm mode y1731	配置 CFM 模式
Switch1(config)# ethernet cfm domain cust level 5	创建维护域 cust
Switch1(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch1(config-ether-cfm)# exit	退出 CFM 配置模式
Switch1(config)# interface eth-0-9	进入端口模式
Switch1(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch1(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过
Switch1(config-if)# ethernet cfm mep down mpid 66 domain cust vlan 30 interval 1	创建维护域端点
Switch1(config-if)# ethernet cfm mep crosscheck mpid 99 domain cust vlan 30 mac d036.4567.8009	创建维护域远端节点，mac 为远端 mep 的 mac
Switch1(config-if)# no shutdown	打开端口
Switch1(config-if)# exit	退出端口模式
Switch1(config)# ethernet cfm cc enable domain cust vlan 30	启用维护域 cust 的服务 cst 的连续性检查功能
Switch1(config)# ethernet cfm ais suppress alarm enable domain cust vlan 30	配置当收到告警指示信号 ais 报文且存在 loc 错误时抑制其它 loc error
Switch1(config)# end	退出全局配置模式

Switch 2

Switch2# configure terminal	进入全局配置模式
Switch2(config)# vlan database	进入 VLAN 配置模式
Switch2(config-vlan)# vlan 30	创建 VLAN 30
Switch2(config-vlan)# exit	退出 VLAN 配置模式
Switch2(config)# ethernet cfm enable	全局使能 CFM
Switch2(config)# ethernet cfm mode y1731	配置 CFM 模式

Switch2(config)# ethernet cfm domain cust level 5	创建维护域 cust
Switch2(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch2(config-ether-cfm)# exit	退出 CFM 配置模式
Switch2(config)# ethernet cfm domain provid level 3	创建维护域 provid
Switch2(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch2(config-ether-cfm)# exit	退出 CFM 配置模式
Switch2(config)# interface eth-0-9	进入端口模式
Switch2(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch2(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过
Switch2(config-if)# ethernet cfm mip level 5 vlan 30	创建维护域中间节点
Switch2(config-if)# ethernet cfm mep up mpid 666 domain provid vlan 30 interval 1	创建维护域端点
Switch2(config-if)# ethernet cfm mep crosscheck mpid 999 domain provid vlan 30 mac 6a08.051e.bd09	创建维护域远端节点，mac 为远端 mep 的 mac
Switch2(config-if)# ethernet cfm ais status enable all domain provid vlan 30 level 5 multicast	使能 ais
Switch2(config-if)# ethernet cfm server-ais status enable level 5 interval 1	配置 ais 服务器
Switch2(config-if)# no shutdown	打开端口
Switch2(config-if)# exit	退出端口模式
Switch2(config)# interface eth-0-17	进入端口模式
Switch2(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch2(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过
Switch2(config-if)# no shutdown	打开端口
Switch2(config-if)# exit	退出端口模式
Switch2(config)# ethernet cfm cc enable domain provid vlan 30	启用维护域 provid 的服务 cst 的连续

	性检查功能
Switch2(config)# end	退出全局配置模式

Switch3

Switch3# configure terminal	进入全局配置模式
Switch3(config)# vlan database	进入 VLAN 配置模式
Switch3(config-vlan)# vlan 30	创建 VLAN 30
Switch3(config-vlan)# exit	退出 VLAN 配置模式
Switch3(config)# ethernet cfm enable	全局使能 CFM
Switch3(config)# ethernet cfm mode y1731	配置 CFM 模式
Switch3(config)# ethernet cfm domain cust level 5	创建维护域 cust
Switch3(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch3(config-ether-cfm)# exit	退出 CFM 配置模式
Switch3(config)# ethernet cfm domain provid level 3	创建维护域 provid
Switch3(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch3(config-ether-cfm)# exit	退出 CFM 配置模式
Switch3(config)# interface eth-0-9	进入端口模式
Switch3(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch3(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过
Switch3(config-if)# ethernet cfm mip level 5 vlan 30	创建维护域中间节点
Switch3(config-if)# ethernet cfm mep up mpid 999 domain provid vlan 30 interval 1	创建维护域端点
Switch3(config-if)# ethernet cfm mep crosscheck mpid 666 domain provid vlan 30 mac 0e1d.a7d7.fb09	创建维护域远端节点，mac 为远端 mep 的 mac
Switch3(config-if)# no shutdown	打开端口

Switch3(config-if)# exit	退出端口模式
Switch3(config)# interface eth-0-17	进入端口模式
Switch3(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch3(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过
Switch3(config-if)# no shutdown	打开端口
Switch3(config-if)# exit	退出端口模式
Switch3(config)# ethernet cfm cc enable domain provid vlan 30	启用维护域 provid 的服务 cst 的连续性检查功能
Switch3(config)# end	退出全局配置模式

Switch4

Switch4# configure terminal	进入全局配置模式
Switch4(config)# vlan database	进入 VLAN 配置模式
Switch4(config vlan)# vlan 30	创建 VLAN 30
Switch4(config vlan)# exit	退出 VLAN 配置模式
Switch4(config)# ethernet cfm enable	全局使能 CFM
Switch4(config)# ethernet cfm mode y1731	配置 CFM 模式
Switch4(config)# ethernet cfm domain cust level 5	创建维护域 cust
Switch4(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch4(config-ether-cfm)# exit	退出 CFM 配置模式
Switch4(config)# interface eth-0-9	进入端口模式
Switch4(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch4(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过
Switch4(config-if)# ethernet cfm mep down mpid 99 domain cust vlan 30 interval 1	创建维护域端点
Switch4(config-if)# ethernet cfm mep crosscheck mpid 66 domain cust vlan 30 mac fa02.cdff.6a09	创建维护域远端节点，mac 为远端 mep 的 mac

Switch4(config-if)# no shutdown	打开端口
Switch4(config-if)# exit	退出端口模式
Switch4(config)# ethernet cfm cc enable domain cust vlan 30	启用维护域 cust 的服务 cst 的连续性检查功能
Switch4(config)# end	退出全局配置模式

I. 命令验证

检查 MEP and MIP

以下命令可用于查看 Switch1 和 Switch2 上 MEP 和 MIP 的相关信息。

Switch1# show ethernet cfm maintenance-points

```
######Local MEP:
MPID Direction DOMAIN LEVEL TYPE VLAN PORT     CC-Status Mac-address    RDI
Interval
--------------------------------------------------------------------------------
---
66   Down MEP   cust    5      MEP  30   eth-0-9  enabled   fa02.cdff.6a09 True
3.33ms
######Local MIP:
Level   VID  TYPE    PORT           MAC
------------------------------------------------------
######Remote MEP:
MPID  LEVEL VLAN ACTIVE Remote Mac     RDI   FLAGS    STATE
------------------------------------------------------------
99    5     30   Yes    d036.4567.8009 True  Learnt    UP
```

Switch2# show ethernet cfm maintenance-points

```
######Local MEP:
MPID Direction DOMAIN LEVEL TYPE VLAN PORT     CC-Status Mac-address    RDI
--------------------------------------------------------------------------------
666  Up MEP     provid 3      MEP  30    eth-0-9  enabled   0e1d.a7d7.fb09 False
######Local MIP:
Level   VID  TYPE    PORT           MAC
------------------------------------------------------
5       30    MIP     eth-0-9        0e1d.a7d7.fb09
######Remote MEP:
MPID  LEVEL VLAN ACTIVE Remote Mac      RDI   FLAGS     STATE
------------------------------------------------------------
999   3      30    Yes    6a08.051e.bd09  True  Learnt    UP
```

以太网回环检查

以下命令用于根据远端 MEP 的地址回环远端 MEP。

```
Switch1# ethernet cfm loopback mac d036.4567.8009 unicast mepid 66 domain cust vlan
30
Sending 1 Ethernet CFM loopback messages, timeout is 5 seconds:
(! Pass . Fail)
!
 Loopback completed.
 ----------------------------------
Success rate is 100 percent(1/1)
```

以下命令用于根据组播地址回环远端 MEP。

```
Switch1# ethernet cfm loopback multicast mepid 66 domain cust vlan 30
Sending 1 Ethernet CFM loopback messages, timeout is 5 seconds:
(! Pass . Fail)
Host MEP: 66
Number of RMEPs that replied to mcast frame = 1
LBR received from the following
  9667.bb68.f308
 success rate is  100 (1/1)
```

以下命令用于根据远端 MEP 的标识回环远端 MEP。

```
Switch1# ethernet cfm loopback unicast rmepid 99 mepid 66 domain cust vlan 30
Sending 1 Ethernet CFM loopback messages, timeout is 5 seconds:
(! Pass . Fail)
!
 Loopback completed.
 ----------------------------------
 Success rate is 100 percent(1/1)
```

以下命令用于根据远端 MIP 的地址回环远端 MIP。

```
Switch1# ethernet cfm loopback mac 0e1d.a7d7.fb09 unicast mepid 66 domain cust vlan
30
Sending 1 Ethernet CFM loopback messages, timeout is 5 seconds:
(! Pass . Fail)
!
 Loopback completed.
 ----------------------------------
 Success rate is 100 percent(1/1)
```

检查远端故障指示 RDI

以下命令显示 RDI 的信息。

```
Switch1# show ethernet cfm maintenance-points local mep domain cust
MPID Direction DOMAIN LEVEL TYPE VLAN PORT    CC-Status Mac-address    RDI Interval
--------------------------------------------------------------------------------
66   Down MEP  cust   5  MEP  30   eth-0-9  enabled   fa02.cdff.6a09 True 3.33ms
```

错误检查

在清除本地 MEP 错误前，错误信息显示如下所示。

```
Switch1# show ethernet cfm errors domain cust
Level Vlan MPID RemoteMac      Reason                              ServiceId
```

```
5      30   66  d036.4567.8009 errorCCMdefect: rmep not found       cst
5      30   66  d036.4567.8009 errorCCMdefect: rmep not found clear cst
Time
2011/05/27 3:19:18
2011/05/27 3:19:32
```

以下命令用于清除错误信息。

```
Switch1# clear ethernet cfm errors domain cust
```

当清除本地 MEP 的错误信息后，错误信息如下所示。

```
Switch1# clear ethernet cfm errors domain cust
Level Vlan MPID RemoteMac      Reason                              ServiceId
```

检查 AIS

以下命令用于关闭 Switch1 上的连续性检查功能。

```
Switch1(config)# no ethernet cfm cc enable domain cust vlan 30
```

以下命令用于关闭 Switch3 上的连续性检查功能。

```
Switch3(config)# no ethernet cfm cc enable domain cust vlan 30
```

以下命令用于检查 Switch2 上的 ais 的状态。

```
Switch2# show ethernet cfm ais mep 666 domain cust vlan 30
AIS-Status: Enabled
AIS Period: 1
Level to transmit AIS: 7
AIS Condition: No
-----------------------------------------------------
   Configured defect condition     detected(yes/no)
-----------------------------------------------------
   unexpected-period                    no
   unexpected-MEG level            no
   unexpected-MEP                       no
   Mismerge                             no
   LOC                             yes
```

以下命令用于检查 Switch1 上 ais 状态。

```
Switch1# show ethernet cfm ais mep 66 domain cust vlan 30
AIS-Status: Disabled
AIS Condition: Yes
```

检查 LinkTrace

以下命令用于根据远端 MEP 的地址追踪远端 MEP。

```
Switch1# ethernet cfm linktrace mac d036.4567.8009 mepid 66 domain cust vlan 30
Sending Ethernet CFM linktrace messages,TTL is 64.Per-Hop Timeout is 5 seconds:
Please wait a moment
------------------------------
Received Hops: 1
------------------------------
```

```
TTL                         : 63
Fowarded                        : True
Terminal MEP                    : False
Relay Action                    : Rly FDB
Ingress Action              : IngOk
Ingress MAC address         : 0e1d.a7d7.fb09
Ingress Port ID Type        : ifName
Ingress Port ID                 : eth-0-9
--------------------------------
Received Hops: 2
--------------------------------
TTL                         : 62
Fowarded                    : True
Terminal MEP                : False
Relay Action                : Rly FDB
Egress Action               : EgrOk
Egress MAC address          : 6a08.051e.bd09
Egress Port ID Type         : ifName
Egress Port ID              : eth-0-9
--------------------------------
Received Hops: 3
--------------------------------
TTL                         : 61
Fowarded                    : False
Terminal MEP                : True
Relay Action                : Rly Hit
Ingress Action              : IngOk
Ingress MAC address         : d036.4567.8009
Ingress Port ID Type        : ifName
Ingress Port ID             : eth-0-9
```

以下命令用于根据远端 MEP 的标识追踪远端 MEP。

```
Switch1# ethernet cfm linktrace rmepid 99 mepid 66 domain cust vlan 30
Sending Ethernet CFM linktrace messages,TTL is 64.Per-Hop Timeout is 5 seconds:
Please wait a moment
--------------------------------
Received Hops: 1
--------------------------------
TTL                         : 63
Fowarded                        : True
Terminal MEP                    : False
Relay Action                    : Rly FDB
Ingress Action              : IngOk
Ingress MAC address         : 0e1d.a7d7.fb09
Ingress Port ID Type        : ifName
Ingress Port ID             : eth-0-9
--------------------------------
Received Hops: 2
--------------------------------
TTL                         : 62
Fowarded                    : True
Terminal MEP                : False
Relay Action                : Rly FDB
Egress Action               : EgrOk
```

```
Egress MAC address        : 6a08.051e.bd09
Egress Port ID Type       : ifName
Egress Port ID            : eth-0-9
-------------------------------
Received Hops: 3
-------------------------------
TTL                       : 61
Fowarded                  : False
Terminal MEP              : True
Relay Action              : Rly Hit
Ingress Action            : IngOk
Ingress MAC address       : d036.4567.8009
Ingress Port ID Type      : ifName
Ingress Port ID           : eth-0-9
```

以下命令用于根据远端 MIP 的地址追踪远端 MIP。

```
Switch1# ethernet cfm linktrace 6a08.051e.bd09 mepid 66 domain cust vlan 30
Sending Ethernet CFM linktrace messages,TTL is 64.Per-Hop Timeout is 5 seconds:
Please wait a moment
-------------------------------
Received Hops: 1
-------------------------------
TTL                       : 63
Fowarded                  : True
Terminal MEP              : False
Relay Action              : Rly FDB
Ingress Action            : IngOk
Ingress MAC address       : 0e1d.a7d7.fb09
Ingress Port ID Type      : ifName
Ingress Port ID           : eth-0-9
-------------------------------
Received Hops: 2
-------------------------------
TTL                       : 62
Fowarded                  : False
Terminal MEP              : False
Relay Action              : Rly Hit
Egress Action             : EgrOk
Egress MAC address        : 6a08.051e.bd09
Egress Port ID Type       : ifName
Egress Port ID            : eth-0-9
```

帧时延测量检查

以下命令用于测量双向的延时和延时变化：

```
Switch1# ethernet cfm dmm  rmepid 99 mepid 66 count 5 domain cust vlan 30
     Delay measurement statistics:
     DMM Packets transmitted        : 5
     Valid DMR packets received     : 5
     Index      Two-way delay      Two-way delay variation
         1          4288 usec                       0 usec
         2          4312 usec                      24 usec
         3          4296 usec                      16 usec
```

```
        4          4320 usec                    24 usec
        5          4264 usec                    56 usec
    Average delay                 : 4296 usec
    Average delay variation       : 24 usec
    Best case delay               : 4264 usec
    Worst case delay              : 4320 usec
```

在启用单向时延测量前，时钟应该同步。以下命令显示了在 Switch1 上启用了单向时延测量：

```
Switch1#ethernet cfm 1dm rmepid 99 mepid 66 count 5 domain cust vlan 30
```

以下命令在 Switch4 上显示了单向时延测量的结果：

```
Switch4# show ethernet cfm delaymeasurement cache
    Remote MEP       : 66
    Remote MEP vlan  : 30
    Remote MEP level : 5
    DMM Packets transmitted           : 0
    Valid DMR packets received        : 0
    Valid 1DM packets received        : 5
    Index   One-way delay   One-way delay variation   Received Time
        1      16832 usec                   0 usec   2011/07/19 17:27:46
        2      16176 usec                 656 usec   2011/07/19 17:27:47
        3      15448 usec                 728 usec   2011/07/19 17:27:48
        4      14800 usec                 648 usec   2011/07/19 17:27:49
        5      15406 usec                 606 usec   2011/07/19 17:27:50
    Average delay                 : 15732 usec
    Average delay variation       : 527 usec
    Best case delay               : 14800 usec
    Worst case delay              : 16832 usec
```

16.2.5 配置 LCK

I. 拓扑

请参考 1.3.1.

II. 配置

请参考 1.3.2 中关于 MD/MA/MEP 的配置。

在 Switch2 上配锁定。

Switch 2

Switch2# configure terminal	进入全局配置模式
Switch2(config)# interface eth-0-9	进入端口模式
Switch2(config-if)# ethernet cfm lck enable mep 666 domain provid vlan 30 tx-level 5 interval 1	配置锁定 MEP 666，并指定向等级 5 上按周期 1 秒发锁定报文

Switch2(config-if)# end	退出端口模式

I. 命令验证

以下命令用来显示 Switch2 上 LCK 状态：

Switch2# show ethernet cfm lck

```
En-LCK Enable, Y(Yes)/N(No)
Rx-LC, Receive LCK packets and enter LCK condition, Y(Yes)/N(No)
Rx-I, The period which is gotten from LCK packets
Tx-Domain, frames with ETH-LCK information are sent to this Domain
Tx-I, Transmit Interval
--------------------------------------------------------------------------
MPID Domain       VLAN En Rx-LC Rx-I Tx-Domain    Tx-I
--------------------------------------------------------------------------
666  provid       30   Y  N     N/A  cust         1
```

以下命令用来显示 Switch1 上 LCK 状态：

Switch1# show ethernet cfm lck

```
En-LCK Enable, Y(Yes)/N(No)
Rx-LC, Receive LCK packets and enter LCK condition, Y(Yes)/N(No)
Rx-I, The period which is gotten from LCK packets
Tx-Domain, frames with ETH-LCK information are sent to this Domain
Tx-I, Transmit Interval
--------------------------------------------------------------------------
MPID Domain       VLAN En Rx-LC Rx-I Tx-Domain    Tx-I
--------------------------------------------------------------------------
66   cust         30   N  Y     1    N/A          N/A
```

16.2.6 配置 CSF

I. 拓扑

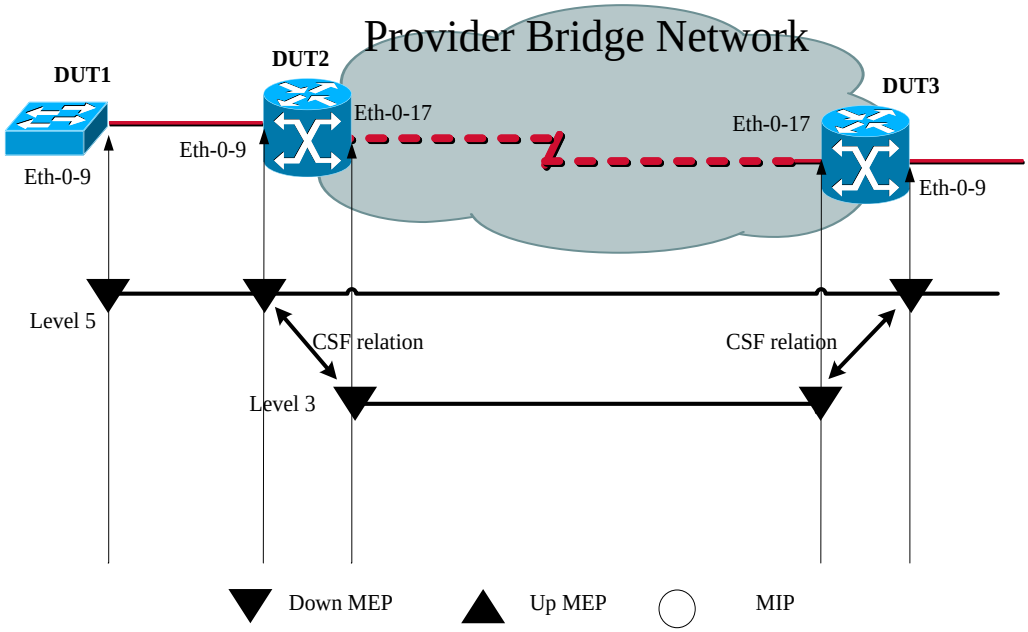


图16-2 CFM CSF 拓扑

II. 配置

Switch 1

Switch1# configure terminal	进入全局配置模式
Switch1(config)# vlan database	进入 VLAN 配置模式
Switch1(config vlan)# vlan 30	创建 VLAN 30
Switch1(config vlan)# exit	退出 VLAN 配置模式
Switch1(config)# ethernet cfm enable	全局使能 CFM
Switch1(config)# ethernet cfm mode y1731	配置 CFM 模式
Switch1(config)# ethernet cfm domain cust level 5	创建维护域 cust
Switch1(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch1(config-ether-cfm)# exit	退出 CFM 配置模式
Switch1(config)# interface eth-0-9	进入端口模式
Switch1(config-if)# switchport mode trunk	配置端口为 trunk 口

Switch1(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过
Switch1(config-if)# ethernet cfm mep down mpid 66 domain cust vlan 30 interval 1	创建维护域端点
Switch1(config-if)# ethernet cfm mep crosscheck mpid 99 domain cust vlan 30 mac d036.4567.8009	创建维护域远端节点，mac 为远端 mep 的 mac
Switch1(config-if)# no shutdown	打开端口
Switch1(config-if)# exit	退出端口模式
Switch1(config)# ethernet cfm cc enable domain cust vlan 30	启用维护域 cust 的服务 cst 的连续性检查功能
Switch1(config)# end	退出全局配置模式

Switch 2

Switch2# configure terminal	进入全局配置模式
Switch2(config)# vlan database	进入 VLAN 配置模式
Switch2(config vlan)# vlan 20,30	创建 VLAN 20,30
Switch2(config vlan)# exit	退出 VLAN 配置模式
Switch2(config)# ethernet cfm enable	全局使能 CFM
Switch2(config)# ethernet cfm mode y1731	配置 CFM 模式
Switch2(config)# ethernet cfm domain cust level 5	创建维护域 cust
Switch2(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch2(config)# ethernet cfm domain provid level 3	创建维护域 provid
Switch2(config-ether-cfm)# service cst vlan 20	创建服务 cst
Switch2(config-ether-cfm)# exit	退出 CFM 配置模式
Switch2(config)# interface eth-0-9	进入端口模式
Switch2(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch2(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过

Switch2(config-if)# ethernet cfm mep down mpid 99 domain cust vlan 30 interval 1	创建维护域端点
Switch2(config-if)# ethernet cfm mep crosscheck mpid 66 domain cust vlan 30 mac fa02.cdff.6a09	创建维护域远端节点，mac 为远端 mep 的 mac
Switch2(config-if)# no shutdown	打开端口
Switch2(config-if)# exit	退出端口模式
Switch2 (config)#interface eth-0-17	进入端口模式
Switch2 (config-if)# switchport mode trunk	配置端口为 trunk 口
Switch2 (config-if)# switchport trunk allowed vlan add 20	配置端口允许 vlan 20 通过
Switch2 (config-if)# ethernet cfm mep down mpid 666 domain provid vlan 20 interval 1	创建维护域端点
Switch2 (config-if)# no shutdown	打开端口
Switch2(config)# ethernet cfm cc enable domain cust vlan 30	启用维护域 cust 的服务 cst 的连续性检查功能
DUT (config)# ethernet cfm csf client domain cust vlan 30 mepid 99 server domain provid vlan 20 mepid 666 interval 1	配置 CSF 连接关系
Switch2(config)# end	退出全局配置模式

Switch 3

Switch3# configure terminal	进入全局配置模式
Switch3(config)# vlan database	进入 VLAN 配置模式
Switch3(config vlan)# vlan 20,30	创建 VLAN 20,30
Switch3(config vlan)# exit	退出 VLAN 配置模式
Switch3(config)# ethernet cfm enable	全局使能 CFM
Switch3(config)# ethernet cfm mode y1731	配置 CFM 模式
Switch3(config)# ethernet cfm domain cust level 5	创建维护域 cust
Switch3(config-ether-cfm)# service cst vlan 30	创建服务 cst
Switch3(config)# ethernet cfm domain provid level 3	创建维护域 provid

Switch3(config-ether-cfm)# service cst vlan 20	创建服务 cst
Switch3(config-ether-cfm)# exit	退出 CFM 配置模式
Switch3(config)# interface eth-0-9	进入端口模式
Switch3(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch3(config-if)# switchport trunk allowed vlan add 30	配置端口允许 vlan 30 通过
Switch3(config-if)# ethernet cfm mep down mpid 88 domain cust vlan 30 interval 1	创建维护域端点
Switch3(config-if)# no shutdown	打开端口
Switch3(config-if)# exit	退出端口模式
Switch3(config)#interface eth-0-17	进入端口模式
Switch3(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch3(config-if)# switchport trunk allowed vlan add 20	配置端口允许 vlan 20 通过
Switch3(config-if)# ethernet cfm mep down mpid 999 domain provid vlan 20 interval 1	创建维护域端点
Switch3(config-if)# no shutdown	打开端口
Switch3(config)# ethernet cfm cc enable domain cust vlan 30	启用维护域 cust 的服务 cst 的连续性检查功能
Switch3(config)# ethernet cfm csf client domain cust vlan 30 mepid 88 server domain provid vlan 20 mepid 999 interval 1	配置 CSF 连接关系
Switch3(config)# end	退出全局配置模式

I. 命令验证

以下命令用来关闭 Switch1 上的连续性检查，使 Switch2 上触发 loc 错误：

```
Switch1 (config)#no ethernet cfm cc enable domain cust vlan 30
```

Switch2 上的 MEP 99 会上报 loc,引发 MEP 666 发送 CSF 报文(原因 los)：

以下命令用来显示 Switch2 上 CSF 状态：

```
Switch2# show ethernet cfm csf
En-CSF Enable, Y(Yes)/N(No)
CTR-Client Trigger reason, L(los)/F(fdi)/R(rdi)/D(dci) or N/A
ECC-Enter CSF Condition, Y(Yes)/N(No)
SRR-Server Rx Reason, L(los)/F(fdi)/R(rdi)/D(dci) or N/A
Tx-I, Transmit Interval
```

```
Rx-I, The period which is gotten from CSF packets
--------------------------------------------------------------------------
Client Mep                        Server Mep
MPID Cli-Domain   VLAN CTR   ECC MPID Srv-Domain   VLAN SRR   Tx-I Rx-I
--------------------------------------------------------------------------
99   cust         30   L      N   666  provid       20   N/A   1    N/A
```

在 Switch3 上，MEP 999 会收到 CSF 的报文并通知客户 MEP 88， 然后客户 MEP 会进入 CSF 状态。

以下命令用来显示 Switch3 上 CSF 状态：

```
Switch3# show ethernet cfm csf
En-CSF Enable, Y(Yes)/N(No)
CTR-Client Trigger reason, L(los)/F(fdi)/R(rdi)/D(dci) or N/A
ECC-Enter CSF Condition, Y(Yes)/N(No)
SRR-Server Rx Reason, L(los)/F(fdi)/R(rdi)/D(dci) or N/A
Tx-I, Transmit Interval
Rx-I, The period which is gotten from CSF packets
--------------------------------------------------------------------------
Client Mep                        Server Mep
MPID Cli-Domain   VLAN CTR   ECC MPID Srv-Domain   VLAN SRR   Tx-I Rx-I
--------------------------------------------------------------------------
88   cust         30   N/A   Y   999  provid       20   L      1    1
```

16.2.7 配置 双端 LM

I. 拓扑

请参考 1.3.1.

II. 配置

请参考 1.3.2 中关于 MD/MA/MEP 的配置。

在 Switch1 和 Switch4 上使能双端 lm 功能。

Switch 1

Switch1# configure terminal	进入全局配置模式
Switch1(config)# ethernet cfm lm enable dual-ended domain cust vlan 30 mepid 66 all-cos cache-size 10	配置双端 lm 功能
Switch1(config)# end	退出配置模式

Switch 4

Switch4# configure terminal	进入全局配置模式
Switch4(config)# ethernet cfm lm enable dual-ended domain cust vlan 30 mepid 99 all-cos	配置双端 lm 功能

cache-size 10	
Switch4(config)# end	退出配置模式

I. 命令验证

以下命令用来显示 Switch1 上 lm 的结果：

```
Switch1# show ethernet cfm lm domain cust vlan 30 mepid 66
DOMAIN     : cust
VLAN       : 30
MEPID      : 66
Start Time : 2013/07/16 1:36:56
End Time   : 2013/07/16 1:37:07
Notes      : 1. When the difference of Tx is less than the difference of Rx,
                the node is invalid, loss and loss ratio should be "-";
             2. When loc is reported for mep, the loss should be "-" and loss
                ratio should be 100%;
             3. When calculate average loss and loss ratio, invalid or loc nodes
                will be excluded;
Latest dual-ended loss statistics:
--------------------------------------------------------------------------------
Index Cos Local-loss Local-loss ratio Remote-loss Remote-loss ratio Time
--------------------------------------------------------------------------------
1     all          0        000.0000%           0         000.0000% 01:36:57
2     all          0        000.0000%           0         000.0000% 01:36:58
3     all          0        000.0000%           0         000.0000% 01:36:59
4     all          0        000.0000%           0         000.0000% 01:37:00
5     all          0        000.0000%           0         000.0000% 01:37:01
6     all          0        000.0000%           0         000.0000% 01:37:02
7     all          0        000.0000%           0         000.0000% 01:37:03
8     all          0        000.0000%           0         000.0000% 01:37:04
9     all          0        000.0000%           0         000.0000% 01:37:05
10    all          0        000.0000%           0         000.0000% 01:37:07
--------------------------------------------------------------------------------
Maximum Local-loss  : 0          Maximum Local-loss Ratio  : 000.0000%
Minimum Local-loss  : 0          Minimum Local-loss Ratio  : 000.0000%
Average Local-loss  : 0          Average Local-loss Ratio  : 000.0000%
Maximum Remote-loss : 0          Maximum Remote-loss Ratio : 000.0000%
Minimum Remote-loss : 0          Minimum Remote-loss Ratio : 000.0000%
Average Remote-loss : 0          Average Remote-loss Ratio : 000.0000%
```

以下命令用来显示 Switch4 上 lm 的结果：

```
Switch4# show ethernet cfm lm domain cust vlan 30 mepid 99
DOMAIN     : cust
VLAN       : 30
MEPID      : 99
Start Time : 2013/07/16 1:37:11
End Time   : 2013/07/16 1:37:22
Notes      : 1. When the difference of Tx is less than the difference of Rx,
                the node is invalid, loss and loss ratio should be "-";
```

```
            2. When loc is reported for mep, the loss should be "-" and loss
               ratio should be 100%;
            3. When calculate average loss and loss ratio, invalid or loc nodes
               will be excluded;
Latest dual-ended loss statistics:
--------------------------------------------------------------------------------
Index Cos Local-loss Local-loss ratio Remote-loss Remote-loss ratio Time
--------------------------------------------------------------------------------
1     all          0        000.0000%           0         000.0000% 01:37:12
2     all          0        000.0000%           0         000.0000% 01:37:13
3     all          0        000.0000%           0         000.0000% 01:37:14
4     all          0        000.0000%           0         000.0000% 01:37:16
5     all          0        000.0000%           0         000.0000% 01:37:17
6     all          0        000.0000%           0         000.0000% 01:37:18
7     all          0        000.0000%           0         000.0000% 01:37:19
8     all          0        000.0000%           0         000.0000% 01:37:20
9     all          0        000.0000%           0         000.0000% 01:37:21
10    all          0        000.0000%           0         000.0000% 01:37:22
--------------------------------------------------------------------------------
Maximum Local-loss  : 0          Maximum Local-loss Ratio  : 000.0000%
Minimum Local-loss  : 0          Minimum Local-loss Ratio  : 000.0000%
Average Local-loss  : 0          Average Local-loss Ratio  : 000.0000%
Maximum Remote-loss : 0          Maximum Remote-loss Ratio : 000.0000%
Minimum Remote-loss : 0          Minimum Remote-loss Ratio : 000.0000%
Average Remote-loss : 0          Average Remote-loss Ratio : 000.0000%
```

16.2.8 配置单端 LM

I. 拓扑

请参考 1.3.1.

II. 配置

请参考 1.3.2 中关于 MD/MA/MEP 的配置。

在 Switch1 和 Switch4 上使能单端 lm 功能。

Switch 1

Switch1# configure terminal	进入全局配置模式
Switch1(config)# ethernet cfm lm enable single-ended domain cust vlan 30 mepid 66 all-cos	配置单端 lm 功能
Switch1(config)# end	退出配置模式

Switch 4

Switch4# configure terminal	进入全局配置模式

Switch4(config)# ethernet cfm lm enable single-ended domain cust vlan 30 mepid 99 all-cos	配置单端 lm 功能
Switch4(config)# end	退出配置模式

I. 命令验证

以下命令用于从 Switch1 上发送 LMM 消息，并显示 LM 的结果:

```
Switch1#  ethernet cfm lm single-ended domain cust vlan 30 rmepid 99 mepid 66 count
10
DOMAIN     : cust
VLAN       : 30
MEPID      : 66
Start Time : 2013/07/16 1:39:38
End Time   : 2013/07/16 1:39:38
Notes      : 1. When the difference of Tx is less than the difference of Rx,
                the node is invalid, loss and loss ratio should be "-";
             2. When loc is reported for mep, the loss should be "-" and loss
                ratio should be 100%;
             3. When calculate average loss and loss ratio, invalid or loc nodes
                will be excluded;
Latest single-ended loss statistics:
--------------------------------------------------------------------------------
Index Cos Local-loss Local-loss ratio Remote-loss Remote-loss ratio
--------------------------------------------------------------------------------
1     all          0        000.0000%           0         000.0000%
2     all          0        000.0000%           0         000.0000%
3     all          0        000.0000%           0         000.0000%
4     all          0        000.0000%           0         000.0000%
5     all          0        000.0000%           0         000.0000%
6     all          0        000.0000%           0         000.0000%
7     all          0        000.0000%           0         000.0000%
8     all          0        000.0000%           0         000.0000%
9     all          0        000.0000%           0         000.0000%
--------------------------------------------------------------------------------
Maximum Local-loss  : 0          Maximum Local-loss Ratio  : 000.0000%
Minimum Local-loss  : 0          Minimum Local-loss Ratio  : 000.0000%
Average Local-loss  : 0          Average Local-loss Ratio  : 000.0000%
Maximum Remote-loss : 0          Maximum Remote-loss Ratio : 000.0000%
Minimum Remote-loss : 0          Minimum Remote-loss Ratio : 000.0000%
Average Remote-loss : 0          Average Remote-loss Ratio : 000.0000%
```

16.2.9 配置 Test

I. 拓扑

请参考 CC Topology.

II. 配置

请参考 CC 中关于 MD/MA/MEP 的配置。

在 Switch1 和 Switch4 上使能 Test 功能。

Switch 1

Switch1# configure terminal	进入全局配置模式
Switch1(config)# ethernet cfm tst transmission enable domain cust vlan 30 mep 66 tx-mode continuous pattern-type random packet-size 64	配置 Test 发送功能
Switch1(config)# end	退出配置模式

Switch 4

Switch4# configure terminal	进入全局配置模式
Switch4(config)# ethernet cfm tst reception enable domain cust vlan 30 mep 99	配置 Test 接收功能
Switch4(config)# end	退出配置模式

I. 命令验证

以下命令用于从 Switch1 上发送 Test 消息:

```
Switch1#  ethernet cfm tst start rate 1000 time second 1
```

以下命令用于在 Switch1 上显示 Test 的信息:

```
Switch1#  show ethernet cfm tst
DOMAIN           : cust
VLAN             : 30
MEPID            : 66
Transmission     : Enabled
Reception        : Disabled
Status           : Non-Running
Start Time       : 06:32:48
Predict End Time : 06:33:18
Actual End Time  : 06:33:18
Packet Type      : TST
Rate             : 1000 mbps
Packet Size      : 64 bytes
Tx Number        : 29
Tx Bytes         : 1856
Rx Number        : 0
Rx Bytes         : 0
```

以下命令用于在 Switch4 上显示 Test 的信息:

```
Switch4#  show ethernet cfm tst
DOMAIN           : cust
VLAN             : 30
MEPID            : 99
Transmission     : Disabled
Reception        : Enabled
Status           : Non-Running
Start Time       : null
End Time         : null
Packet Type      : null
Rate             : null
Packet Size      : null
Tx Number        : 0
Tx Bytes         : 0
Rx Number        : 29
Rx Bytes         : 1856
```

16.3 CPU Traffic 配置

16.3.1 简介

本章描述如何配置 CPU 流量限制并查看 CPU 流量。

CPU 流量限制是一种很有用的保护 CPU 的机制，通过对进入 CPU 的报文的流量进行限制实现。

CPU 流量限制包含两个级别的 CPU 保护措施：

其一，限制各个进入 CPU 的 reason 的流量。在芯片中，是通过配置这个 reason 对应的 queue shaping 实现的。

其二，限制所有进入 CPU 的报文的流量。芯片中是通过配置 CPU 端口上的 shaping 实现的。

注：这里提到的 reason，意思是各种协议报文的类型，比如 bgp、ospf、rip 等分别是不同的 reason。

下表列出了各种 reason 和对应的描述：

Reason	描述
arp	arp 协议报文
bpdu	bpdu 协议报文(包括 STP, RSTP, MSTP)
dhcp	dhcp 协议报文
eapol	dot1x 协议报文
erps	erps 协议报文

Reason	描述
fwd-to-cpu	转发到 CPU 的报文
icmp-redirect	Icmp 重定向
igmp	IGMP 或者 IGMP Snooping 报文
ip-option	带有可选字段的 IP 报文
ipda	ipda 协议报文
ldp	ldp 协议报文
macsa-mismatch	与一个端口 security entry 不匹配时的学习报文
mcast-rpf-fail	组播报文 RPF 检查失败
mld	mld 协议报文
mpls-ttl-fail	ttl 失效 mpls 报文
ip-mtu-fail	需要分片的报文
ospf	ospf 协议报文
pim	pim 协议报文
port-security-discard	端口 security entry 学满时的学习报文
rip	rip 协议报文
sflow-egress	在出口方向 sflow 的采样报文
sflow-ingress	在入口方向 sflow 的采样报文
slow-protocol	slow 协议报文.(包括 EFM, LACP，SYNCE) ttl 失效的组播报文
smart-link	Smart link 协议报文
ucast-ttl-fail	ttl 失效的单播 IP 报文
udld	udld 协议报文
vlan-security-discard	vlan 内学习的 mac 达到限制是的学习报文
vrrp	vrrp 协议报文
bfd-learning	BFD 学习报文

16.3.2 术语

PDU 英文全称 Protocol Data Unit，协议数据单元

16.3.3 缺省配置

默认速率和优先级配置如下：

Reason	rate(pps)	class	reason	rate(pps)	class
arp	640	1	mpls-ttl-fail	64	0
bpdu	64	3	ip-mtu-fail	64	0
dhcp	128	0	ospf	256	1
eapol	128	0	pim	128	1
erps	128	2	port-security-discard	128	0
fwd-to-cpu	64	0	rip	64	1
icmp-redirect	128	0	sflow-egress	128	0
igmp	128	2	sflow-ingress	128	0
ip-option	512	0	slow-protocol	128	1
ipda	1024	0	smart-link	128	2
ldp	512	1	ucast-ttl-fail	64	0
macsa-mismatch	128	0	udld	128	3
mcast-rpf-fail	128	1	vlan-security-discard	128	0
mld	128	2	vrrp	512	1
bfd-learning	128	1			

16.3.4 CPU Traffic 配置

I. 总限速配置

Switch# configure terminal	进入全局配置模式
Switch(config)# cpu-traffic-limit total rate 3000	设置 CPU 流量总限速

II. 单个速率配置

Switch# configure terminal	进入全局配置模式
Switch(config)# cpu-traffic-limit reason rip rate 500	设置 RIP PDU 限速

III. 优先级类别配置

Switch# configure terminal	进入全局配置模式
Switch(config)# cpu-traffic-limit reason rip class 3	修改 RIP PDU 优先级类别

16.3.5 命令验证

使用命令“show cpu traffic-limit”查看配置结果，详细信息类似如下所示。

```
Switch# show cpu traffic-limit
reason                      rate (pps)   class
dot1x-mac-bypass            64            2
bpdu                        64            3
slow-protocol               128           1
eapol                       128           0
erps                        128           2
smart-link                  128           2
udld                        128           3
loopback-detection          64            3
arp                         256           1
dhcp                        128           0
rip                         500           3
ldp                         512           1
ospf                        256           1
pim                         128           1
vrrp                        512           1
ipda                        1024          0
icmp-redirect               128           0
mcast-rpf-fail              128           1
macsa-mismatch              128           0
port-security-discard       128           0
vlan-security-discard       128           0
mtu-dontfrag                64            0
mtu-frag                    64            0
ip-mtu-fail                 64            0
bfd-learning                128           1
ip-option                   512           0
ucast-ttl-fail              64            0
mpls-ttl-fail               64            0
igmp                        128           2
sflow-ingress               128           0
sflow-egress                128           0
fwd-to-cpu                  64            0
l2protocol-tunnel           1024          0

Total rate:     3000 (pps)
```

使用命令“show cpu traffic-statistics receive all”查看流量统计，详细信息类似如下所示。

```
Switch# show cpu traffic-statistics receive all
```

```
statistics rate time is 5 second(s)
reason            count(packets)    rate(pps)
dot1x-mac-bypass       0                 0
bpdu                   0                 0
slow-protocol          0                 0
eapol                  0                 0
erps                   0                 0
smart-link             0                 0
udld                   0                 0
loopback-detection     0                 0
arp                    0                 0
dhcp                   0                 0
rip                    0                 0
ldp                    0                 0
ospf                   0                 0
pim                    0                 0
bgp                    0                 0
vrrp                   0                 0
rsvp                   0                 0
ipda                   0                 0
icmp-redirect          0                 0
mcast-rpf-fail         0                 0
macsa-mismatch         0                 0
port-security-discard  0                 0
vlan-security-discard  0                 0
ip-mtu-fail            0                 0
bfd-learning           0                 0
ptp                    0                 0
ip-option              0                 0
tunnel-gre-keepalive   0                 0
ucast-ttl-fail         0                 0
mpls-ttl-fail          0                 0
igmp                   0                 0
sflow-ingress          0                 0
sflow-egress           0                 0
fwd-to-cpu             0                 0
l2protocol-tunnel      0                 0
mirror-to-cpu          0                 0
mpls-tp-pwoam          0                 0
other                  0                 0
Total                  0                 0
```

16.4 UDLD 配置

16.4.1 简介

UDLD (Unidirectional Link Detection，单向链路检测) 是一种可以检测和禁用单向链路的轻量级的协议。通过使用 UDLD 可以防止生成树等协议在单向链接时产生的异常情况。

16.4.2 拓扑

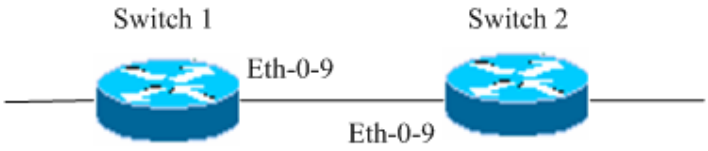


图16-3 UDLD 典型拓扑图

16.4.3 配置

Switch 1

在 Switch 1 的 eth-0-9 接口上使能 UDLD 协议。

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# udld port	在接口上使能 UDLD 协议
Switch(config-if)# exit	退出接口模式
Switch(config)# udld enable	全局使能 UDLD 协议
Switch(config)# udld message interval 10	设置 UDLD 消息发送间隔

Switch 2

在 Switch 2 的 eth-0-9 接口上使能 UDLD 协议。

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-9	进入接口模式
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# udld port	在接口上使能 UDLD 协议
Switch(config-if)# exit	退出接口模式
Switch(config)# udld enable	全局使能 UDLD 协议
Switch(config)# udld message interval 10	设置 UDLD 消息发送间隔

16.4.4 验证配置

Switch 1

```
Switch# show udld eth-0-9
Interface eth-0-9
---
UDLD mode         : normal
Operation state  : Bidirectional
Message interval : 10
Message timeout   : 3
  Neighbor 1
  ---
  Device ID       : 4c7b.8510.ab00
  Port ID         : eth-0-9
  Device Name     : Switch
  Message interval: 10
  Message timeout : 3
  Link Status     : bidirectional
  Expiration time : 29
```

Switch 2

```
Switch# show udld eth-0-9
Interface eth-0-9
---
UDLD mode       : normal
Operation state : Bidirectional
Message interval: 10
Message timeout : 3
  Neighbor 1
  ---
  Device ID       : 28bc.83db.8400
  Port ID         : eth-0-9
  Device Name     : Switch
  Message interval: 10
  Message timeout : 3
  Link Status     : bidirectional
  Expiration time : 23
```

16.5 Smart-Link 配置

16.5.1 简介

Smart Link，中文译为灵活链路，又称为备份链路，是一种为链路双上行提供可靠高效的备份和切换机制的解决方案，常用于双上行组网。相比 STP（Spanning Tree Protocol, 生成树协议），Smart Link 技术能够提供更快速的收敛性能，相比 ERPS，Smart Link 技术提供了更简洁的配置使用方式。

该功能还能提供链路负载均衡的功能。

16.5.2 拓扑

下面是一个 Smart-link 的典型配置，交换机 1 和 2 配置 Smart-link 组；交换机 3、4 和 5 配置 Smart-link 的报文接收。

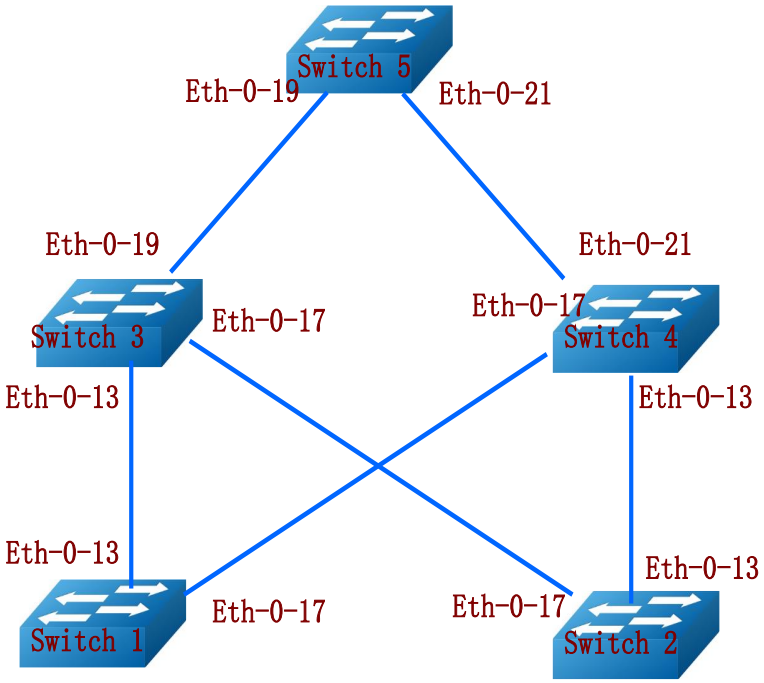


图16-4 Smart-Link Typical Topology

16.5.3 配置

下面的示例给出了 Smart Link 链路双上行保护的配置，拓扑图示图 16-4。

注意事项：

- Smart Link 组的控制 vlan 和保护 vlan 必须事先在 vlan database 创建好。
- Smart Link 组的端口必须把 STP 关闭。
- Smart Link 组的保护实例必须事先在 MSTP 模块先创建好。

给每台交换机配置 VLAN 1-300，MSTP instance1-3。

Switch 1 配置

Switch1# configure terminal	进入配置模式
Switch1(config)# vlan database	进入 VLAN 模式
Switch1(config- vlan)# vlan 2-20	创建 VALN2 到 20
Switch1(config- vlan)# exit	退出 VLAN 模式

Switch1(config)# spanning-tree mode mstp	配置 STP 的模式
Switch1(config)# spanning-tree mst configuration	进入 MSTP 配置模式
Switch1(config-mst)# instance 1 vlan 1	配置 MSTP 的实例 1 关联 VLAN 1
Switch1(config-mst)# instance 2 vlan 2	配置 MSTP 的实例 2 关联 VLAN 2
Switch1(config-mst)# instance 3 vlan 3	配置 MSTP 的实例 3 关联 VLAN 3
Switch1(config-mst)# exit	退出 MSTP 配置模式
Switch1(config)# interface eth-0-13	进入端口 13
Switch1(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch1(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch1(config-if)# spanning-tree port disable	关口端口上的 STP 功能
Switch1(config-if)# no shutdown	打开接口
Switch1(config-if)# exit	退出接口模式
Switch1(config)# interface eth-0-17	进入端口 17
Switch1(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch1(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch1(config-if)# spanning-tree port disable	关口端口上的 STP 功能
Switch1(config-if)# no shutdown	打开接口
Switch1(config-if)# exit	退出接口模式
Switch1(config)# smart-link group 1	创建组 1
Switch1(config-smlk-group)# interface eth-0-13 master	指定接口为 master 端口
Switch1(config-smlk-group)# interface eth-0-17 slave	指定接口为 slave 端口
Switch1(config-smlk-group)# protected mstp instance 1	指定保护的 MSTP Instance
Switch1(config-smlk-group)# protected mstp instance 2	指定保护的 MSTP Instance
Switch1(config-smlk-group)# protected mstp instance 3	指定保护的 MSTP Instance

Switch1(config-smlk-group)# load-balance instance 3	使能负载均衡的 Instance
Switch1(config-smlk-group)# restore time 40	设置自动倒换等待时间,范围为 30s-1200s
Switch1(config-smlk-group)# restore enable	启用自动倒换的功能
Switch1(config-smlk-group)# flush send control-vlan 10 password simple test	设置控制 VLAN 并指定 Smart-lingk 接收端的密码
Switch1(config-smlk-group)# group enable	启用 Smart-link 组
Switch1(config-smlk-group)# end	退出组模式

Switch 2 配置同 Switch 1。

Switch 3 配置

Switch3# configure terminal	进入配置模式
Switch3(config)# interface eth-0-13	进入端口 13
Switch3(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch3(config-if)# no shutdown	打开接口
Switch3(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch3(config-if)# smart-link flush receive control-vlan 10 password simple test	设置控制 VLAN 并指定 Smart-lingk 接收端的密码
Switch3(config-if)# exit	退出接口模式
Switch3(config)# interface eth-0-17	进入端口 17
Switch3(config-if)# no shutdown	打开接口
Switch3(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch3(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch3(config-if)# smart-link flush receive control-vlan 10 password simple test	设置控制 VLAN 并指定 Smart-lingk 接收端的密码
Switch3 (config-if)# exit	退出接口模式

NOTE

Switch 4 配置同 Switch 3。

Switch 5 配置

Switch5# configure terminal	进入配置模式
Switch5(config)# interface eth-0-19	进入端口 19
Switch5(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch5(config-if)# no shutdown	打开端口
Switch5(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch5(config-if)# smart-link flush receive control-vlan 10 password simple test	设置控制 VLAN 并指定 Smart-lingk 接收端的密码
Switch5(config-if)# exit	退出接口模式
Switch5(config)# interface eth-0-21	进入端口 21
Switch5(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch5(config-if)# no shutdown	打开端口
Switch5(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
smart-link flush receive control-vlan 10 password simple test	设置控制 VLAN 并指定 Smart-lingk 接收端的密码
Switch5(config-if)# exit	退出接口模式
Switch5(config)# no smart-link relay enable	取消 relay 功能

16.5.4 命令验证

Switch 1

Switch1# show smart-link group 1

```
Smart-link group 1 information:
 The smart-link group was enabled.
 ============================================================
 Auto-restore:
   state       time            count    Last-time
   enabled      40               0      N/A
 ============================================================
 Protected instance: 1  2  3
```

```
Load balance instance: 3
Flush sender , Control-vlan ID: 10    Password:test
============================================================
INTERFACE:
Role   Member    DownCount Last-Down-Time  FlushCount Last-Flush-Time
MASTER eth-0-13    0       N/A                   0     N/A
SLAVE  eth-0-17    0       N/A                   0     N/A
============================================================
Instance states in the member interfaces:
  A - ACTIVE ,   B -BLOCK  , D-The interface is link-down
Map-instance-ID   MASTER(eth-0-13)    SLAVE(eth-0-17)
      1              A                  B
      2              A                  B
      3              B                  A
```

Switch 2

Switch2# show smart-link group 1

```
Smart-link group 1 information:
The smart-link group was enabled.
============================================================
Auto-restore:
  state       time            count    Last-time
  enabled      40               0      N/A
============================================================
Protected instance: 1  2  3
Load balance instance: 3
Flush sender , Control-vlan ID: 10    Password:test
============================================================
INTERFACE:
Role   Member    DownCount Last-Down-Time  FlushCount Last-Flush-Time
MASTER eth-0-13    0       N/A                   0     N/A
SLAVE  eth-0-17    0       N/A                   0     N/A
============================================================
Instance states in the member interfaces:
  A - ACTIVE ,   B -BLOCK  , D-The interface is link-down
Map-instance-ID   MASTER(eth-0-13)    SLAVE(eth-0-17)
      1              A                  B
      2              A                  B
      3              B                  A
```

Switch 3

Switch3# show smart-link

```
Relay smart-link flush packet is enabled
Smart-link flush receiver interface:
   eth-0-13   control-vlan:10   password:test
   eth-0-17   control-vlan:10   password:test
Smart-link received flush packet number:0
Smart-link processed flush packet number:0
Smart link Group Number is 0.
```

Switch 4

Switch4# show smart-link

```
 Relay smart-link flush packet is enabled
 Smart-link flush receiver interface:
    eth-0-13   control-vlan:10   password:test
    eth-0-17   control-vlan:10   password:test
Smart-link received flush packet number:0
Smart-link processed flush packet number:0
Smart link Group Number is 0.
```

Switch 5

Switch5# show smart-link

```
 Relay smart-link flush packet is disabled
 Smart-link flush receiver interface:
    eth-0-21   control-vlan:10   password: test
    eth-0-19   control-vlan:10   password:test
Smart-link received flush packet number:0
Smart-link processed flush packet number:0
 Smart link Group Number is 0.
```

16.6 Multi-Link 配置

16.6.1 简介

Multi-Link，中文译为多链路，又称为多备份链路，是一种为链路多上行提供可靠高效的备份和切换机制的解决方案。该计划功能和 Smart Link 类似，备份的链路从一条扩充为多条，最多可以有 4 个成员。

该功能还能提供链路负载均衡的功能。

16.6.2 拓扑

下面是一个 Mulit-link 的典型配置，交换机 1 配置 Multi-link 组；交换机 2、3、4 和 5 配置 Multi-link 的报文接收。

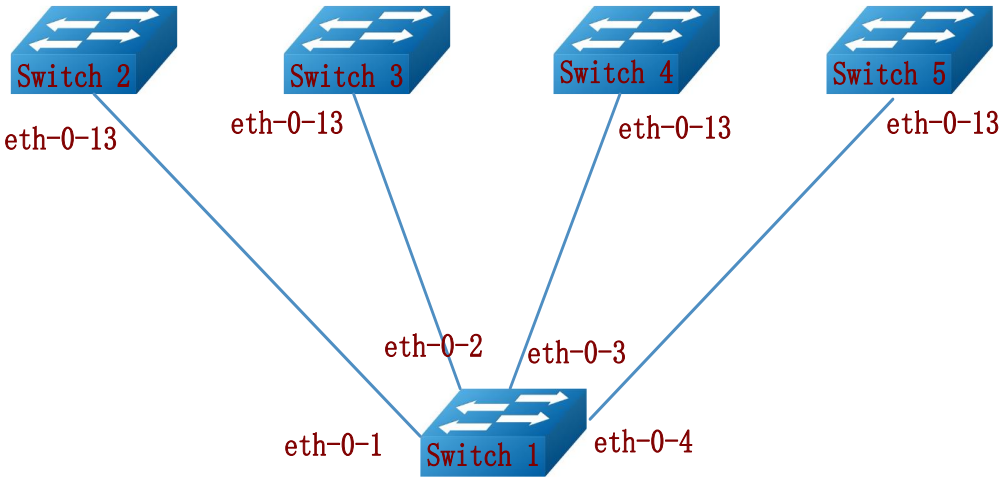


图16-5 Multi-Link Typical Topology

16.6.3 配置

下面的示例给出了 Multi-Link 链路多上行保护的配置，拓扑图示图 16-4。

注意事项：

- Multi-Link 组的控制 vlan 和保护 vlan 必须事先在 vlan database 创建好。
- Multi-Link 组的端口必须把 STP 关闭。
- Multi-Link 组的保护实例必须事先在 MSTP 模块先创建好。

给每台交换机配置 VLAN 2-10，MSTP instance1-4。

Switch 1 配置

Switch1# configure terminal	进入配置模式
Switch1(config)# vlan database	进入 VLAN 模式
Switch1(config- vlan)# vlan 2-10	创建 VALN2 到 10
Switch1(config- vlan)# exit	退出 VLAN 模式
Switch1(config)# spanning-tree mode mstp	配置 STP 的模式
Switch1(config)# spanning-tree mst configuration	进入 MSTP 配置模式

Switch1(config-mst)# instance 1 vlan 1	配置 MSTP 的实例 1 关联 VLAN 1
Switch1(config-mst)# instance 2 vlan 2	配置 MSTP 的实例 2 关联 VLAN 2
Switch1(config-mst)# instance 3 vlan 3	配置 MSTP 的实例 3 关联 VLAN 3
Switch1(config-mst)# instance 4 vlan 4-10	配置 MSTP 的实例 4 关联 VLAN 4
Switch1(config-mst)# exit	退出 MSTP 配置模式
Switch1(config)# interface range eth-0-1 - 4	进入端口 1-4
Switch1(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch1(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch1(config-if)# spanning-tree port disable	关口端口上的 STP 功能
Switch1(config-if)# no shutdown	打开接口
Switch1(config-if)# exit	退出接口模式
Switch1(config)# multi-link group 1	创建组 1
Switch1(config-multilk-group)# interface eth-0-1 priority 1	指定接口 1 优先级为 1
Switch1(config-multilk-group)# interface eth-0-2 priority 2	指定接口 2 优先级为 2
Switch1(config-multilk-group)# interface eth-0-3 priority 3	指定接口 3 优先级为 3
Switch1(config-multilk-group)# interface eth-0-4 priority 4	指定接口 4 优先级为 4
Switch1(config-multilk-group)# protected mstp instance 1	指定保护的 MSTP Instance
Switch1(config-multilk-group)# protected mstp instance 2	指定保护的 MSTP Instance
Switch1(config-multilk-group)# protected mstp instance 3	指定保护的 MSTP Instance
Switch1(config-multilk-group)# protected mstp instance 4	指定保护的 MSTP Instance
Switch1(config-multilk-group)# load-balance instance 2 priority 2	使能负载均衡的 Instance
Switch1(config-multilk-group)# load-balance instance 3 priority 3	使能负载均衡的 Instance
Switch1(config-multilk-group)# load-balance instance 4 priority 4	使能负载均衡的 Instance

Switch1(config-multilk-group)# restore time 40	设置自动倒换等待时间,范围为 30s-1200s
Switch1(config-multilk-group)# restore enable	启用自动倒换的功能
Switch1(config-multilk-group)# flush send control-vlan 10 password simple test	设置控制 VLAN 并指定 Multi-lingk 接收端的密码
Switch1(config-multilk-group)# group enable	启用 Multi-link 组
Switch1(config-multilk-group)# end	退出组模式

Switch 2 配置

Switch2# configure terminal	进入配置模式
Switch2(config)# interface eth-0-13	进入端口 13
Switch2(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch2(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch2(config-if)# no shutdown	打开接口
Switch2(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch2(config-if)# multi-link flush receive control-vlan 10 password simple test	设置控制 VLAN 并指定 Multi-lingk 接收端的密码
Switch2(config-if)# exit	退出接口模式

NOTE

Switch 3-5 配置同 Switch 2。

16.6.4 命令验证

Switch 1

Switch1# show multi-link group 1

```
Multi-link group 1 information:
 The multi-link group was enabled.
 ============================================================
 Auto-restore:
   state       time            count    Last-time
```

```
  enabled      40              0      N/A
============================================================
Protected instance: 1  2  3  4
Load balance instance: 2(to P2)  3(to P3)  4(to P4)
Flush sender , Control-vlan ID: 10    Password:test
============================================================
INTERFACE:
Role   Member    DownCount Last-Down-Time      FlushCount Last-Flush-Time
PRI1   eth-0-1   0         N/A                1          2016/09/05,07:13:24
PRI2   eth-0-2   0         N/A                1          2016/09/05,07:13:24
PRI3   eth-0-3   0         N/A                1          2016/09/05,07:13:24
PRI4   eth-0-4   0         N/A                1          2016/09/05,07:13:24
============================================================
Instance states in the member interfaces:
  A - ACTIVE ,   B -BLOCK  , D-The interface is link-down
Map-instance-ID  P1(eth-0-1 )  P2(eth-0-2 )  P3(eth-0-3 )  P4(eth-0-4 )
1                A             B             B             B
2                B             A             B             B
3                B             B             A             B
4                B             B             B             A
```

Switch 2

Switch2# show multi-link

```
Relay multi-link flush packet is enabled
Multi-link flush receiver interface:
   eth-0-13   control-vlan:10   password:test
Multi-link received flush packet number:0
Multi-link processed flush packet number:0
Multi-link tcn is disabled
Multi-link tcn query count    :2
Multi-link tcn query interval :10
Multi-link Group Number is 0.
```

16.7 Multi-Link 增强配置

16.7.1 简介

Multi-Link，中文译为多链路，又称为多备份链路，是一种为链路多上行提供可靠高效的备份和切换机制的解决方案。该计划功能和 Smart Link 类似，备份的链路从一条扩充为多条，最多可以有 4 个成员。

当分布在不同交换机的两组 multi-link 作相互链路备份时，会由于一方的 multi-link 成员的保护实例被 block 住而无法进行链路相互备份。

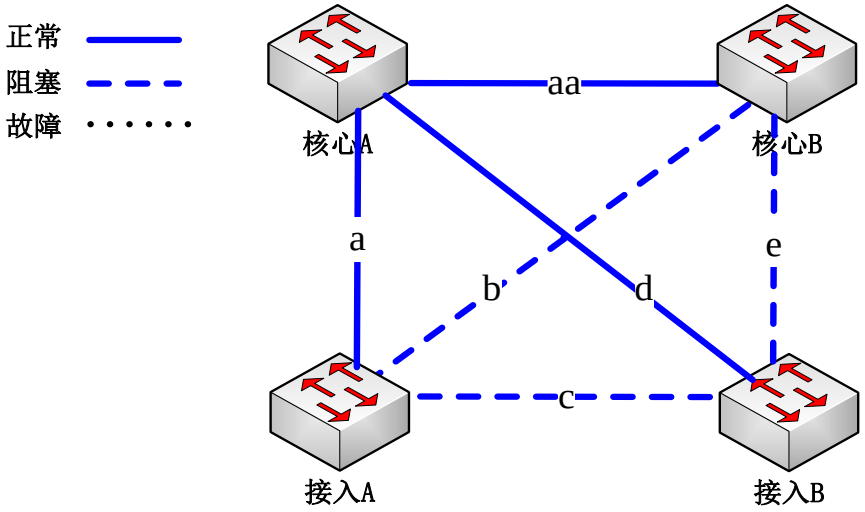
比如如下用户场景：

核心交换机 A、核心交换机 B、接入交换机 A、接入交换机 B 形成全网状拓扑。

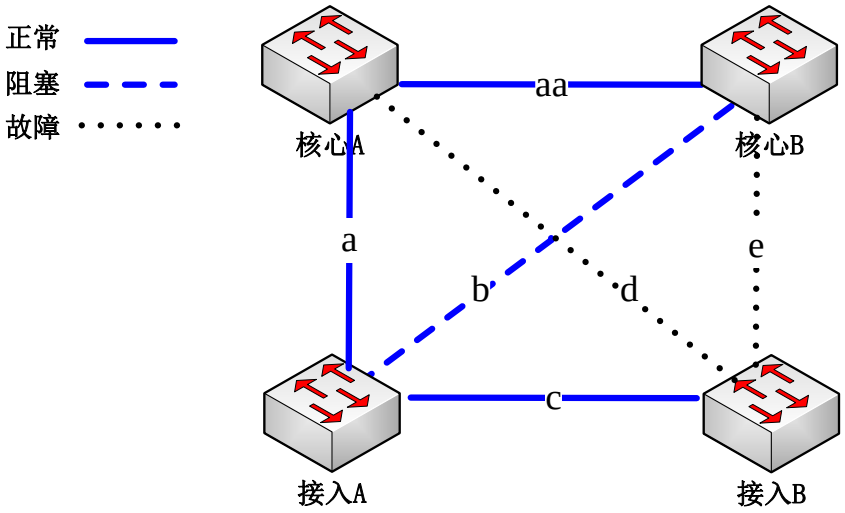
接入交换机 A 配置 multi-link 协议，链路 a、b、c 优先级分别为 1、2、3；

接入交换机 B 配置 multi-link 协议，链路 d、e 优先级分别为 1、2；

正常情况下，链路 b、c、e 处于 block 状态，链路 a、d 处于 active 状态，如下图所示：



当接入交换机 B 链路 d、e 全部断掉后，仅余下链路 c 与接入交换机 A 连接，如下图所示：



此时，接入交换机 A 链路 a 处于 active 状态，接入交换机对应链路 c 的端口处于 block 状态，接入交换机 B 处于孤岛状态。

16.7.2 拓扑

下面是一个 Mulit-link 的典型配置，交换机 1, 2 均配置 Multi-link 组。Switch 1 multi-link 组里面配有三个成员，且优先级最低的成员为 multi-link 增强的接受口。Switch 2 multi-link 组里面配有两个成员，此外还配有 multi-link 增强的发送口。

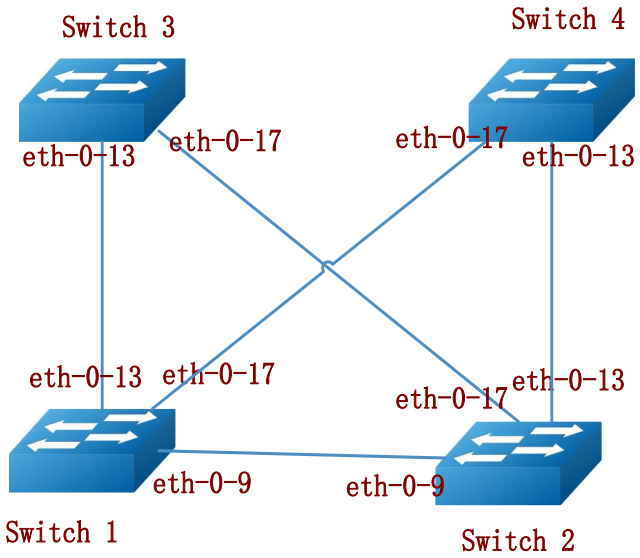


图16-6 Multilink-enhance Typical Topology

16.7.3 配置

下面的示例给出了 Multi-Link 链路多上行保护的配置，拓扑图示图 16-4。

注意事项：

- Multi-Link 组的控制 vlan 和保护 vlan 必须事先在 vlan database 创建好。
- Multi-Link 组的端口必须把 STP 关闭。
- Multi-Link 组的保护实例必须事先在 MSTP 模块先创建好。
- Multi-link 组中需要先配 flush send 的 control vlan 和 password，然后才能配置 multilink 增强

给每台交换机配置 VLAN 10, 20, 30, 40，MSTP instance1,2。

Switch 1 配置

Switch1# configure terminal	进入配置模式
Switch1(config)# vlan database	进入 VLAN 模式
Switch1(config- vlan)# vlan 10	创建 VALN 10
Switch1(config- vlan)# vlan 20	创建 VALN 20
Switch1(config- vlan)# vlan 30	创建 VALN 30
Switch1(config- vlan)# vlan 40	创建 VALN 40

Switch1(config- vlan)# exit	退出 VLAN 模式
Switch1(config)# spanning-tree mode mstp	配置 STP 的模式
Switch1(config)# spanning-tree mst configuration	进入 MSTP 配置模式
Switch1(config-mst)# instance 1 vlan 10	配置 MSTP 的实例 1 关联 VLAN 10
Switch1(config-mst)# instance 1 vlan 30	配置 MSTP 的实例 1 关联 VLAN 30
Switch1(config-mst)# instance 2 vlan 20	配置 MSTP 的实例 2 关联 VLAN 20
Switch1(config-mst)# instance 2 vlan 40	配置 MSTP 的实例 2 关联 VLAN 40
Switch1(config-mst)# exit	退出 MSTP 配置模式
Switch1(config)# interface range eth-0-9	进入端口 9
Switch1(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch1(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch1(config-if)# spanning-tree port disable	关口端口上的 STP 功能
Switch1(config-if)# no shutdown	打开接口
Switch1(config-if)# exit	退出接口模式
Switch1(config)# interface range eth-0-13	进入端口 13
Switch1(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch1(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch1(config-if)# spanning-tree port disable	关口端口上的 STP 功能
Switch1(config-if)# no shutdown	打开接口
Switch1(config-if)# exit	退出接口模式
Switch1(config)# interface range eth-0-17	进入端口 17
Switch1(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch1(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch1(config-if)# spanning-tree port disable	关口端口上的 STP 功能
Switch1(config-if)# no shutdown	打开接口
Switch1(config-if)# exit	退出接口模式

Switch1(config)# multi-link group 1	创建组 1
Switch1(config-multilk-group)# interface eth-0-13 priority 1	指定接口 13 优先级为 1
Switch1(config-multilk-group)# interface eth-0-17 priority 2	指定接口 17 优先级为 2
Switch1(config-multilk-group)# interface eth-0-9 priority 3	指定接口 9 优先级为 3
Switch1(config-multilk-group)# protected mstp instance 1	指定保护的 MSTP Instance
Switch1(config-multilk-group)# protected mstp instance 2	指定保护的 MSTP Instance
Switch1(config-multilk-group)# flush send control-vlan 30 password simple a	设置控制 VLAN 并指定 Multi-lingk 发送端的密码
Switch1(config-multilk-group)# multilink-enhance receive control-vlan 10 password b interface eth-0-9	启用 eth-0-9 口接受 multilink-enhance 报文
Switch1(config-multilk-group)# group enable	启用 Multi-link 组
Switch1(config-multilk-group)# end	退出组模式

Switch 2 配置

Switch2# configure terminal	进入配置模式
Switch2(config)# vlan database	进入 VLAN 模式
Switch2(config- vlan)# vlan 10	创建 VALN 10
Switch2(config- vlan)# vlan 20	创建 VALN 20
Switch2(config- vlan)# exit	退出 VLAN 模式
Switch2(config)# spanning-tree mode mstp	配置 STP 的模式
Switch2(config)# spanning-tree mst configuration	进入 MSTP 配置模式
Switch2(config-mst)# instance 1 vlan 10	配置 MSTP 的实例 1 关联 VLAN 10
Switch2(config-mst)# instance 2 vlan 20	配置 MSTP 的实例 2 关联 VLAN 20
Switch2(config-mst)# exit	退出 MSTP 配置模式
Switch2(config)# interface eth-0-13	进入端口 13

Switch2(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch2(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch2(config-if)# no shutdown	打开接口
Switch2(config-if)# exit	退出接口模式
Switch2(config)# interface eth-0-17	进入端口 13
Switch2(config-if)# switchport mode trunk	配置端口为 trunk 口
Switch2(config-if)# switchport trunk allowed vlan all	配置端口允许所有 VLAN 通过
Switch2(config-if)# no shutdown	打开接口
Switch2(config-if)# exit	退出接口模式
Switch2(config)# multi-link group 1	创建组 1
Switch2(config-multilk-group)# interface eth-0-13 priority 1	指定接口 13 优先级为 1
Switch2(config-multilk-group)# interface eth-0-17 priority 2	指定接口 17 优先级为 2
Switch2(config-multilk-group)# protected mstp instance 1	指定保护的 MSTP Instance
Switch2(config-multilk-group)# protected mstp instance 2	指定保护的 MSTP Instance
Switch2(config-multilk-group)# flush send control-vlan 10 password simple b	设置控制 VLAN 并指定 Multi-lingk 发送端的密码
Switch2(config-multilk-group)# multilink-enhance interface eth-0-9	设置发送 multilink 增强报文的接口
Switch2(config-multilk-group)# group enable	启用 Multi-link 组
Switch2(config-multilk-group)# exit	退出组模式
Switch2(config)# interface eth-0-9	进入接口配置模式下
Switch2(config-if)# multi-link flush receive control-vlan 30 password simple a	配置该接口可以接受 flush 报文，control vlan id 和 passward 要和 swith1 配置 flush send 一致
Switch2(config-if)#end	退出接口模式

NOTE

Switch3-4 配置只需要配置接受 flush 报文的节奏即可。

16.7.4 命令验证

Switch 1

Switch1# show multi-link group 1

```
Multi-link group 1 information:
 The multi-link group was enabled.
 ============================================================
 Auto-restore:
   state       time            count     Last-time
   disabled    60              0         N/A
 ============================================================
 Protected instance: 1  2
 Load balance instance:
 Flush sender , Control-vlan ID: 30    Password: a
 ============================================================
 INTERFACE:
 Role   Member    DownCount Last-Down-Time       FlushCount Last-Flush-Time
 PRI1   eth-0-13  0         N/A                 5          2017/05/15,07:50:11
 PRI2   eth-0-17  0         N/A                 0          N/A
 PRI3   eth-0-9   1         2017/05/15,07:48:46 5          2017/05/15,07:50:11
 PRI4   N/A       0         N/A                 0          N/A
 ============================================================
 Instance states in the member interfaces:
   A-ACTIVE ,    B-BLOCK ,  A(E)-ENHANCE_ACTIVE    D-The interface is link-down
 Map-instance-ID  P1(eth-0-13)    P2(eth-0-17)    P3(eth-0-9)     P4(N/A)
 1                A               B               B               D
 2                A               B               B               D
Switch1# show multi-link
Relay multi-link flush packet is enabled
 Multi-link enhance receiver interface:
    eth-0-9    control-vlan:10   password:b
 Multi-link received flush packet number : 0
 Multi-link processed flush packet number: 0
 Multi-link received enhance packet number : 4
 Multi-link processed enhance packet number: 4
 Multi-link tcn is disabled
 Multi-link tcn query count    : 2
 Multi-link tcn query interval : 10
 Multi-link Group Number is 1.
  Group-ID    State       Pri-1      Pri-2      Pri-3      Pri-4
  1           enabled     eth-0-13   eth-0-17   eth-0-9    N/A
```

Switch 2

Switch2# show multi-link group1

```
Multi-link group 1 information:
```

```
 The multi-link group was enabled.
 ============================================================
 Auto-restore:
   state       time            count     Last-time
   disabled    60              0         N/A
 ============================================================
 Protected instance: 1  2
 Load balance instance:
 Flush sender , Control-vlan ID: 10    Password: b
 Multilk enhance interface: eth-0-9, Control-vlan ID: 10  Password: b
 ============================================================
 INTERFACE:
 Role    Member     DownCount Last-Down-Time       FlushCount Last-Flush-Time
 PRI1    eth-0-13   1         2017/05/15,07:49:15 0          N/A
 PRI2    eth-0-17   2         2017/05/15,07:50:03 3          2017/05/15,07:50:11
 PRI3    N/A        0         N/A                 0          N/A
 PRI4    N/A        0         N/A                 0          N/A
 ============================================================
ENHANCE INTERFACE:
 Role    Member     DownCount Last-Down-Time       EnhanceCount Last-SendEnhance-Ti
me
 M-En    eth-0-9    0         N/A                 0            N/A
 ============================================================
 Instance states in the member interfaces:
   A-ACTIVE ,    B-BLOCK ,  A(E)-ENHANCE_ACTIVE    D-The interface is link-down
 Map-instance-ID  P1(eth-0-13)    P2(eth-0-17)    P3(N/A)          P4(N/A)
 1                A               B               D                D
 2                A               B               D                D
Switch2# show multi-link
Relay multi-link flush packet is enabled
 Multi-link received flush packet number : 0
 Multi-link processed flush packet number: 0
 Multi-link received enhance packet number : 0
 Multi-link processed enhance packet number: 0
 Multi-link tcn is disabled
 Multi-link tcn query count    : 2
 Multi-link tcn query interval : 10
 Multi-link Group Number is 1.
  Group-ID    State       Pri-1      Pri-2      Pri-3      Pri-4
  1           enabled     eth-0-13   eth-0-17   N/A        N/A
```

16.8 Monitor-Link 配置

16.8.1 简介

Monitor Link 是对 Smart Link 进行补充而引入的端口联动方案，用于扩展 Smart Link 的链路备份的范围，通过监控上行链路对下行链路进行同步设置，达到上行链路故障迅速传达给下行设备，从而触发 Smart Link 的主备链路切换，防止长时间因上行链路故障而出现流量丢失。

16.8.2 拓扑

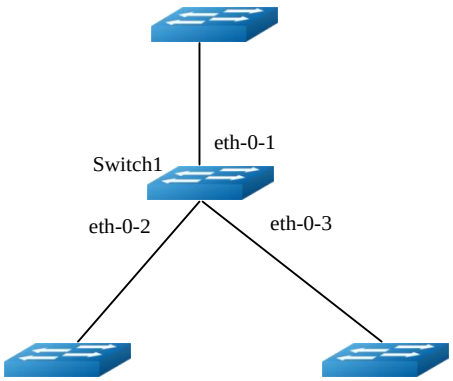


图16-7 配置 monitor link

16.8.3 配置

Switch1# configure terminal	进入配置模式
Switch1(config)# interface range eth-0-1 - 3	进入接口模式
Switch1(config-if-range)# no shutdown	打开端口
Switch1(config-if-range)# exit	退出接口模式
Switch1(config)# monitor-link group 1	创建 monitor link 组 1
Switch1(config-mtlk-group)# monitor-link uplink interface eth-0-1	将端口 eth-0-1 作为上联端口
Switch1(config-mtlk-group)# monitor-link downlink interface eth-0-2	将端口 eth-0-2 作为下联端口
Switch1(config-mtlk-group)# monitor-link downlink interface eth-0-3	将端口 eth-0-3 作为下联端口
Switch1(config-mtlk-group)# end	退出 monitor link 模式

16.8.4 Validation

Switch1# show monitor-link group

```
Group Id: 1
Monitor link status: UP
Role     Member   Last-up-time      Last-down-time          upcount   downcount
UpLk 1   eth-0-1  2011/07/15,02:07:31  2011/07/15,02:07:31     2         1
DwLk 1   eth-0-2  2011/07/15,02:07:34  2011/07/15,02:07:31     1         1
DwLk 2   eth-0-3  N/A                      N/A                        0         0
```

16.9 VRRP 配置

16.9.1 简介

通常，子网内的所有主机都设置一条相同的到网关的缺省路由。主机发出的所有目的地址不在本网段的报文将通过缺省路由发往网关，从而实现主机与外部网络的通信。当网关发生故障时，本网段内所有以网关为缺省路由的主机将中断与外部网络的通信。

缺省路由为用户的配置操作提供了方便，但是对缺省网关设备提出了很高的稳定性要求。增加出口网关是提高系统可靠性的常见方法，此时如何在多个出口之间进行选路就成为需要解决的问题。

VRRP（Virtual Router Redundancy Protocol，虚拟路由器冗余协议）可以解决以上问题。在具有多播或广播能力的局域网（如以太网）中，借助 VRRP 能在某台设备出现故障时仍然提供高可靠的缺省链路，而无需修改用户的配置信息。

VRRP 将局域网内的一组路由器（包括一个 Master 路由器和若干个 Backup 路由器）组成一个备份组，功能上相当于一台虚拟路由器。

VRRP 备份组具有以下特点：

局域网内的主机仅需要知道这个虚拟路由器的 IP 地址，并将其设置为缺省路由的下一跳地址。

网络内的主机通过这个虚拟路由器与外部网络进行通信。

备份组内的路由器根据一定的选举机制，分别承担网关的功能。当备份组内承担网关功能的路由器发生故障时，其余的路由器将取代它继续履行网关职责。

16.9.2 参考

VRRP 参考文档如下：

RFC 3768 (VRRP)：Knight, S., et.al "Virtual Router Redundancy Protocol (VRRP)

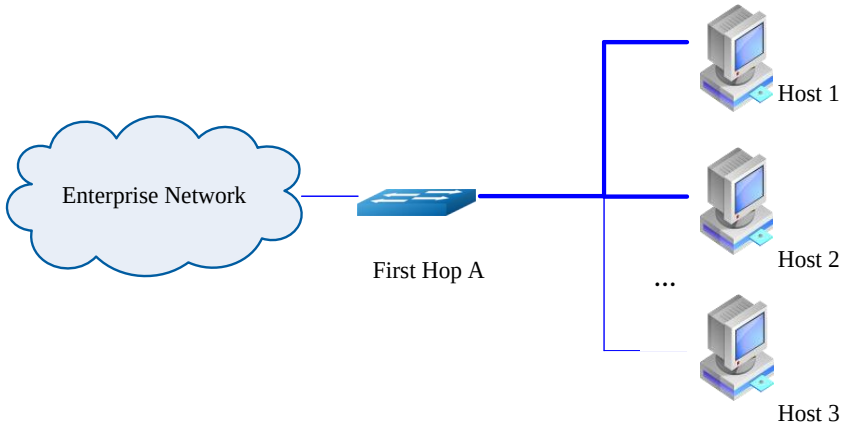
16.9.3 术语

- **Backup Router**：VRRP 备份路由器。当 Master 路由器转发失败的时候启用备份路由器。
- **Critical IP**：VRRP 路由器发送/接收一个特定的会话信息的 IP 地址。
- **IP Address Owner**：VRRP 路由器将虚拟路由器的 IP 地址作为真实的接口地址。当这台设备正常工作时，它会响应目的地址是虚拟 IP 地址的报文，如 ping、TCP 连接等。
- **Master Router**：拥有虚拟 IP 地址的路由器。此时它成为主机的默认网关，负责转发数据流。
- **Virtual IP**：虚拟路由器的 IP 地址，一个虚拟路由器可以有一个 IP 地址，由用户配置。

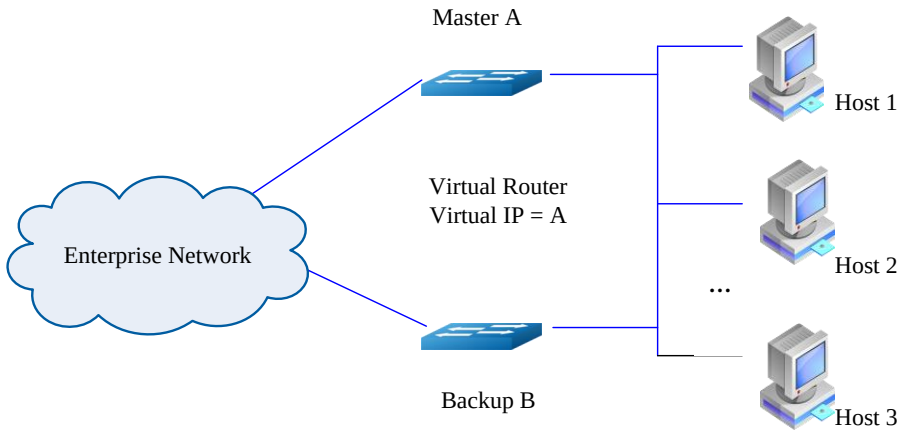
- **Virtual Router**：由 VRRP 管理的抽象设备，又称为 VRRP 备份组，被当作一个共享局域网内主机的缺省网关。它包括了一个虚拟路由器标识符和一个虚拟 IP 地址。
- **VRRP Router**：运行 VRRP 的设备，它可能属于一个或多个虚拟路由器。

16.9.4 VRRP Process

通常情况下，终端主机是通过将在同一个局域网内的路由器作为其第一个下一跳连接到企业网的。终端主机最常见的配置就是静态配置这个默认网关。这可以最大限度地减少配置和处理开销。此配置方法的主要问题是如果这第一跳路由器出问题，它会产生一个单点故障。.



虚拟路由器冗余协议试图通过引入一个虚拟路由器的概念来解决这个问题，它通常由在同一子网中的两个或两个以上的 VRRP 路由器组成。同时它还引入了一个虚拟 IP 地址的概念，终端主机使用这个 IP 作为它们的默认网关地址。只有主路由器负责转发数据包，在主路由器出现故障时，其他路由器（备份）中的一个代替主路由器负责转发。



上述配置概述可能不是非常有用，因为它的成本增加一倍，并且一台路由器在大部分时间都闲置。然而，我们可以创建两个虚拟路由器进行负载分担来避免这个问题。

16.9.5 配置 VRRP (一个虚拟路由器)

主备备份方式表示业务仅由 Master 路由器承担。当 Master 路由器出现故障时，才会从其他 Backup 路由器选举出一个接替工作。主备备份方式仅需要一个备份组，不同路由器在该备份组中拥有不同优先级，优先级最高的路由器将成为 Master 路由器。

下面的例子中，所有的终端主机将虚拟路由器 1 作为其默认网关。路由器 R1 和 R2 都运行了 VRRP 协议。 R1 配置为虚拟路由器 1（VRID = 1）的主路由器， R2 作为虚拟路由器 1 的备份路由器。如果 R1 出现问题，R2 将接管转发，并为主机提供不间断的服务。这样的配置只有一个虚拟路由器，R2 被闲置。

I. 拓扑

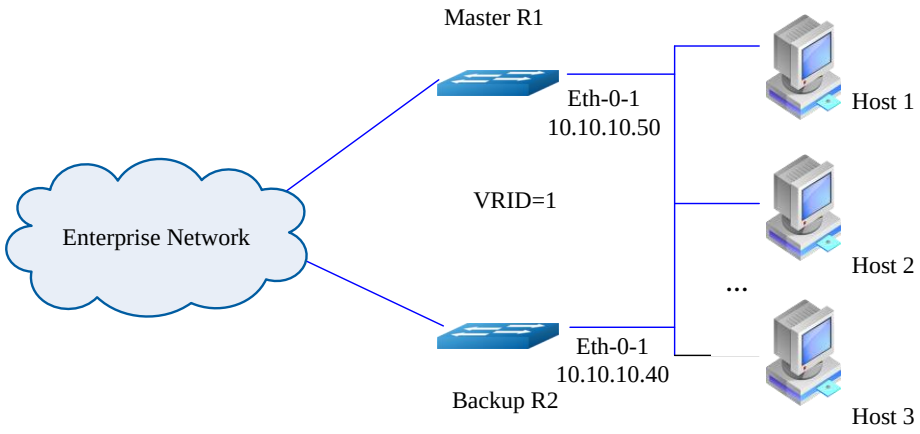


图16-8 单 VRRP 路由器

II. 配置

R1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入端口模式
Switch(config-if)# no switchport	设置端口为三层接口
Switch(config-if)# ip address 10.10.10.50/24	设置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router vrrp 1	创建虚拟路由器组 1
Switch(config-router)# virtual-ip 10.10.10.50	设置虚拟 IP 地址.
Switch(config-router)# interface eth-0-1	配置 VRRP 组的应用端口
Switch(config-router)# preempt-mode true	设置抢占模式
Switch(config-router)# advertisement-interval	配置通告时间间隔

5	
Switch (config-router)# bfd 10.10.10.40	配置 BFD 会话
Switch(config-router)# enable	使能 VRRP 组 1

R2

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入端口模式
Switch(config-if)# no switchport	设置端口为三层接口
Switch(config-if)# ip address 10.10.10.40/24	设置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router vrrp 1	创建 VRRP 虚拟路由器组 1
Switch(config-router)# virtual-ip 10.10.10.50	设置虚拟 IP 地址
Switch(config-router)# interface eth-0-1	配置 VRRP 组的应用端口
Switch(config-router)# priority 200	配置 VRRP 的优先级
Switch(config-router)# preempt-mode true	设置抢占模式
Switch(config-router)# advertisement-interval 5	配置通告时间间隔
Switch (config-router)# bfd 10.10.10.50	配置 BFD 会话
Switch(config-router)# enable	使能 VRRP 组 1

16.9.6 配置 VRRP (两个虚拟路由器)

在路由器的一个接口上可以创建多个备份组，使得该路由器可以在一个备份组中作为 Master 路由器，在其他的备份组中作为 Backup 路由器。

负载分担方式是指多台路由器同时承担业务，因此负载分担方式需要两个或者两个以上的备份组，每个备份组都包括一个 Master 路由器和若干个 Backup 路由器，各备份组的 Master 路由器可以各不相同。

下面的例子讲述如何使用两个虚拟路由器进行负载分担。R1 和 R2 各自转发不同的流量，他们之间互为备份，确保流量的负载均衡。

I. 拓扑

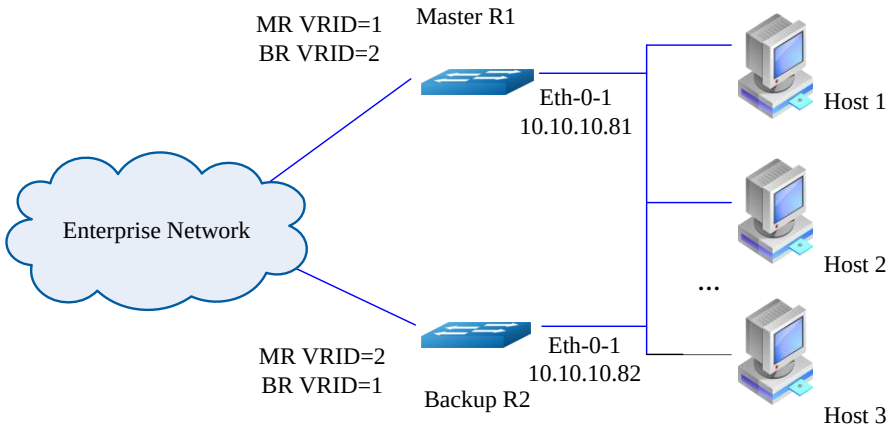


图16-9 Two Virtual Router

II. 配置

R1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置端口为三层接口
Switch(config-if)# ip address 10.10.10.81/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router vrrp 1	创建 VRRP 虚拟路由器组 1
Switch(config-router)# virtual-ip 10.10.10.81	设置虚拟 IP 地址
Switch(config-router)# interface eth-0-1	配置 VRRP 组的应用端口
Switch(config-router)# preempt-mode true	设置抢占模式
Switch(config-router)# advertisement-interval 5	配置通告时间间隔为 5 秒
Switch(config-router)# enable	使能 VRRP 组 1
Switch(config-router)# exit	退出路由模式
Switch(config)# router vrrp 2	创建 VRRP 虚拟路由器组 2
Switch(config-router)# virtual-ip 10.10.10.82	设置虚拟 IP 地址
Switch(config-router)# interface eth-0-1	配置 VRRP 组的应用端口
Switch(config-router)# priority 200	配置 VRRP 的优先级
Switch(config-router)# preempt-mode true	设置抢占模式

Switch(config-router)# advertisement-interval 5	配置通告时间间隔 5 秒
Switch (config-router)# bfd 10.10.10.82	配置 BFD 会话
Switch(config-router)# enable	使能 VRRP 组 2

R2

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置三层接口
Switch(config-if)# ip address 10.10.10.82/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router vrrp 1	创建 VRRP 虚拟路由器组 1
Switch(config-router)# virtual-ip 10.10.10.81	设置虚拟 IP 地址
Switch(config-router)# interface eth-0-1	配置 VRRP 组应用端口
Switch(config-router)# priority 200	设置 VRRP 的优先级
Switch(config-router)# preempt-mode true	设置抢占模式
Switch(config-router)# advertisement-interval 5	配置通告时间间隔 5 秒
Switch(config-router)# enable	使能 VRRP 组 1
Switch(config-router)# exit	退出路由模式
Switch(config)# router vrrp 2	创建 VRRP 虚拟路由器组 2
Switch(config-router)# virtual-ip 10.10.10.82	设置虚拟 IP 地址
Switch(config-router)# interface eth-0-1	配置 VRRP 组应用端口
Switch(config-router)# preempt-mode true	设置抢占模式
Switch(config-router)# advertisement-interval 5	配置通告时间间隔 5 秒
Switch (config-router)# bfd 10.10.10.81	配置 BFD 会话
Switch(config-router)# enable	使能 VRRP 组 2

I. 命令验证

```
Switch# show vrrp 1
```

```
VRID <1>
 State               : Initialize(Interface down)
 Virtual IP          : 10.10.10.81(IP owner)
 Interface           : eth-0-1
 VMAC                : 0000.5e00.0101
 VRF                 : Default
 Advt timer          : 5 second(s)
 Preempt mode        : TRUE
 Conf pri            : Unset          Run pri   : 255
 Master router ip    : Unknown
 Master priority     : Unknown
 Master advt timer   : Unknown
 Master down timer   : Unknown
 Preempt delay       : 0 second(s)
 Learn master mode   : FALSE
Switch# show vrrp 2
VRID <2>
 State               : Initialize(Interface down)
 Virtual IP          : 10.10.10.82(Not IP owner)
 Interface           : eth-0-1
 VMAC                : 0000.5e00.0102
 VRF                 : Default
 Advt timer          : 5 second(s)
 Preempt mode        : TRUE
 Conf pri            : 200            Run pri   : 200
 Master router ip    : Unknown
 Master priority     : Unknown
 Master advt timer   : Unknown
 Master down timer   : Unknown
 Preempt delay       : 0 second(s)
 Learn master mode   : FALSE
```

16.9.7 配置 VRRP Circuit Failover

之所以需要 VRRP 链路故障检测功能，是由于 VRRPv2 无法跟踪网关上行链路状态。引入对上行链路的监控可以有效的驱动虚拟路由器的切换从而避免“黑洞路由”。当主路由器上行接口链路发生故障时，原来的 master 路由器将切换为 backup 路由器，而原来的 backup 路由器将接替成为 master 路由器。

为了实现 VRRP 链路故障检测功能，我们需要为监视的接口配置一个 priority-delta 值，这个值将被附加到 master 路由器上以实现 VRRP 路由器从 master 从 backup 的切换。

在下面的例子中，两个路由器 R1 和 R2 配置了不同的优先级值，priority-delta 的配置要大于 R1 和 R2 优先级之间的差值。R1 配置有一个 100 的优先级，R2 有一个 90 的优先级，由于 R1 优先级较高成为 master 路由器。priority-delta 值配置为 20，当在 R1 上的上行接口 eth2 发生故障，R1 的优先级将变为 80（100－20）。此时由于 R2 比 R1 有更大的优先级，R2 变成 Master。当 R1 恢复时候，R1 优先级为 100，重新成为 Master。

I. 拓扑

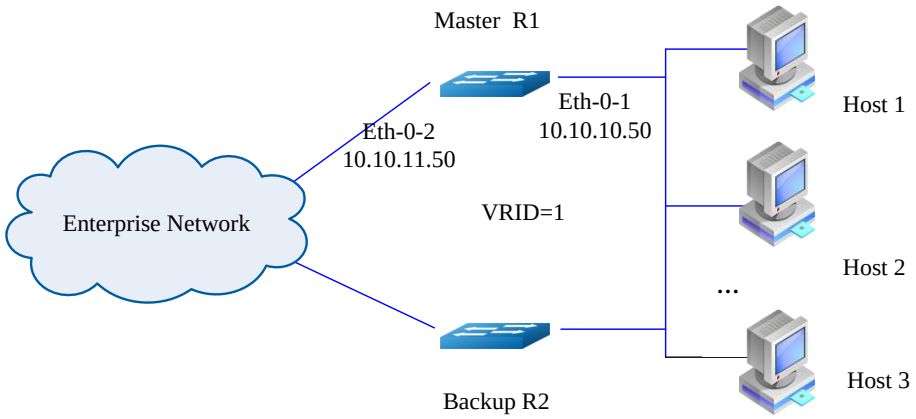


图16-10 VRRP Example

II. 配置

R1

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置三层接口
Switch(config-if)# ip address 10.10.10.50/24	配置 IP 地址
Switch(config-if)# exit	退出接口地址
Switch(config)# interface eth-0-2	进入接口模式
Switch(config-if)# no switchport	设置三层接口
Switch(config-if)# ip address 10.10.11.50/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# track 10 interface eth-0-2 linkstate	创建跟踪条件为端口的链路状态
Switch(config)# router vrrp 1	创建 VRRP 组 1
Switch(config-router)# virtual-ip 10.10.10.1	指定虚拟 IP 地址
Switch(config-router)# interface eth-0-1	配置 VRRP 组应用端口
Switch(config-router)# preempt-mode true	设置抢占模式
Switch(config-router)# advertisement-interval 5	配置通告时间间隔 5 秒
Switch(config-router)# priority 100	配置 VRRP 的优先级 100

Switch(config-router)# track 10 decrement 20	跟踪 track10 并且 priority-delta 值为 20
Switch(config-router)# enable	使能 VRRP 组 1

R2

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	设置三层接口
Switch(config-if)# ip address 10.10.10.40/24	配置 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# router vrrp 1	创建 VRRP 组 1
Switch(config-router)# virtual-ip 10.10.10.1	设置虚拟 IP 地址
Switch(config-router)# interface eth-0-1	配置 VRRP 组应用端口
Switch(config-router)# preempt-mode true	设置抢占模式
Switch(config-router)# advertisement-interval 5	配置通告时间间隔 5 秒
Switch(config-router)# priority 90	配置优先级 90
Switch(config-router)# enable	使能 VRRP1

16.9.8 限制

VRRP 不支持 MD5 验证。

16.10 Track 配置

16.10.1 配置 IP SLA

I. 简介

IP SLA (Service Level Agreement) 是一种通过”动态监测”来实施网络性能的测量和诊断工具。”动态监测”是指在交换机中， 用 ping 的方式，来衡量网络是否连通和网络的性能。每一个 IP SLA 操作均维护其各自操作时生成的一个返回值。这个返回值将会被 tracking 进程所中断。返回值可以是 OK，超过阈值，还有其他返回代码。不同的操作可以有不同的返回值。因此在系统中， 只使用那些对于所有操作类型来说共通的返回值。在 IPSLA 中，我们可以通过使用 ICMP echo 来检查状态或路由的可达性。

II. 拓扑

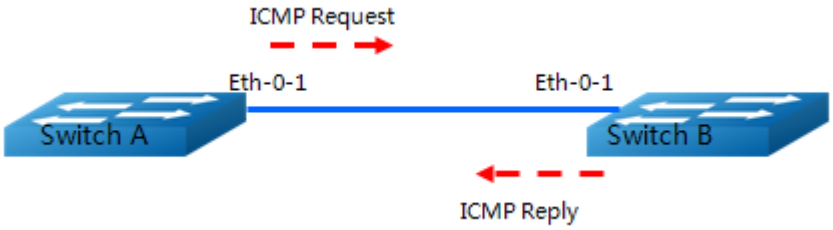


图16-11 拓扑

III. 配置 VRF 接口

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# ip vrf vpn1	创建 VRF 条目
Switch(config-vrf)# exit	退出 VRF 模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip vrf forwarding vpn1	在接口上启用 VRF 转发表
Switch(config-if)# ip address 192.168.0.2/24	配置 IP 地址
Switch(config)# ip sla monitor 1	创建一个 IPSLA 条目并且进入 IPSLA 配置模式
Switch(config-ipsla)# type icmp-echo 192.168.0.1	定义一个 ICMP 报文的 echo 操作，并输入它的目的 IP 地址
Switch(config-ipsla)# frequency 35	设置发送间隔
Switch(config-ipsla)# timeout 6	设置超时时间
Switch(config-ipsla)# threshold 6000	设置阈值时间
Switch(config-ipsla)# ttl 65	设置 ttl
Switch(config-ipsla)# tos 1	设置 tos
Switch(config-ipsla)# data-size 29	设置 data size
Switch(config-ipsla)# data-pattern abababab	设置 data pattern
Switch(config-ipsla)# fail-percent 90	设置 fail 百分比
Switch(config-ipsla)# packets-per-test 4	设置单次检测探针数
Switch(config-ipsla)# interval 9	设置探针之间时间间隔

Switch(config-ipsla)# statistics packet 10	设置 packet 统计数
Switch(config-ipsla)# statistics test 3	设置保存最近几次测试结果
Switch(config-ipsla)# vrf vpn1	应用 VPN1
Switch(config-ipsla)# exit	退出 IPSLA 模式
Switch(config)# ip sla monitor schedule 1	启用 IP SLA 功能
Switch(config)# exit	退出配置模式

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# ip vrf vpn1	创建 VRF 条目
Switch(config-vrf)# exit	退出 VRF 模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip vrf forwarding vpn1	在接口上启用 VRF 转发表
Switch(config-if)# ip address 192.168.0.1/24	配置 IP 地址

I. 命令验证

```
DUT1# sho ip sla monitor 1
Entry 1
    Type                 : Echo
    Admin state          : Disable
    Destination address  : 192.168.0.1
    Frequency            : 35s
    Timeout              : 6s
    Threshold            : 6000ms
    Interval             : 9s
    Packet per test      : 4
    TTL                  : 65
    TOS                  : 1
    Data Size            : 29 bytes
    Fail Percent         : 90%
    Packet Item Cnt      : 10
    Test Item Cnt        : 3
    Vrf                  : vpn1
    Return code          : Unknown
```

II. 配置三层接口

Switch A

Switch#configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip address 192.168.0.2/24	配置 IP 地址
Switch(config)# ip sla monitor 1	创建一个 IPSLA 条目并且进入 IPSLA 配置模式
Switch(config-ipsla)# type icmp-echo 192.168.0.1	定义一个 ICMP 报文的 echo 操作，并输入它的目的 IP 地址或者主机名
Switch(config-ipsla)#frequency 10	设置发送间隔
Switch(config-ipsla)#timeout 5	设置超时时间
Switch(config-ipsla)#threshold 1	设置阀值时间
Switch(config-ipsla)#exit	退出 IP SLA 模式
Switch(config)# ip sla monitor schedule 1	启用 IP SLA 功能
Switch(config)#exit	退出配置模式

Switch B

Switch#configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip address 192.168.0.1/24	配置 IP 地址

I. 命令验证

```
Switch# show ip sla monitor
Entry 1
    Type                    : Echo
    Admin state                 : Enable
    Destination address         : 192.168.0.1
    Frequency                   : 10 seconds
    Timeout                 : 5 seconds
    Threshold                   : 5 seconds
    Running Frequency           : 8 seconds
Return code                 : OK
Switch# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
64 bytes from 192.168.0.1: icmp_seq=1 ttl=64 time=0.846 ms
```

```
64 bytes from 192.168.0.1: icmp_seq=2 ttl=64 time=0.643 ms
64 bytes from 192.168.0.1: icmp_seq=3 ttl=64 time=0.978 ms
64 bytes from 192.168.0.1: icmp_seq=4 ttl=64 time=0.640 ms
64 bytes from 192.168.0.1: icmp_seq=5 ttl=64 time=0.704 ms
```

Switch B

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入端口模式
Switch(config-if)# shutdown	打开端口

```
Switch# show ip sla monitor
Entry 1
    Type                        : Echo
    Admin state                     : Enable
    Destination address             : 192.168.0.1
    Frequency                       : 10 seconds
    Timeout                     : 5 seconds
    Threshold                       : 5 seconds
    Running Frequency               : 9 seconds
    Running Timeout                 : 4 seconds
    Running Threshold               : 4 seconds
Return code                     : Timeout
```

I. 配置静态路由的端口

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip address 192.168.0.2/24	配置 IP 地址
Switch(config)# ip sla monitor 2	创建 SLA 条目
Switch(config-ipsla)# type icmp-echo 1.1.1.1	定义一个 ICMP 报文的 echo 操作，并输入它的目的 IP 地址或者主机名
Switch(config-ipsla)# frequency 10	设置发送间隔
Switch(config-ipsla)# timeout 5	设置超时时间
Switch(config-ipsla)# threshold 1	设置阀值时间
Switch(config-ipsla)# exit	退出 IP SLA 模式
Switch(config)# ip sla monitor schedule 2	启用 IP SLA 功能
Switch(config)# exit	退出配置模式

I. 命令验证

```
Switch# show ip sla monitor 2
Entry 2
    Type                        : Echo
    Admin state                     : Enable
    Destination address             : 1.1.1.1
    Frequency                   : 10 seconds
    Timeout                     : 5 seconds
    Threshold                       : 5 seconds
    Running Frequency               : 1 seconds
    Return code                     : Unreachable
Switch# ping 1.1.1.1
connect: Network is unreachable
```

Switch A

Switch#configure terminal	进入配置模式
Switch(config)# ip route 1.1.1.1/32 192.168.0.1	配置静态路由

```
Switch# ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=64 time=1.03 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=64 time=1.63 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=64 time=0.661 ms
64 bytes from 1.1.1.1: icmp_seq=4 ttl=64 time=0.762 ms
64 bytes from 1.1.1.1: icmp_seq=5 ttl=64 time=0.942 ms
Switch# show ip sla monitor 2
Entry 2
    Type                        : Echo
    Admin state                     : Enable
    Destination address             : 1.1.1.1
    Frequency                   : 10 seconds
    Timeout                     : 5 seconds
    Threshold                       : 5 seconds
    Running Frequency               : 8 seconds
    Return code                     : OK
```

16.10.2 配置 TRACK

I. 简介

VRRP 的监视接口功能更好地扩充了备份功能：不仅能在备份组中某路由器的接口出现故障时提供备份功能，还能在路由器的其它接口（如连接上行链路的接口）不可用时提供备份功能。路由器连接上行链路的接口出现故障时，备份组无法感知上行链路的故障，如果该路由器此时处于 Master 状态，将会导致局域网内的主机无法访问外部网络。通过监视指定接口的功能，可以解决该问题。当连接上行链路的接口处于 down 状态时，路由器主动降低自己的优先级，使得备份组内其它路由器的优先级高于这个路由器，以便优先级最高的路由器成为 Master，承担转发任务。

II. 拓扑

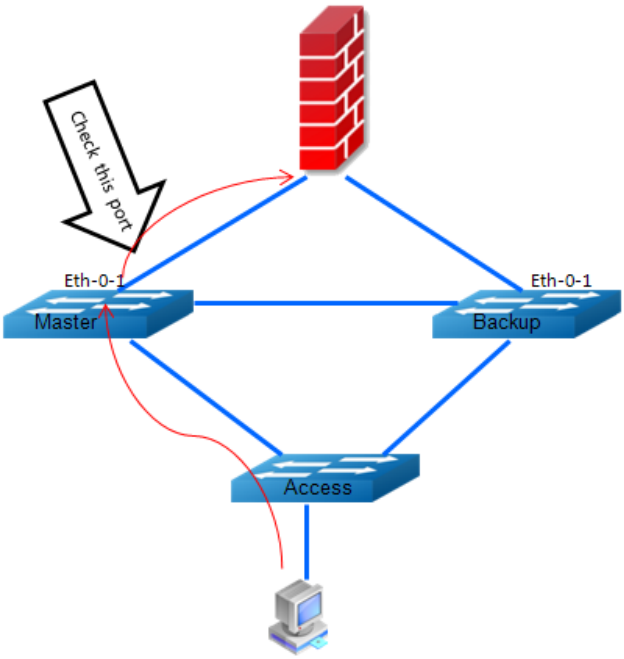


图16-12 TRACK 拓扑

III. 配置

Switch# configure terminal	进入配置模式
Switch(config)# track 1 interface eth-0-1 linkstate	创建跟踪的条目
Switch(config-track)# delay up 30	从 Down 到 Up 的时间
Switch(config-track)# delay down 30	从 Up 到 Down 的时间
Switch(config-track)#exit	退出监控模式
Switch(config)# exit	退出配置模式

IV. 命令验证

Switch#show track

```
Track 2
    Type                    : Interface Link state
    Interface               : eth-0-1
    State                   : down
    Delay up                : 30 seconds
Delay down                  : 30 seconds
```

V. 拓扑

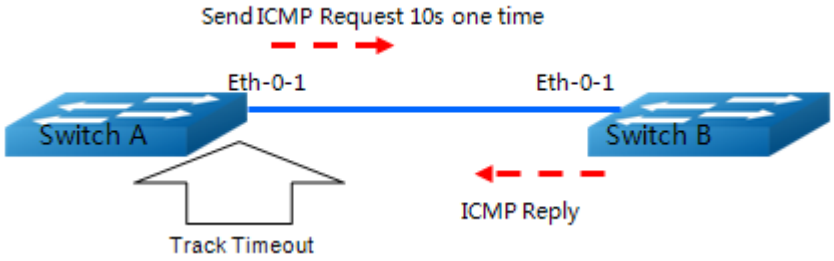


图16-13 拓扑

VI. 配置

Switch A

Switch# configure terminal	进入配置模式
Switch(config)# track 1 rtr 1 reachability	配置 Track 的可达性
Switch(config-track)# delay up 30	从 Down 到 Up 的时间
Switch(config-track)# delay down 30	从 Up 到 Down 的时间
Switch(config-track)#exit	退出监控模式
Switch(config)#exit	退出配置模式
Switch#configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip address 192.168.0.2/24	配置 IP 地址
Switch(config)# ip sla monitor 1	创建 SLA 的条目
Switch(config-ipsla)# type icmp-echo 192.168.0.1	定义一个 ICMP 报文的 echo 操作，并输入它的目的 IP 地址
Switch(config-ipsla)# frequency 10	设置发送间隔
Switch(config-ipsla)# timeout 5	设置超时时间
Switch(config-ipsla)# threshold 1	设置阀值时间
Switch(config-ipsla)# exit	退出 SLA 模式
Switch(config)# ip sla monitor schedule 1	启用 IP SLA 功能
Switch(config)# exit	退出配置模式

Switch B

Switch#configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip address 192.168.0.1/24	配置 IP 地址

I. 验证

```
Switch#show track
Track 1
    Type                    : Response Time Reporter(RTR) Reachability
    RTR entry number      : 1
    State                     : up
    Delay up                : 30 seconds
Delay down                 : 30 seconds
```

II. 拓扑

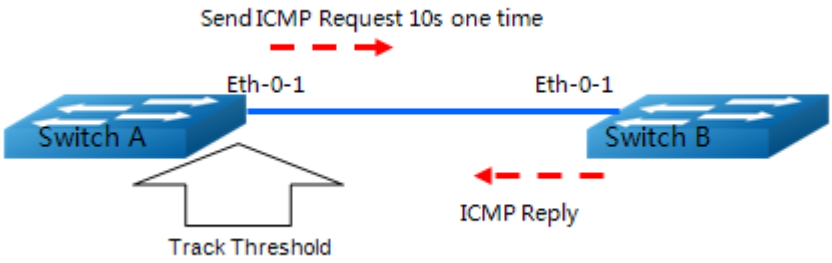


图16-14 拓扑

III. 配置

Switch A

命令	描述
Switch#configure terminal	进入配置模式
Switch(config)# track 1 rtr 1 state	配置 Track 的状态
Switch(config-track)# delay up 30	从 Down 到 Up 的时间
Switch(config-track)# delay down 30	从 Up 到 Down 的时间
Switch(config-track)#exit	退出 Track 模式
Switch(config)#exit	退出配置模式
Switch#configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式

命令	描述
Switch(config-if)# ip address 192.168.0.2/24	配置 IP 地址
Switch(config)# ip sla monitor 1	创建 IP Sla 的条目
Switch(config-ipsla)# type icmp-echo 192.168.0.1	定义 IP SLA 的协议类型
Switch(config-ipsla)#frequency 10	设置发送间隔
Switch(config-ipsla)#timeout 5	设置超时时间
Switch(config-ipsla)#threshold 1	设置阀值时间
Switch(config-ipsla)#exit	退出 IP SLA 模式
Switch(config)# ip sla monitor schedule 1	启用 IP SLA 的监控
Switch(config)#exit	退出配置模式

Switch B

命令	描述
Switch#configure terminal	进入配置模式
Switch(config)# interface eth-0-1	进入接口模式
Switch(config-if)# ip address 192.168.0.1/24	配置 IP 地址

I. 命令验证

```
Switch# show track
Track 1
    Type                   : Response Time Reporter(RTR) State
    RTR entry number     : 1
    State                  : up
    Delay up              : 30 seconds
Delay down                : 30 seconds
```

16.10.3 配置 track bfd

拓扑

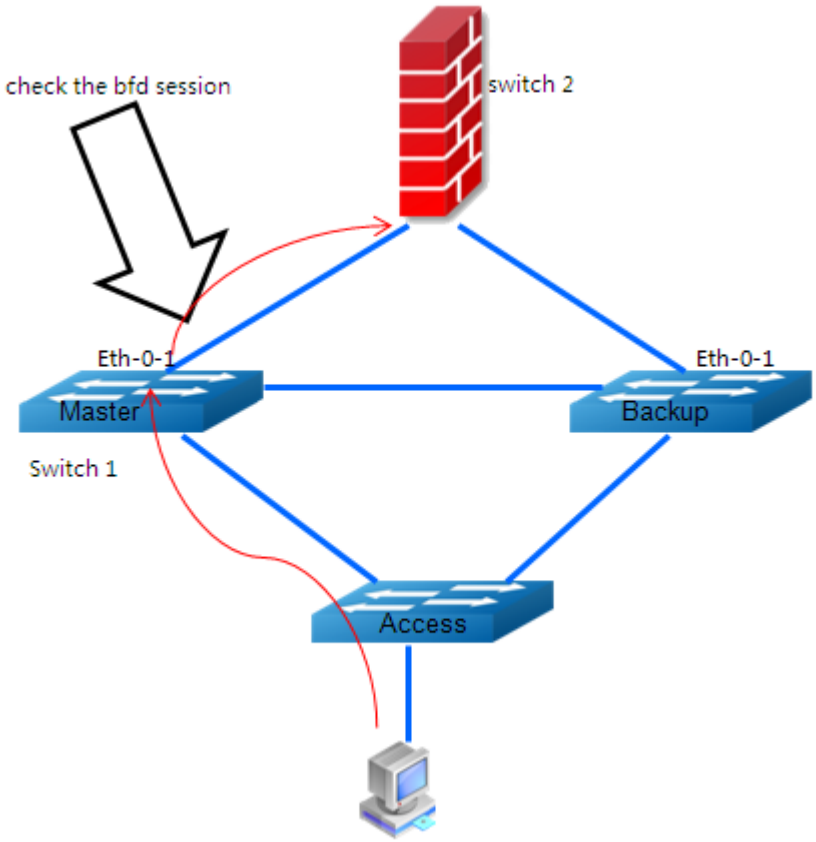


图16-15 VRRP Track bfd 拓扑

配置

根据下表步骤，配置 vrrp track bfd：

switch1 配置

Switch1# configure terminal	进入配置模式
Switch 1(config)# interface eth-0-1	进入端口配置模式
Switch 1(config-if)# no switchport	设置端口为三层接口
Switch 1(config-if)# no shutdown	使能端口
Switch 1(config-if)# ip address 9.9.9.1/24	配置 IP 地址
Switch 1(config-if)# quit	退出端口模式
Switch1(config)# track 1 bfd source	创建 bfd session 的 track 对象

interface eth-0-1 destination 9.9.9.2	
Switch1(config-track)# delay up 30	从 Down 到 Up 的时间
Switch1(config-track)# delay down 30	从 Up 到 Down 的时间
Switch1(config-track)# exit	退出 Track 模式
Switch1(config)# exit	退出配置模式

switch2 配置

Switch2# configure terminal	进入配置模式
Switch2(config)# interface eth-0-1	进入端口配置模式
Switch2(config-if)# no switchport	设置端口为三层接口
Switch2(config-if)# no shutdown	使能端口
Switch2(config-if)# ip address 9.9.9.2/24	配置 IP 地址
Switch2(config-if)# quit	退出端口模式
Switch2(config)# track 1 bfd source interface eth-0-1 destination 9.9.9.1	创建 bfd session 的 track 对象
Switch1(config-track)# delay up 30	从 Down 到 Up 的时间
Switch1(config-track)# delay down 30	从 Up 到 Down 的时间
Switch1(config-track)# exit	退出 Track 模式
Switch1(config)# exit	退出配置模式

命令验证

使用下列命令显示 track bfd 的配置：

```
Switch #show track
Track 1
    Type                : BFD state
    Source interface    : eth-0-1
    Destination IP      : 9.9.9.2
    BFD Local discr     : 1
    State               : up
Switch2 # show track
Track 1
    Type                : BFD state
    Source interface    : eth-0-1
    Destination IP      : 9.9.9.1
    BFD Local discr     : 1
    State               : up
```

16.10.4 配置 VRRP TRACK

I. 拓扑

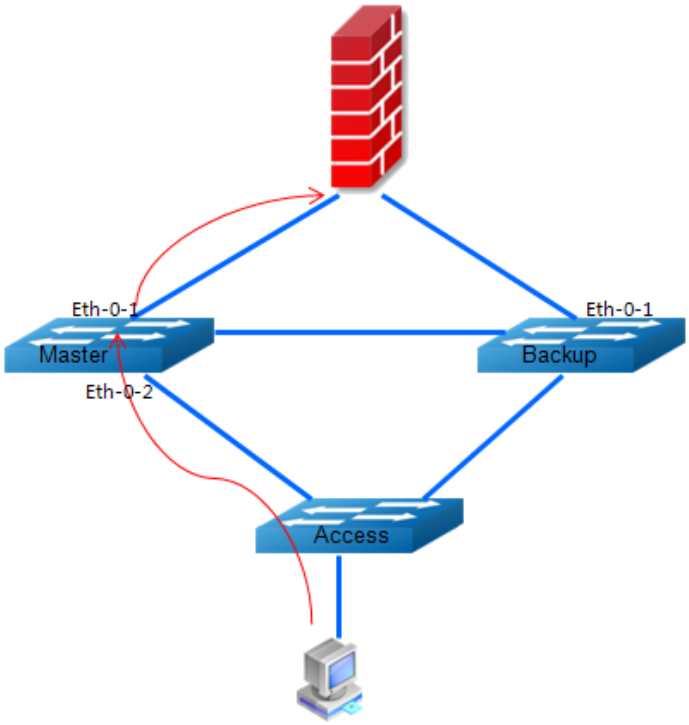


图16-16 拓扑

II. 配置

Switch# configure terminal	进入配置模式
Switch(config)# track 1 interface eth-0-1 linkstate	配置 Track 条目
Switch(config-track)# exit	退出 Track 模式
Switch(config)# router vrrp 1	创建 VRRP 条目
Switch(config-router)# track 1 decrement 30	设置 VRRP 的规则
Switch(config-router)# exit	退出路由模式
Switch(config)# exit	退出配置模式

III. 命令验证

```
Switch# show vrrp
VRID <1>
 State              : Master
 Virtual IP          : 172.16.10.100(Not IP owner)
 Interface           : eth-0-2
 VMAC                 : 0000.5e00.0101
```

```
Advt timer          : 1
Preempt mode       : TRUE
Auth type           : NONE
Conf pri            : Unset          Run pri    : 70
Track Object        : 1
Delta pri            : 30
Master router ip     : 172.16.10.1
Master priority      : 70
Master advt timer    : 1
Master down timer    : 4
Learn master mode    : FALSE
```

16.10.5 配置静态路由 TRACK

I. 拓扑

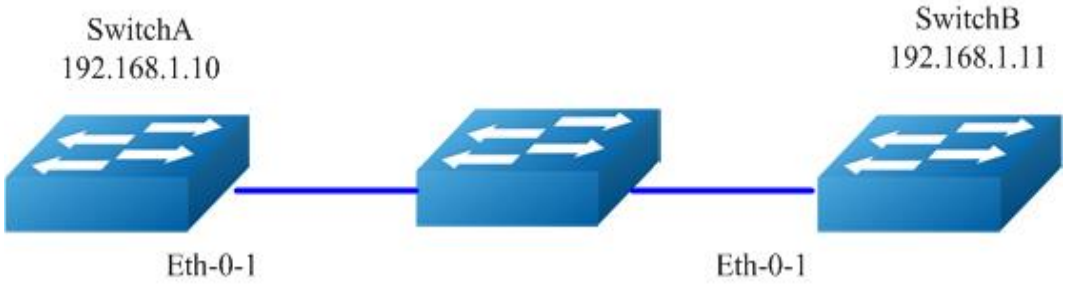


图16-17 静态路由 Track 拓扑

II. 配置

Switch A

Switch# configure terminal	进入全局配置模式
Switch(config)#interface eth-0-1	进入接口模式
Switch(config-if)# no switchport	将端口配置为三层接口
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# ip address 192.168.1.10/24	配置接口 IP 地址
Switch(config-if)# exit	退出接口模式
Switch(config)# ip sla monitor 1	创建一条 IP SLA 条目并且进入 IP SLA 配置模式.
Switch(config-ipsla)# type icmp-echo 192.168.1.11	定义一个 ICMP 报文的 echo 操作，并输入它的目的 IP 地址
Switch(config-ipsla)# exit	退出 IP SLA 配置模式
Switch(config)# ip sla monitor schedule 1	启用 IP SLA 功能

Switch(config)# track 1 rtr 1 reachability	配置 Track 条目，并进入 Track 配置模式.
Switch(config-track)# exit	退出 Track 配置模式
Switch(config)#ip route 10.10.10.0/24 192.168.1.11 track 1	配置静态路由并指定 Track 条目
Switch(config)# exit	退出全局配置模式

Switch B

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# no switchport	将端口配置为三层接口
Switch(config-if)# no shutdown	打开端口
Switch(config-if)# ip address 192.168.1.11/24	配置接口 IP 地址

I. 命令验证

```
Switch# show ip sla monitor 1
Entry 1
    Type                  : Echo
    Admin state           : Enable
    Destination address   : 192.168.1.11
    Frequency             : 60 seconds
    Timeout               : 5 seconds
    Threshold             : 5 seconds
    Running Frequency     : 49 seconds
Return code           : OK
Switch# show track 1
Track 1
    Type                : Response Time Reporter(RTR) Reachability
    RTR entry number    : 1
    State               : up
Switch# show ip route static
S        10.10.10.0/24 [1/0] via 192.168.1.11, eth-0-1
```

Switch B

Switch# configure terminal	进入全局配置模式
Switch(config)# interface eth-0-1	进入接口配置模式
Switch(config-if)# shutdown	关闭端口

```
Switch# show ip sla monitor 1
```

```
Entry 1
    Type                 : Echo
    Admin state          : Enable
    Destination address  : 192.168.1.11
    Frequency            : 60 seconds
    Timeout              : 5 seconds
    Threshold            : 5 seconds
    Running Frequency    : 8 seconds
Return code          : Timeout
Switch# show track 1
Track 1
    Type                : Response Time Reporter(RTR) Reachability
    RTR entry number    : 1
    State               : down
Switch# show ip route static
Switch#
```

16.11 IP BFD 配置

16.11.1 简介

随着对网络的可靠性要求越来越高，快速寻找、切换到备份链路保证网络通畅也显得越来越重要。但是对于很多硬件或者软件无法提供这个功能，比如以太网。还有一些无法实现路径检测，比如转发引擎或者接口等，无法实现端到端的检测。

目前的网络一般采用慢 Hello 机制，尤其在路由协议中，在没有硬件帮助下，检测时间会很长。当数据速率越来越大，故障感应时间长代表着大量数据的丢失，并且对于不允许路由协议的节点没有办法检测链路的状态。同时，在现有的 IP 网络中并不具备秒以下的间歇性故障修复功能，而传统路由架构在对实时应用（如语音）进行准确故障检测方面能力有限。

BFD（双向链路检测），提出了一种轻载的、快速的链路状态检测的解决方案。BFD 能够在系统之间的任何类型通道上进行故障检测，这些通道包括直接的物理链路、虚电路、隧道、MPLS LSP、多跳路由通道，以及非直接的通道。

16.11.2 限制

当物理口上配置了 CFM 的 mep 并且使能了 LM， 同时，IP BFD 配置在 vlan interface 上且该物理口是 vlan interface 的 member, 则 IP BFD 无法正常工作。当 LM 关闭后，IP BFD 应可以正常工作。

16.11.3 拓扑

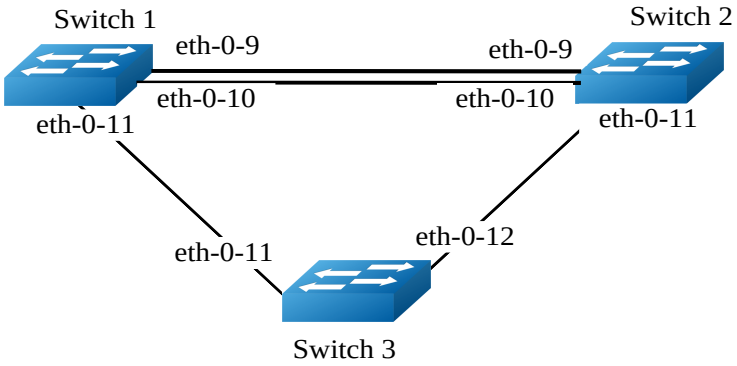


图16-18 IP BFD 单跳会话基本配置

16.11.4 配置

这个拓扑包含 3 条 BFD 会话，其中一条基于静态配置且绑定静态路由，一条基于 OSPF，一条是 bfd 与 vrrp 联动的。

Switch1 的配置

Switch1# configure terminal	进入全局配置模式
Switch1(config)# interface eth-0-9	进入 eth-0-9 接口配置模式
Switch1(config-if)# no switchport	将接口设置为非二层口
Switch1(config-if)# no shutdown	使能接口
Switch1(config-if)# ip address 9.9.9.1/24	配置接口 ip 地址
Switch1(config-if)# bfd interval mintx 1 minrx 1 multiplier 3	配置接口的 BFD 收发包速录和检测倍数
Switch1(config-if)# exit	退出接口模式
Switch1(config)# interface eth-0-10	进入 eth-0-10 接口配置模式
Switch1(config-if)# no switchport	将接口设置为非二层口
Switch1(config-if)# no shutdown	使能接口
Switch1(config-if)# ip address 10.10.10.1/24	配置接口 ip 地址
Switch1(config-if)# bfd interval mintx 2 minrx 2 multiplier 3	配置接口的 BFD 收发包速录和检测倍数
Switch1(config-if)# ip ospf bfd	使能基于 OSPF 的 BFD
Switch1(config-if)# exit	退出接口模式

Switch1(config)# router ospf	进入 OSPF 模式
Switch1 (config-router)# network 10.10.10.0/24 area 0	配置 OSPF 网段
Switch1 (config-router)# exit	退出 OSPF 模式
Switch1 (config)#interface eth-0-11	进入端口模式
Switch1 (config-if)#no switchport	设置端口为三层接口
Switch1 (config-if)#ip address 11.11.11.1/24	设置 IP 地址
Switch 1(config-if)#exit	退出接口模式
Switch1 (config)#router vrrp 1	创建虚拟路由器组 1
Switch (config-router)#virtual-ip 11.11.11.100	设置虚拟 IP 地址.
Switch 1(config-router)#interface eth-0-11	配置 VRRP 组的应用端口
Switch 1(config-router)# bfd 11.11.11.2	配置 BFD 会话
Switch1(config-router)# enable	使能 VRRP 组 1
Switch1(config)# bfd test peer-ip 9.9.9.2 interface eth-0-9 auto	创建 bfd session 会话
Switch1(config)# ip route 1.1.1.0/24 9.9.9.2 bind bfd test	配置静态路由 并绑定 BFD 会话
Switch1(config)# end	退出全局配置模式

Switch2 的配置

Switch2# configure terminal	进入全局配置模式
Switch2(config)# interface eth-0-9	进入 eth-0-9 接口配置模式
Switch2(config-if)# no switchport	将接口设置为非二层口
Switch2(config-if)# no shutdown	使能接口
Switch2(config-if)# ip address 9.9.9.2/24	配置接口 ip 地址
Switch2(config-if)# bfd interval mintx 1 minrx 1 multiplier 3	配置接口的 BFD 收发包速录和检测倍数
Switch2(config-if)# exit	退出接口模式
Switch2(config)# interface eth-0-10	进入 eth-0-10 接口配置模式
Switch2(config-if)# no switchport	将接口设置为非二层口

Switch2(config-if)# no shutdown	使能接口
Switch2(config-if)# ip address 10.10.10.2/24	配置接口 ip 地址
Switch2(config-if)# bfd interval mintx 2 minrx 2 multiplier 3	配置接口的 BFD 收发包速录和检测倍数
Switch2(config-if)# ip ospf bfd	使能基于 OSPF 的 BFD
Switch2(config-if)# exit	退出接口模式
Switch2(config)# router ospf	进入 OSPF 模式
Switch2 (config-router)# network 10.10.10.0/24 area 0	配置 OSPF 网段
Switch2 (config-router)# exit	退出 OSPF 模式
Switch2 (config)#interface eth-0-11	进入端口模式
Switch2 (config-if)#no switchport	设置端口为三层接口
Switch2(config-if)#ip address 11.11.11.2/24	设置 IP 地址
Switch2 (config-if)#exit	退出接口模式
Switch2 (config)#router vrrp 1	创建虚拟路由器组 1
Switch2 (config-router)#virtual-ip 11.11.11.100	设置虚拟 IP 地址.
Switch2 (config-router)#interface eth-0-11	配置 VRRP 组的应用端口
Switch2 (config-router)# bfd 11.11.11.1	配置 BFD 会话
Switch2 (config-router)# enable	使能 VRRP 组 1
Switch1(config)# bfd test peer-ip 9.9.9.1 interface eth-0-9 auto	创建 bfd session 会话
Switch2(config)# ip route 2.2.2.0/24 9.9.9.1 bind bfd test	配置静态路由并绑定 BFD 会话
Switch2(config)# end	退出全局配置模式

Switch3 的配置

Switch3# configure terminal	进入全局配置模式
Switch3(config)# interface eth-0-11	进入 eth-0-11 接口配置模式
Switch3(config-if)# no shutdown	使能接口
Switch 3(config-if)#exit	退出接口模式

Switch3(config)# interface eth-0-12	进入 eth-0-12 接口配置模式
Switch3(config-if)# no shutdown	使能接口
Switch 3(config-if)#exit	退出接口模式

16.11.5 命令验证

使用命令“show bfd session”查看配置结果，详细信息类似如下所示。

```
Switch1# show bfd session
abbreviation:
LD: local Discriminator.    RD: Discriminator
S: single hop session.    M: multi hop session.
SD: Static Discriminator.  DD: Dynamic Discriminator
A: Admin down.    D:down.    I:init.    U:up.
=====================================================
LD   RD   TYPE ST   UP-Time    Remote-Addr          vrf
1    1    S-DD U    00:01:05   9.9.9.2        default
2    2    S-DD U    00:00:25   10.10.10.2     default
3    3    S-DD U    00:00:25   11.11.11.2     default
Number of Sessions:    3
Switch2# show bfd session
abbreviation:
LD: local Discriminator.    RD: Discriminator
S: single hop session.    M: multi hop session.
SD: Static Discriminator.  DD: Dynamic Discriminator
A: Admin down.    D:down.    I:init.    U:up.
=====================================================
LD   RD   TYPE ST   UP-Time    Remote-Addr         vrf
1    1    S-DD    U    00:01:27   9.9.9.1       default
2    2    S-DD    U    00:00:46   10.10.10.1    default
3    3    S-DD    U    00:00:25   11.11.11.3    default
Number of Sessions:    3
```

16.11.6 多跳拓扑

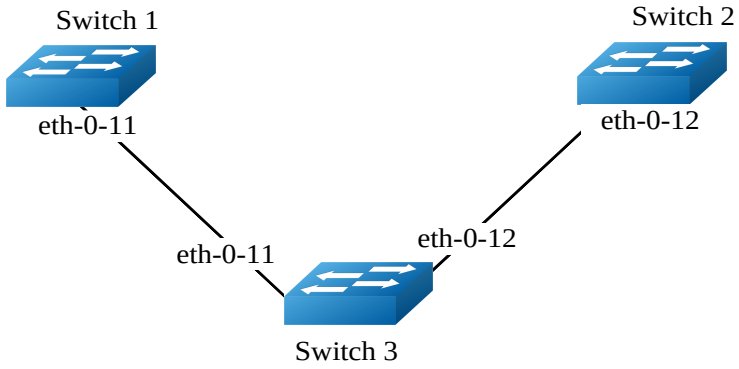


图16-19 IP BFD 多跳会话基本配置

16.11.7 多跳配置

这个拓扑包含静态配置的多跳 bfd 会话且绑定静态路由。

Switch1 的配置

Switch1# configure terminal	进入全局配置模式
Switch1(config)# interface eth-0-11	进入 eth-0-11 接口配置模式
Switch1(config-if)# no switchport	将接口设置为非二层口
Switch1(config-if)# no shutdown	使能接口
Switch1(config-if)#ip address 11.11.11.1/24	配置接口上的 ip 地址
Switch1(config-if)#exit	退出接口模式
Switch1(config)#ip route 12.12.12.2/24 11.11.11.2	配置到达 switch3 的静态路由
Switch1(config)# bfd test peer-ip 12.12.12.2/24 source 11.11.11.1 local 10 remote 20	配置静态多跳 bfd 且指定本地标识符
Switch1(config)# ip route 192.168.1.1/24 12.12.12.2 bind bfd test	将 bfd 与某个静态路由绑定

Switch2 的配置

Switch2# configure terminal	进入全局配置模式
Switch2(config)# interface eth-0-11	进入 eth-0-11 接口配置模式

Switch2(config-if)# no switchport	将接口设置为非二层口
Switch2(config-if)# no shutdown	使能接口
Switch2(config-if)#ip address 11.11.11.2/24	配置接口 ip 地址
Switch2(config-if)#exit	退出接口模式
Switch2(config)#interface eth-0-12	进入 eth-0-12 接口配置模式
Switch2(config-if)#no switchport	将接口设置为非二层口
Switch2(config-if)#no shutdown	使能接口
Switch2(config-if)#ip address 12.12.12.1/24	配置接口 ip 地址
Switch2(config-if)#exit	退出接口模式

Switch3 的配置

Switch2# configure terminal	进入全局配置模式
Switch2(config)# interface eth-0-12	进入 eth-0-11 接口配置模式
Switch2(config-if)#no switchport	将接口设置为非二层口
Switch2(config-if)#no shutdown	使能接口
Switch2(config-if)#ip address 12.12.12.2/24	配置接口 ip 地址
Switch2(config-if)#exit	退出接口模式
Switch2(config)#ip route 11.11.11.1/24 12.12.12.1	配置到达 swith1 的静态路由
Switch2(config)#bfd test peer-ip 11.11.11.1 source-ip 12.12.12.2 local 20 remote 10	配置静态多跳 bfd
Switch2(config)#ip route 2.2.2.2/24 11.11.11.1 bind bfd test	配置静态路由绑定 bfd

16.11.8 多跳命令验证

使用命令“show bfd session”查看配置结果，详细信息类似如下所示。

```
Switch1# show bfd session
abbreviation:
LD: local Discriminator.    RD: Discriminator
S: single hop session.    M: multi hop session.
SD: Static Discriminator.  DD: Dynamic Discriminator
A: Admin down.    D:down.    I:init.    U:up.
=====================================================
```

```
LD   RD   TYPE ST   UP-Time    Remote-Addr         vrf
10   20    S-SD U    00:01:27   12.12.12.2       default
Switch1# show bfd session
abbreviation:
LD: local Discriminator.    RD: Discriminator
S: single hop session.    M: multi hop session.
SD: Static Discriminator.  DD: Dynamic Discriminator
A: Admin down.    D:down.    I:init.    U:up.
=======================================================
LD   RD   TYPE ST   UP-Time    Remote-Addr         vrf
20   10    S-SD U    00:01:27   11.11.11.1       default
```

16.12 VARP 配置

16.12.1 简介

虚拟 ARP 允许多台交换机根据相同的目的 MAC 地址同时路由报文。每台交换机都会配置相同的虚拟 MAC 地址，作为 VLAN 接口上虚拟 ip 地址的对应 MAC 地址。因为虚拟 ARP 工作在双活模式，并且没有额外的开销，所以在 MLAG 的应用环境中优于 VRRP.

对于虚拟 IP 地址的 ARP 和 GARP 请求, 虚拟 ARP 将会使用虚拟 MAC 地址回应。虚拟 MAC 地址只会在入方向的报文里出现，不会出现在出方向的报文源 IP 字段。

16.12.2 拓扑

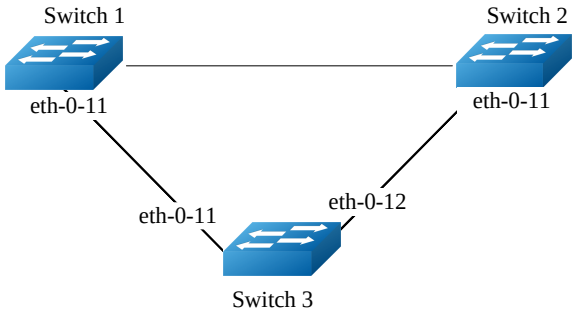


图16-20 VARP & MLAG 拓扑

16.12.3 配置

配置 VARP.

Switch1 的配置

Switch1# configure terminal	进入全局配置模式
Switch1(config)# ip virtual-router mac a.a.a	配置虚拟 MAC 地址

Switch1(config)# vlan database	进入 VLAN 配置模式
Switch1(config-vlan)# vlan 2	创建 VLAN 2
Switch1(config-vlan)# exit	退出 VLAN 配置模式
Switch1(config)# interface eth-0-11	进入 eth-0-11 的接口配置模式
Switch1(config-if)# switchport access vlan 2	将接口加入 VLAN 2
Switch1(config-if)# no shutdown	使能接口
Switch1(config-if)# interface vlan 2	进入 vlan2 的接口配置模式
Switch1(config-if)# ip address 10.10.10.1/24	配置 IP 地址
Switch1(config-if)# ip virtual-router address 10.10.10.254	配置虚拟 IP 地址
Switch1(config-if)# end	退出接口配置模式

Switch2 的配置

Switch2# configure terminal	进入全局配置模式
Switch2(config)# ip virtual-router mac a.a.a	配置虚拟 MAC 地址
Switch2(config)# vlan database	进入 VLAN 配置模式
Switch2(config-vlan)# vlan 2	创建 VLAN 2
Switch2(config-vlan)# exit	退出 VLAN 配置模式
Switch2(config)# interface eth-0-11	进入 eth-0-11 的接口配置模式
Switch2(config-if)# switchport access vlan 2	将接口加入 VLAN 2
Switch2(config-if)# no shutdown	使能接口
Switch2(config-if)# interface vlan 2	进入 vlan2 的接口配置模式
Switch2(config-if)# ip address 10.10.10.2/24	配置 IP 地址
Switch2(config-if)# ip virtual-router address 10.10.10.254	配置虚拟 IP 地址
Switch2(config-if)# end	退出接口配置模式

16.12.4 命令验证

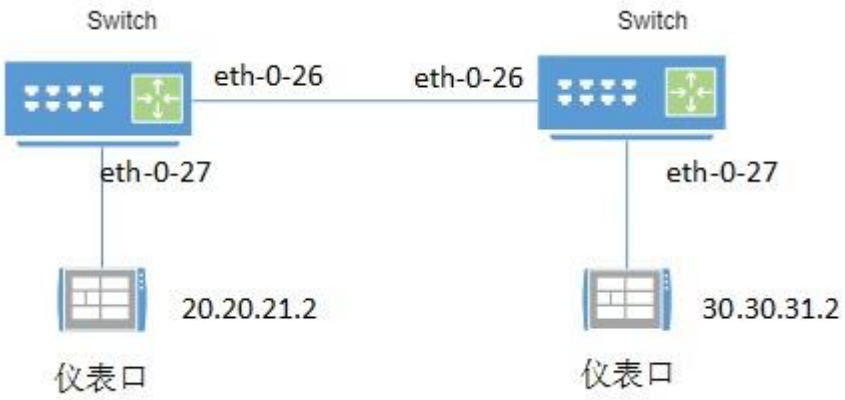
显示 ARP 表项的结果如下：

```
Switch1# show ip arp
 Protocol    Address          Age (min)  Hardware Addr   Interface
 Internet    10.10.10.1             -   cef0.12da.8100  vlan2
 Internet    10.10.10.254           -   000a.000a.000a  vlan2
Switch2#  show ip arp
 Protocol    Address          Age (min)  Hardware Addr   Interface
 Internet    10.10.10.2             -   66d1.4c26.e100  vlan2
 Internet    10.10.10.254           -   000a.000a.000a  vlan2
```

17 EVPN 配置指导

17.1 Inpsur 设备测试

17.1.1 拓扑



17.1.2 DUT1 配置

PS: border，需要引入外部路由（type5）

DUT1:

DUT1# configure terminal	进入全局配置模式
DUT1(config)# ip vrf test	创建 vrf
DUT1(config-vrf)# vni 50000 l3	创建 L3 VNI
DUT1(config-vrf)# rd 1:50000	配置 L3 VNI RD
DUT1(config-vrf)#route-target both 50:50000	配置 L3 VNI RT
DUT1(config-vrf)# exit	返回全局配置模式
DUT1(config)# vlan database	进入 vlan 配置模式
DUT1(config-vlan)# vlan 20,30,50	创建 vlan 20,50
DUT1(config-vlan)# vlan 20 overlay enable	使能 vlan 20 的 overlay 功能

DUT1(config-vlan)# vlan 30 overlay enable	使能 vlan 30 的 overlay 功能
DUT1(config-vlan)# vlan 50 overlay enable	使能 vlan 50 的 overlay 功能
DUT1(config-vlan)# exit	返回全局配置模式
DUT1(config)# overlay	进入 overlay 配置模式
DUT1(config-overlay)# source 1.1.1.1	配置 vxlan 的源 vtep 地址
DUT1(config-overlay)# vtep reachability protocol bgp	开启动态建 VxLAN 隧道功能
DUT1(config-overlay)# vlan 20 vni 20000	配置 vlan 和 vni 的映射
DUT1(config-overlay)# vlan 30 vni 30000	配置 vlan 和 vni 的映射
DUT1(config-overlay)# vlan 50 vni 50000	配置 vlan 和 vni 的映射
DUT1(config- overlay)# exit	返回全局配置模式
DUT1(config)# evpn	进入 EVPN 配置模式
DUT1(config-evpn)# vni 20000	创建 L2 VNI
DUT1(config-evi)# rd auto	配置自动生成 RD
DUT1(config-evi)# route-target both auto	配置自动生成 RT
DUT1(config-evi)# exit	返回 EVPN 配置模式
DUT1(config-evpn)# vni 30000	创建 EVPN 实例并进入 EVPN 实例配置模式
DUT1(config-evi)# rd auto	配置自动生成 RD
DUT1(config-evi)# route-target both auto	配置自动生成 RT
DUT1(config-evi)# exit	返回 EVPN 配置模式
DUT1(config-evpn)# exit	返回全局配置模式
DUT1(config)# interface eth-0-26	进入端口 eth-0-26 配置
DUT1(config-if)# no switchport	更改成路由端口
DUT1(config-if)# ip address 26.26.26.1/24	配置 ip 地址
DUT1(config-if)# overlay uplink enable	使能 overlay 的上联口
DUT1(config-if)# exit	返回全局配置模式
DUT1(config)# interface eth-0-27	进入接口 eth-0-27 的配置
DUT1(config-if)# switchport access vlan 20	将接口加入 vlan 20
DUT1(config)# interface vlan 20	进入接口 vlanif 20 的配置
DUT1(config-if)# ip vrf forwarding test	将端口加入 vrf 转发

DUT1(config-if)#overlay distributed-gateway enable	使能分布式网关
DUT1(config-if)# ip address 20.20.20.1/24	配置接口 vlanif 20 的地址
DUT1(config-if)# ip virtual-router address 20.20.21.1/24	配置接口 vlanif 的虚拟 IP 地址
DUT1(config-if)#overlay host-collect enable	使能主机信息搜集功能
DUT1(config-if)#exit	返回全局配置模式
DUT1(config)# interface vlan 30	进入接口 vlanif 30 的配置
DUT1(config-if)# ip vrf forwarding test	将端口加入 vrf 转发
DUT1(config-if)#overlay distributed-gateway enable	使能分布式网关
DUT1 (config-if)# ip address 30.30.30.1/24	配置接口 vlanif 30 的地址
DUT1(config-if)# ip virtual-router address 30.30.31.1/24	配置接口 vlanif 的虚拟 IP 地址
DUT1(config-if)#overlay host-collect enable	使能主机信息搜集功能
DUT1(config-if)#exit	返回全局配置模式
DUT1(config)# interface vlan 50	进入接口 vlanif 50 的配置
DUT1(config-if)# ip vrf forwarding test	将端口加入 vrf 转发
DUT1(config-if)# exit	返回全局配置模式
DUT1(config)# interface loopback 0	创建环回口
DUT1(config-if)# ip address 1.1.1.1/32	配置 ip 地址
DUT1(config-if)# exit	返回全局配置模式
DUT1(config)# router bgp 100	创建 BGP 100 并进入路由配置模式
DUT1(config)# bgp router-id 1.1.1.1	配置 BGP router-id
DUT1(config-router)# neighbor 2.2.2.2 remote-as 100	创建 IBGP 邻居
DUT1(config-router)# neighbor 2.2.2.2 update-source loopback0	指定更新源端口
DUT1(config- router)# address-family l2vpn evpn	进入 EVPN 地址族配置模式
DUT1(config- router-af)# neighbor 2.2.2.2 activate	使能与邻居交换路由信息
DUT1(config- router-af)# exit	返回路由配置模式
DUT1(config-router)# address-family ipv4 vrf test	进入 IPV4 VRF 地址族配置模式
DUT1(config-router-af)# redistribute connected	配置路由重发布

DUT1(config-router-af)# advertise l2vpn	配置重发布路由引入 EVPN
DUT1(config-router-af)# exit	返回路由配置模式
DUT1(config- router)# exit	返回全局配置模式
DUT1(config)# ip route 2.2.2.0/24 26.26.26.2	配置静态路由
DUT1(config)#ip virtual-router mac 0001.0001.0001	配置虚拟 mac

17.1.3 DUT2 配置

DUT2:

DUT2# configure terminal	进入全局配置模式
DUT2(config)# ip vrf test	创建 vrf
DUT2(config-vrf)# vni 50000 l3	创建 L3 VNI
DUT2(config-vrf)# rd 1:50000	配置 L3 VNI RD
DUT2(config-vrf)#route-target both 50:50000	配置 L3 VNI RT
DUT2(config)# vlan database	进入 vlan 配置模式
DUT2(config-vlan)# vlan 20,30,50	创建 vlan 20,50
DUT2(config-vlan)# vlan 20 overlay enable	使能 vlan 20 的 overlay 功能
DUT2(config-vlan)# vlan 30 overlay enable	使能 vlan 30 的 overlay 功能
DUT2(config-vlan)# vlan 50 overlay enable	使能 vlan 50 的 overlay 功能
DUT2(config-vlan)# exit	返回全局配置模式
DUT2(config)# overlay	进入 overlay 配置模式
DUT2(config-overlay)# source 2.2.2.2	配置 vxlan 的源 vtep 地址
DUT2(config-overlay)# vtep reachability protocol bgp	开启动态建 VxLAN 隧道功能
DUT2(config-overlay)# vlan 20 vni 20000	配置 vlan 和 vni 的映射
DUT2(config-overlay)# vlan 30 vni 30000	配置 vlan 和 vni 的映射
DUT2(config-overlay)# vlan 50 vni 50000	配置 vlan 和 vni 的映射
DUT2(config- overlay)# exit	返回全局配置模式
DUT2(config)# evpn	进入 EVPN 配置模式
DUT2(config-evpn)# vni 20000	创建 EVPN 实例并进入 EVPN 实例配置模式
DUT2(config-evi)# rd auto	配置自动生成 RD

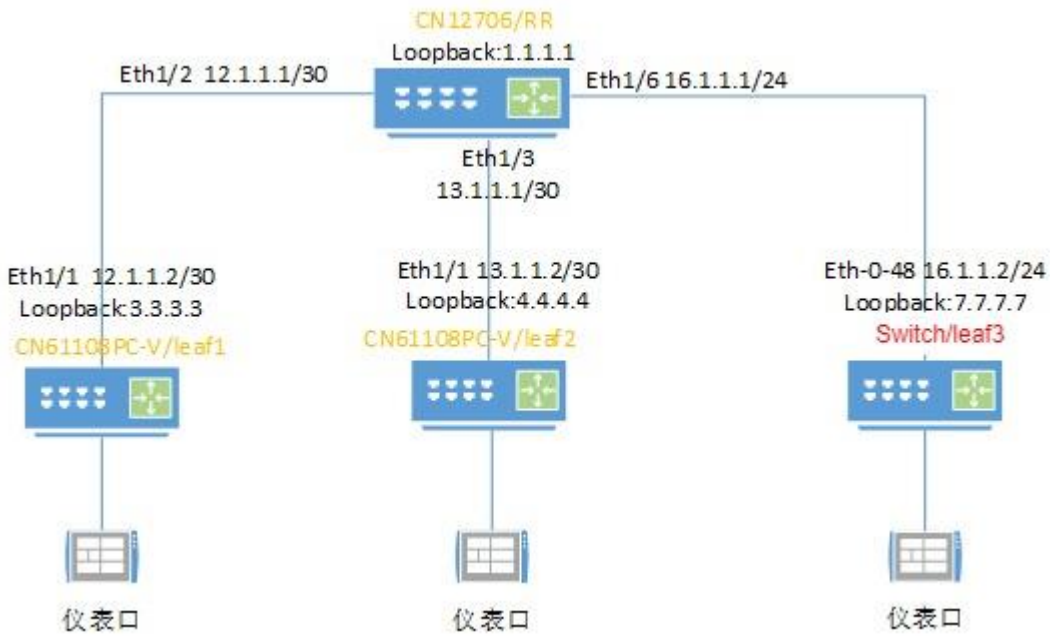
DUT2(config-evi)# route-target both auto	配置自动生成 RT
DUT2(config-evi)# exit	返回 EVPN 配置模式
DUT2(config-evpn)# vni 30000	创建 EVPN 实例并进入 EVPN 实例配置模式
DUT2(config-evi)# rd auto	配置自动生成 RD
DUT2(config-evi)# route-target both auto	配置自动生成 RT
DUT2(config-evi)# exit	返回 EVPN 配置模式
DUT2(config-evpn)# exit	返回全局配置模式
DUT2(config)# interface eth-0-26	进入端口 eth-0-26 配置
DUT2(config-if)# no switchport	更改成路由端口
DUT2(config-if)# ip address 26.26.26.1/24	配置 ip 地址
DUT2(config-if)# overlay uplink enable	使能 overlay 的上联口
DUT2(config-if)# exit	返回全局配置模式
DUT2(config)# interface eth-0-27	进入接口 eth-0-27 的配置
DUT2(config-if)# switchport access vlan 30	将接口加入 vlan 30
DUT2(config-if)#exit	返回全局配置模式
DUT2 (config)# interface vlan 20	进入接口 vlanif 20 的配置
DUT2(config-if)# ip vrf forwarding test	将端口加入 vrf 转发
DUT2(config-if)#overlay distributed-gateway enable	使能分布式网关
DUT2(config-if)# ip address 20.20.20.1/24	配置接口 vlanif 20 的地址
DUT2(config-if)# ip virtual-router address 20.20.21.1/24	配置接口 vlanif 的虚拟 IP 地址
DUT2(config-if)#overlay host-collect enable	使能主机信息搜集功能
DUT2(config-if)#exit	返回全局配置模式
DUT2 (config)# interface vlan 30	进入接口 vlanif 30 的配置
DUT2(config-if)# ip vrf forwarding test	将端口加入 vrf 转发
DUT2(config-if)#overlay distributed-gateway enable	使能分布式网关
DUT2(config-if)# ip address 30.30.30.1/24	配置接口 vlanif 30 的地址
DUT2(config-if)# ip virtual-router address 30.30.31.1/24	配置接口 vlanif 的虚拟 IP 地址
DUT2(config-if)#overlay host-collect enable	使能主机信息搜集功能

DUT2(config-if)# ip vrf forwarding test	将端口加入 vrf 转发
DUT2(config-if)#exit	返回全局配置模式
DUT2 (config)# interface vlan 50	进入接口 vlanif 50 的配置
DUT2(config-if)# ip vrf forwarding test	将端口加入 vrf 转发
DUT2 (config-if)# exit	返回全局配置模式
DUT2 (config)# interface loopback 0	创建环回口
DUT2 (config-if)# ip address 2.2.2.2/32	配置 ip 地址
DUT2 (config-if)# exit	返回全局配置模式
DUT2 (config)# router bgp 100	创建 BGP 100 并进入路由配置模式
DUT2 (config)# bgp router-id 2.2.2.2	配置 BGP router-id
DUT2 (config-router)# neighbor 1.1.1.1 remote-as 100	创建 IBGP 邻居
DUT2 (config-router)# neighbor 1.1.1.1 update-source loopback0	指定更新源端口
DUT2 (config- router)# address-family l2vpn evpn	进入 EVPN 地址族配置模式
DUT2 (config- router-af)# neighbor 1.1.1.1 activate	使能与邻居交换路由信息
DUT2(config- router-af)# exit	返回路由配置模式
DUT2(config- router)# exit	返回全局配置模式
DUT2 (config)# ip route 1.1.1.0/24 26.26.26.1	配置静态路由
DUT2(config)#ip virtual-router mac 0001.0001.0001	配置虚拟 mac

17.2 inpsur 与思科设备对接测试

17.2.1 有 RR 的测试用例

17.2.2 拓扑



CN12706 和 CN61108PC-V 为思科设备，CN12706 作为路由反射器，Switch 为 inspur 设备。

17.2.3 RR 的配置

RR# configure terminal	进入全局配置模式
RR(config)# nv overlay evpn	为 VXLAN 启用 EVPN 控制平面
RR(config)# feature bgp	Enable bgp
RR(config)# feature ospf	Enable ospf
RR(config)# feature interface-vlan	启用 interface vlan
RR(config)# feature vn-segment-vlan-based	Enable VLAN-based VXLAN
RR(config)# feature nv overlay	Enable vxlan
RR(config)# router ospf 1	启用 ospf
RR(config)# interface ethernet 1/2	进入端口 ethernet 1/2 配置
RR(config)# no switchport	更改成路由端口
RR(config-if)# ip address 12.1.1.1/30	配置 ip 地址
RR(config-if)# ip router ospf 1 area 0.0.0.0	端口上启用 ospf 协议

RR(config-if)#no shutdown	开启端口
RR(config-if)# exit	返回全局配置模式
RR (config)# interface ethernet 1/3	进入端口 ethernet 1/3 配置
RR(config)# no switchport	更改成路由端口
RR(config-if)# ip address 13.1.1.1/30	配置 ip 地址
RR(config-if)# ip router ospf 1 area 0.0.0.0	端口上启用 ospf 协议
RR(config-if)#no shutdown	开启端口
RR(config-if)# exit	返回全局配置模式
RR (config)# interface ethernet 1/6	进入端口 ethernet 1/3 配置
RR(config)# no switchport	更改成路由端口
RR(config-if)# ip address 16.1.1.1/30	配置 ip 地址
RR(config-if)# ip router ospf 1 area 0.0.0.0	端口上启用 ospf 协议
RR(config-if)#no shutdown	开启端口
RR(config-if)# exit	返回全局配置模式
RR(config)# interface loopback 1	创建环回口
RR(config-if)# ip address 1.1.1.1/32	配置 ip 地址
RR(config-if)# ip router ospf 1 area 0.0.0.0	端口上启用 ospf 协议
RR(config-if)# exit	返回全局配置模式
RR(config)# router bgp 65101	创建 BGP 65101 并进入路由配置模式
RR(config-router)# router-id 1.1.1.1	配置 router ID
RR(config- router)# address-family ipv4 unicast	进入 ipv4 unicast 地址族配置模式
RR(config-router)# address-family l2vpn evpn	进入 l2vpn evpn 地址族配置模式
RR(config- router-af)# retain route-target all	保留路由目标属性
RR(config-router)# template peer VTEP	iBGP peer template
RR(config-router-neighbor)# remote-as 65101	邻居 AS
RR(config-router-neighbor)# update-source loopback1	更新源端口
RR(config-router-neighbor)# address-family ipv4 unicast	进入 ipv4 unicast 地址族配置模式
RR(config-router-neighbor-af)# send-community	地址族中的发送团体名
RR(config-router-neighbor-af)# send-community extended	地址族中的发送扩展团体名

RR(config-router-neighbor-af)# route-reflector-client	启用 RR
RR(config-router-neighbor-af)#exit	返回邻居配置模式
RR(config-router-neighbor)# address-family l2vpn evpn	进入 l2vpn evpn 地址族配置模式
RR(config-router-neighbor-af)# send-community	地址族中的发送团体名
RR(config-router-neighbor-af)# send-community extended	地址族中的发送扩展团体名
RR(config-router-neighbor-af)# exit	返回邻居配置模式
RR(config-router-neighbor)# exit	返回路由配置模式
RR(config-router)# neighbor 3.3.3.3	创建 IBGP 邻居
RR(config-router-neighbor)# inherit peer VTEP	用 peer 模板建立邻居
RR(config-router-neighbor)# exit	返回路由配置模式
RR(config-router)# neighbor 4.4.4.4	创建 IBGP 邻居
RR(config-router-neighbor)# inherit peer VTEP	用 peer 模板建立邻居
RR(config-router-neighbor)# exit	返回路由配置模式
RR(config-router)# neighbor 7.7.7.7	创建 IBGP 邻居
RR(config-router-neighbor)# inherit peer VTEP	用 peer 模板建立邻居

17.2.4 leaf1 的配置

LEAF1 # configure terminal	进入全局配置模式
LEAF1(config)# nv overlay evpn	为 VXLAN 启用 EVPN 控制平面
LEAF1(config)# feature bgp	Enable bgp
LEAF1(config)# feature ospf	Enable ospf
LEAF1(config)# feature interface-vlan	启用 interface vlan
LEAF1(config)# feature vn-segment-vlan-based	Enable VLAN-based VXLAN
LEAF1(config)# feature nv overlay	Enable vxlan
LEAF1(config)# router ospf 1	启用 ospf 协议
LEAF1(config)# vlan 100,200,300	创建 vlan
LEAF1(config-vlan)#exit	返回全局配置模式
LEAF1(config)# vlan 100	进入 vlan 100
LEAF1(config-vlan)# vn-segment 100	创建 VLAN 与 VNI 的映射
LEAF1(config-vlan)#exit	返回全局配置模式

LEAF1(config)# vlan 200	进入 vlan 200
LEAF1(config-vlan)# vn-segment 200	创建 VLAN 与 VNI 的映射
LEAF1(config-vlan)#exit	返回全局配置模式
LEAF1(config)# vlan 300	进入 vlan 300
LEAF1(config-vlan)# vn-segment 300	创建 VLAN 与 VNI 的映射
LEAF1(config)#vrf context evpn-tenant-1	创建 vrf
LEAF1(config-vrf)# vni 300	创建 L3 vni
LEAF1(config-vrf)# rd 1:300	配置 RD
LEAF1(config-vrf)# address-family ipv4 unicast	进入 ipv4 unicast 地址族配置模式
LEAF1(config-vrf-af-ipv4)#route-target import 300:300	配置 RT
LEAF1(config-vrf-af-ipv4)#route-target export 300:300	配置 RT
LEAF1(config-vrf-af-ipv4)# interface Vlan100	进入 interface vlan
LEAF1(config-if)# no shutdown	开启端口
LEAF1(config-if)# vrf member evpn-tenant-1	加入 vrf
LEAF1(config-if)# ip address 10.1.1.1/24	配置 IP 地址
LEAF1(config-if)#fabric forwarding mode anycast-gateway	启用分布式任播网关
LEAF1(config-if)# interface Vlan200	进入 interface vlan
LEAF1(config-if)# no shutdown	开启端口
LEAF1(config-if)# vrf member evpn-tenant-1	加入 vrf
LEAF1(config-if)# ip address 20.1.1.1/24	配置 IP 地址
LEAF1(config-if)#fabric forwarding mode anycast-gateway	启用分布式任播网关
LEAF1(config-if)# interface Vlan300	进入 interface vlan
LEAF1(config-if)# no shutdown	开启端口
LEAF1(config-if)# vrf member evpn-tenant-1	加入 vrf
LEAF1(config-if)# ip forward	Forward 到 vrf
LEAF1(config-if)# interface nve1	进入 interface nve
LEAF1(config-if-nve)# no shutdown	开启端口
LEAF1(config-if-nve)#host-reachability protocol bgp	设置主机路由可达协议为 bgp
LEAF1(config-if-nve)#source-interface loopback1	设置 source 的端口

LEAF1(config-if-nve)# member vni 100	L2 vni 与 nve 关联
LEAF1(config-if-nve-vni)# ingress-replication protocol bgp	启用头端复制
LEAF1(config-if-nve)# member vni 200	L2 vni 与 nve 关联
LEAF1(config-if-nve-vni)# ingress-replication protocol bgp	启用头端复制
LEAF1(config-if-nve)#member vni 300 associate-vrf	添加 L3 vni
LEAF1(config-if-nve-vni)# interface Ethernet1/1	进入端口 Ethernet1/1
LEAF1(config-if)# no switchport	更改成路由端口
LEAF1(config-if)# ip address 12.1.1.2/30	配置 IP
LEAF1(config-if)# ip router ospf 1 area 0.0.0.0	端口上启用 ospf 协议
LEAF1(config-if)# no shutdown	开启端口
LEAF1(config-if)# interface Ethernet1/6	进入端口 Ethernet1/6
LEAF1(config-if)# switchport access vlan 100	加入 vlan100
LEAF1(config-if)# interface Ethernet1/7	进入端口 Ethernet1/7
LEAF1(config-if)# switchport access vlan 200	加入 vlan200
LEAF1(config-if)# interface loopback1	创建环回口
LEAF1(config-if)# ip address 3.3.3.3/32	配置 ip 地址
LEAF1(config-if)# ip router ospf 1 area 0.0.0.0	端口上启用 ospf 协议
LEAF1(config-if)# router bgp 65101	创建 BGP 65101 并进入路由配置模式
LEAF1(config-router)# router-id 3.3.3.3	配置 router ID
LEAF1(config-router)# template peer LEAF1	iBGP peer template
LEAF1(config-router-neighbor)# remote-as 65101	邻居 AS
LEAF1(config-router-neighbor)# update-source loopback1	更新源端口
LEAF1(config-router-neighbor)# address-family ipv4 unicast	进入 ipv4 unicast 地址族配置模式
LEAF1(config-router-neighbor-af)# send-community	地址族中的发送团体名
LEAF1(config-router-neighbor-af)# send-community extended	地址族中的发送扩展团体名

LEAF1(config-router-neighbor-af)#exit	返回邻居配置模式
LEAF1(config-router-neighbor)# address-family l2vpn evpn	进入 l2vpn evpn 地址族配置模式
LEAF1(config-router-neighbor-af)# send-community	地址族中的发送团体名
LEAF1(config-router-neighbor-af)# send-community extended	地址族中的发送扩展团体名
LEAF1(config-router-neighbor-af)#exit	返回邻居配置模式
LEAF1(config-router-neighbor)#exit	返回路由配置模式
LEAF1(config-router)# neighbor 1.1.1.1	创建 IBGP 邻居
LEAF1(config-router-neighbor)# inherit peer LEAF1	用 peer 模板建立邻居
LEAF1(config-router)# evpn	进入 evpn 配置模式
LEAF1(config-evpn)# vni 100 l2	创建 L2 vni
LEAF1(config-evpn-evi)# rd 100:100	配置 RD
LEAF1(config-evpn-evi)# route-target import 100:100	配置 RT
LEAF1(config-evpn-evi)# route-target export 100:100	配置 RT
LEAF1(config-evpn-evi)#exit	返回 evpn 配置模式
LEAF1(config-evpn)# vni 200 l2	创建 L2 vni
LEAF1(config-evpn-evi)# rd 200:200	配置 RD
LEAF1(config-evpn-evi)# route-target import 200:200	配置 RT
LEAF1(config-evpn-evi)# route-target export 200:200	配置 RT

17.2.5 leaf2 的配置

LEAF2 # configure terminal	进入全局配置模式
LEAF2(config)# nv overlay evpn	为 VXLAN 启用 EVPN 控制平面
LEAF2(config)# feature bgp	Enable bgp
LEAF2(config)# feature ospf	Enable ospf
LEAF2(config)# feature interface-vlan	启用 interface vlan
LEAF2(config)# feature vn-segment-vlan-based	Enable VLAN-based VXLAN
LEAF2(config)# feature nv overlay	Enable vxlan
LEAF2(config)# router ospf 1	启用 ospf 协议
LEAF2(config)# vlan 100,200,300	创建 vlan
LEAF2(config-vlan)#exit	返回全局配置模式

LEAF2(config)# vlan 100	进入 vlan 100
LEAF2(config-vlan)# vn-segment 100	创建 VLAN 与 VNI 的映射
LEAF2(config-vlan)#exit	返回全局配置模式
LEAF2(config)# vlan 200	进入 vlan 200
LEAF2(config-vlan)# vn-segment 200	创建 VLAN 与 VNI 的映射
LEAF2(config-vlan)#exit	返回全局配置模式
LEAF2(config)# vlan 300	进入 vlan 300
LEAF2(config-vlan)# vn-segment 300	创建 VLAN 与 VNI 的映射
LEAF2(config)#vrf context evpn-tenant-1	创建 vrf
LEAF2(config-vrf)# vni 300	创建 L3 vni
LEAF2(config-vrf)# rd 1:300	配置 RD
LEAF2(config-vrf)# address-family ipv4 unicast	进入 ipv4 unicast 地址族配置模式
LEAF2(config-vrf-af-ipv4)#route-target import 300:300	配置 RT
LEAF2(config-vrf-af-ipv4)#route-target export 300:300	配置 RT
LEAF2(config-vrf-af-ipv4)# interface Vlan100	进入 interface vlan
LEAF2(config-if)# no shutdown	开启端口
LEAF2(config-if)# vrf member evpn-tenant-1	加入 vrf
LEAF2(config-if)# ip address 10.1.1.1/24	配置 IP 地址
LEAF2(config-if)#fabric forwarding mode anycast-gateway	启用分布式任播网关
LEAF2(config-if)# interface Vlan200	进入 interface vlan
LEAF2(config-if)# no shutdown	开启端口
LEAF2(config-if)# vrf member evpn-tenant-1	加入 vrf
LEAF2(config-if)# ip address 20.1.1.1/24	配置 IP 地址
LEAF2(config-if)#fabric forwarding mode anycast-gateway	启用分布式任播网关
LEAF2(config-if)# interface Vlan300	进入 interface vlan
LEAF2(config-if)# no shutdown	开启端口
LEAF2(config-if)# vrf member evpn-tenant-1	加入 vrf
LEAF2(config-if)# ip forward	Forward 到 vrf
LEAF2(config-if)# interface nve1	进入 interface nve

LEAF2(config-if-nve)# no shutdown	开启端口
LEAF2(config-if-nve)#host-reachability protocol bgp	设置主机路由可达协议为 bgp
LEAF2(config-if-nve)#source-interface loopback1	设置 source 的端口
LEAF2(config-if-nve)# member vni 100	L2 vni 与 nve 关联
LEAF2(config-if-nve-vni)# ingress-replication protocol bgp	启用头端复制
LEAF2(config-if-nve)# member vni 200	L2 vni 与 nve 关联
LEAF2(config-if-nve-vni)# ingress-replication protocol bgp	启用头端复制
LEAF2(config-if-nve)#member vni 300 associate-vrf	添加 L3 vni
LEAF2(config-if-nve-vni)# interface Ethernet1/1	进入端口 Ethernet1/1
LEAF2(config-if)# no switchport	更改成路由端口
LEAF2(config-if)# ip address 13.1.1.2/30	配置 IP
LEAF2(config-if)# ip router ospf 1 area 0.0.0.0	端口上启用 ospf 协议
LEAF2(config-if)# no shutdown	开启端口
LEAF2(config-if)# interface Ethernet1/6	进入端口 Ethernet1/6
LEAF2(config-if)# switchport access vlan 100	加入 vlan100
LEAF2(config-if)# interface Ethernet1/7	进入端口 Ethernet1/7
LEAF2(config-if)# switchport access vlan 200	加入 vlan200
LEAF2(config-if)# interface loopback1	创建环回口
LEAF2(config-if)# ip address 4.4.4.4/32	配置 ip 地址
LEAF2(config-if)# ip router ospf 1 area 0.0.0.0	端口上启用 ospf 协议
LEAF2(config-if)# router bgp 65101	创建 BGP 65101 并进入路由配置模式
LEAF2(config-router)# router-id 4.4.4.4	配置 router ID
LEAF2(config-router)# template peer LEAF2	iBGP peer template
LEAF2(config-router-neighbor)# remote-as 65101	邻居 AS
LEAF2(config-router-neighbor)# update-source loopback1	更新源端口
LEAF2(config-router-neighbor)# address-family ipv4 unicast	进入 ipv4 unicast 地址族配置模式

LEAF2(config-router-neighbor-af)# send-community	地址族中的发送团体名
LEAF2(config-router-neighbor-af)# send-community extended	地址族中的发送扩展团体名
LEAF2(config-router-neighbor-af)#exit	返回邻居配置模式
LEAF2(config-router-neighbor)# address-family l2vpn evpn	进入 l2vpn evpn 地址族配置模式
LEAF2(config-router-neighbor-af)# send-community	地址族中的发送团体名
LEAF2(config-router-neighbor-af)# send-community extended	地址族中的发送扩展团体名
LEAF2(config-router-neighbor-af)# exit	返回邻居配置模式
LEAF2(config-router-neighbor)#exit	返回路由配置模式
LEAF2(config-router)# neighbor 1.1.1.1	创建 IBGP 邻居
LEAF2(config-router-neighbor)# inherit peer LEAF2	用 peer 模板建立邻居
LEAF2(config-router)# evpn	进入 evpn 配置模式
LEAF2(config-evpn)# vni 100 l2	创建 L2 vni
LEAF2(config-evpn-evi)# rd 100:100	配置 RD
LEAF2(config-evpn-evi)# route-target import 100:100	配置 RT
LEAF2(config-evpn-evi)# route-target export 100:100	配置 RT
LEAF2(config-evpn-evi)#exit	返回 evpn 配置模式
LEAF2(config-evpn)# vni 200 l2	创建 L2 vni
LEAF2(config-evpn-evi)# rd 200:200	配置 RD
LEAF2(config-evpn-evi)# route-target import 200:200	配置 RT
LEAF2(config-evpn-evi)# route-target export 200:200	配置 RT

17.2.6 leaf3 的配置

LEAF3 # configure terminal	进入全局配置模式
LEAF3(config)# ip vrf evpn-tenant-1	创建 vrf
LEAF3(config-vrf)# vni 300 l3	创建 L3 VNI
LEAF3(config-vrf)# rd 1:300	配置 L3 VNI RD
LEAF3(config-vrf)#route-target both 300:300	配置 L3 VNI RT
LEAF3(config-vrf)#exit	返回全局配置模式

LEAF3(config)# vlan database	进入 VLAN 配置模式。
LEAF3(config-vlan)# vlan 100,200,300	创建 vlan
LEAF3(config-vlan)# vlan 100 overlay enable	使能 vlan 的 overlay 功能
LEAF3(config-vlan)# vlan 200 overlay enable	使能 vlan 的 overlay 功能
LEAF3(config-vlan)# vlan 300 overlay enable	使能 vlan 的 overlay 功能
LEAF3(config-vlan)# exit	返回全局配置模式
LEAF3(config)# overlay gateway enable	使能 overlay 的 gateway 增强功能
LEAF3(config)# overlay	进入 overlay 配置模式。
LEAF3(config-overlay)# source 7.7.7.7	配置 vxlan 的源 vtep 地址
LEAF3 (config-overlay)# vtep reachability protocol bgp	开启动态建 VxLAN 隧道功能
LEAF3(config-overlay)# vlan 100 vni 100	配置 vlan 和 vni 的映射
LEAF3(config-overlay)# vlan 200 vni 200	配置 vlan 和 vni 的映射
LEAF3(config-overlay)# vlan 300 vni 300	配置 vlan 和 vni 的映射
LEAF3(config)# evpn	进入 evpn 配置模式
LEAF3(config-evpn)# vni 100	创建 L2 VNI
LEAF3(config-evpn-evi)# rd 100:100	配置 L2 VNI RD
LEAF3(config-evpn-evi)# route-target import 100:100	配置 L2 VNI RT
LEAF3(config-evpn-evi)# route-target export 100:100	配置 L2 VNI RT
LEAF3(config-evpn-evi)# exit	返回 evpn 配置模式
LEAF3(config-evpn)# vni 200	创建 L2 VNI
LEAF3(config-evpn-evi)# rd 200:200	配置 L2 VNI RD
LEAF3(config-evpn-evi)# route-target import 200:200	配置 L2 VNI RT
LEAF3(config-evpn-evi)# route-target export 200:200	配置 L2 VNI RT
LEAF3(config-evpn-evi)# exit	返回 evpn 配置模式
LEAF3(config-evpn)# exit	返回全局配置模式
LEAF3(config)# interface eth-0-48	进入端口 eth-0-48 配置
LEAF3(config-if)# no switchport	更改成路由端口
LEAF3(config-if)# overlay uplink enable	配置 ip 地址
LEAF3(config-if)# ip address 16.1.1.2/24	使能 overlay 的上联口
LEAF3(config-if)# exit	返回全局配置模式
LEAF3(config)# interface vlan100	进入端口 vlan100
LEAF3(config-if)# ip vrf forwarding evpn-tenant-1	将端口加入 vrf 转发

LEAF3(config-if)# overlay distributed-gateway enable	使能分布式网关
LEAF3(config-if)# overlay host-collect enable	使能主机信息搜集功能
LEAF3(config-if)# ip address 10.3.1.1/31	配置接口 vlanif 的 IP 地址
LEAF3(config-if)#ip virtual-router address 10.1.1.1/24	配置接口 vlanif 的虚拟 IP 地址
LEAF3(config-if)# exit	返回全局配置模式
LEAF3(config)# interface vlan200	进入端口 vlan105
LEAF3(config-if)# ip vrf forwarding test	将端口加入 vrf 转发
LEAF3(config-if)#overlay distributed-gateway enable	使能分布式网关
LEAF3(config-if)# overlay host-collect enable	使能主机信息搜集功能
LEAF3(config-if)# ip address 20.3.1.1/24	配置接口 vlanif 的 IP 地址
LEAF3(config-if)#ip virtual-router address 20.1.1.1/24	配置接口 vlanif 的虚拟 IP 地址
LEAF3(config-if)# exit	返回全局配置模式
LEAF3(config)# interface loopback0	创建环回口
LEAF3(config-if)# ip address 7.7.7.7/32	配置 ip 地址
LEAF3(config-if)# exit	返回全局配置模式
LEAF3(config)#ip virtual-router mac 0001.0001.0001	配置虚拟 mac
LEAF3(config-if)# router ospf 1	启用 ospf 协议
LEAF3(config-router)# network 7.7.7.7 0.0.0.0 area 0	宣告网段
LEAF3(config-router)# network 16.1.1.0 0.0.0.3 area 0	宣告网段
LEAF3(config-if)# router bgp 65101	创建 BGP 65101 并进入路由配置模式
LEAF3(config-router)# neighbor 1.1.1.1 remote-as 65101	创建 IBGP 邻居
LEAF3(config-router)# neighbor 1.1.1.1 update-source loopback0	指定更新源端口
LEAF3(config-router)# address-family l2vpn evpn	进入 l2vpn evpn 地址族配置模式
LEAF3(config-router-af)# neighbor 1.1.1.1 activate	使能与邻居交换路由信息
LEAF3(config-router-af)# exit	返回路由配置模式
LEAF3 (config-router)# address-family ipv4 vrf test	进入 IPV4 VRF 地址族配置模式
LEAF3 (config-router-af)# redistribute connected	配置路由重发布
LEAF3 (config-router-af)# advertise l2vpn	配置重发布路由引入 EVPN
LEAF3 (config-router-af)# end	返回用户模式