第 1 章 实验拓扑、终端服务器配置

本章将首先简要介绍如何从计算机上访问路由器以对它们进行配置，通常可以通过 console 口或者 telnet 来连接路由器。随后还要介绍本书中一直要用到的网络拓扑，并将详细介绍如何配置终端服务器以达到方便控制各个路由器和交换机的目的。

1.1 访问 CISCO 路由器的方法

路由器没有键盘和鼠标，要初始化路由器需要把计算机的串口和路由器的 console 口进行连接。访问 CISCO 路由器的方法还有 telnet、web browser、网管软件（例如 CISCO Works）等，本节讨论前 2 种。

1.1.1 通常 console 口访问路由器

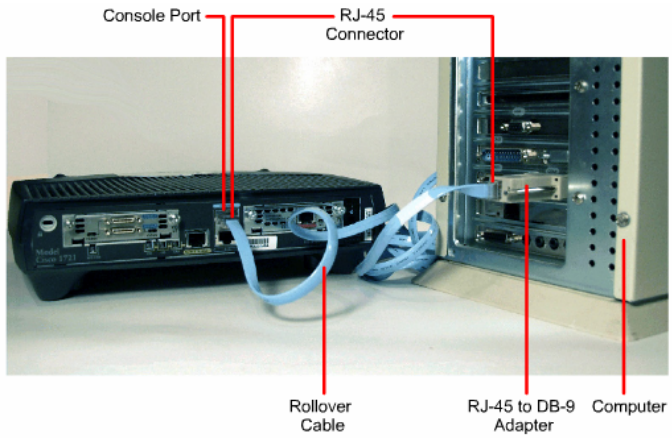

图 1-1 计算机和路由器通过 roll over 线进行连接

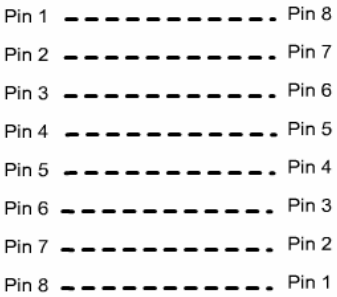

图 1-2 反转线的线序

计算机的串口和路由器的 console 口是通过反转线（roll over）进行连接的，反转线的一端接在路由器的 console 口上，另一端接到一个 DB9-RJ45 的转接头上，DB9 则接到计算机的串口上，如图 1-1。所谓的反转线就是线两端的 RJ45 接头上的线序是反的，如图 1-2。计算机和路由器连接好后，可以使用各种各样的终端软件配置路由器了。

1.1.2 通过 telnet 访问路由器

如果管理员不在路由器跟前，可以通过 telnet 远程配置路由器，当然这需要预先在路

由器上配置了 IP 地址和密码，并保证管理员的计算机和路由器之间是 IP 可达的（简单讲就是能ping通）。CISCO路由器通常支持多人同时telnet，每一个用户称为一个虚拟终端(VTY)。第一个用户为 vty 0，第二个用户为 vty 1，依次类推，路由器通常达 vty 4。

1.1.3 终端访问服务器

稍微复杂一点的实验就会用到多台路由器或者交换机，如果通过计算机的串口和它们连接，就需要经常性拔插 console 线。终端访问服务器可以解决这个问题，连接图如图 1-3。终端访问服务器实际上就是有 8 个或者 16 个异步口的路由器，从它引出多条连接线到各个路由器上的 console 口。使用时，首先登录到终端访问服务器，然后从终端访问服务器再登录到各个路由器。

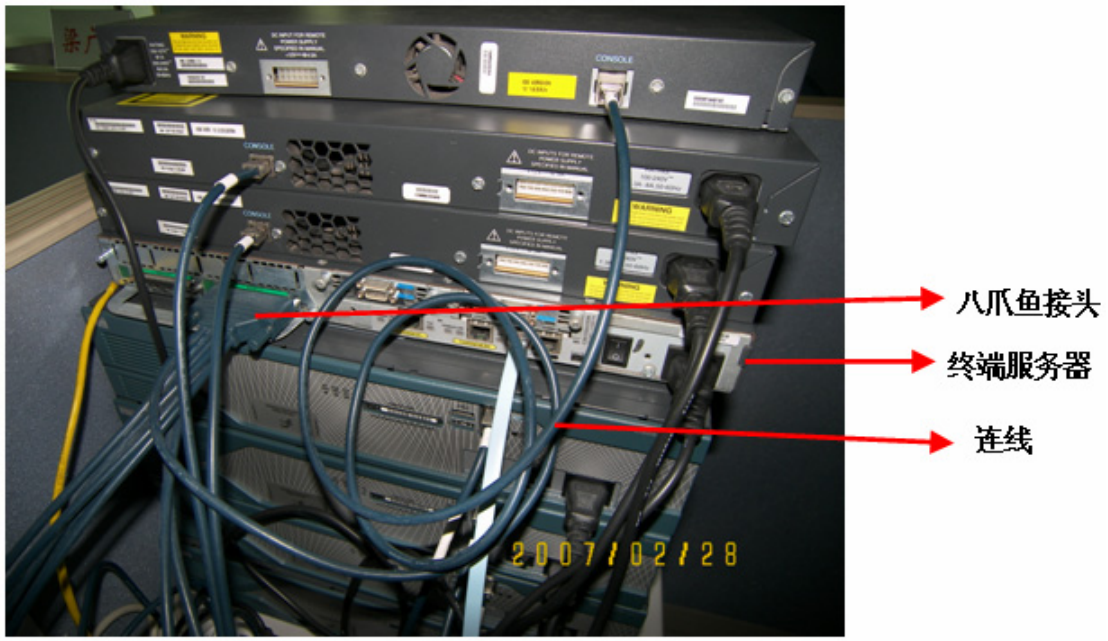

图 1-3 终端访问服务器和路由器的连接方法

1.1.4 本书实验拓扑

为了完成各种实验，需要构建不同的拓扑，这花费大量的时间。我们设计了一个功能强大的网络拓扑，如图 1-4（图中不包含显示终端服务器和它们的连接），本书所有的实验均可以使用该拓扑完成；该拓扑还可以满足 CCNA 和 CCNP 的绝大多数实验、以及 CCIE 的部分实验。拓扑中的路由器和交换机均通过终端访问服务器来进行控制，每个拓扑可以满足 1—7 人共同操作。

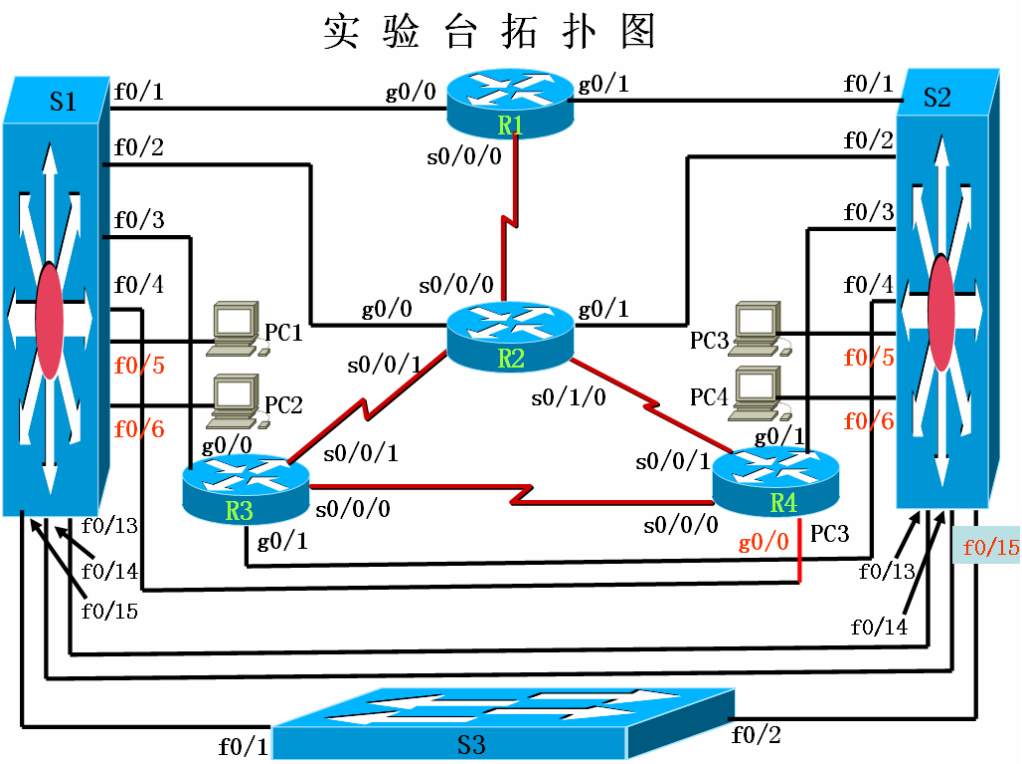

图 1-4 本书实验拓扑

拓扑中 4 台路由器均为 CISCO2821 路由器，也可以采用 CISCO2801 路由器（差别在于 CISCO2821 的以太网接口为千兆口，而 CISCO2801 的以太网接口为百兆口），IOS 采用 c2800nm-adventerprisek9-mz.124-11.T1.bin ；S1 和 S2 交换机为 Catalyst 3560， IOS 采用 c3560-ipbasek9-mz.122-25.SEB4.bin ； S3 为 Catalyst 2950 ， IOS 采用 c2950-i6q412-mz.121-6.EA2c.bin 。

拓扑中，4 台路由器之间通过串行链路进行连接。同时所有路由器的 g0/0 以太网接口和交换机 S1 进行连接；g0/1 以太网接口则和交换机 S2 进行连接。S1 和 S2 交换机之间通过 f0/13 和 f0/14 进行连接；S3 交换机的 f0/1 接口连接到 S1 的 f0/15 上，f0/2 接口连接到 S2 的 f0/15 上。计算机 PC1 和 PC2 连接到 S1 交换机的 f0/5 和 f0/6 上；计算机 PC3 和 PC4 则连接到 S2 交换机的 f0/5 和 f0/6 上。

图中的计算机应该有 2 个网卡（图中没有画出），其中一个网卡和终端服务器连接，另一网卡和图 1-4 中的交换机连接。

终端服务器可以采用 CISCO2509 或者带有 8 个或者 16 个异步模块的路由器。

1.2 实验 1：通过 console 口访问路由器

1. 实验目的

通过本实验，读者可以掌握如下技能：

(1) 计算机的串口和路由器 console 口的连接方法；

(2) 使用 Windows 系统自带的超级终端软件配置路由器；

(3) 路由器的开机。

2. 实验拓扑

如图 1-1。

3. 实验步骤

（1） 步骤 1：如图 1-1，连接好计算机 COM 1 口和路由器的 CONSOLE 口，路由器开机

（2） 步骤 2：打开超级终端

在 Windows 中的【开始】→【程序】→【附件】→【通信】菜单下打开“超级终端”程序，出现图 1-5 窗口。在“名称”对话框中输入名称，例如“Router”；按【确定】按钮。出现图 1-6 窗口时，在“连接时使用”下拉菜单中选择计算机的 COM 1 口，按【确定】按钮。

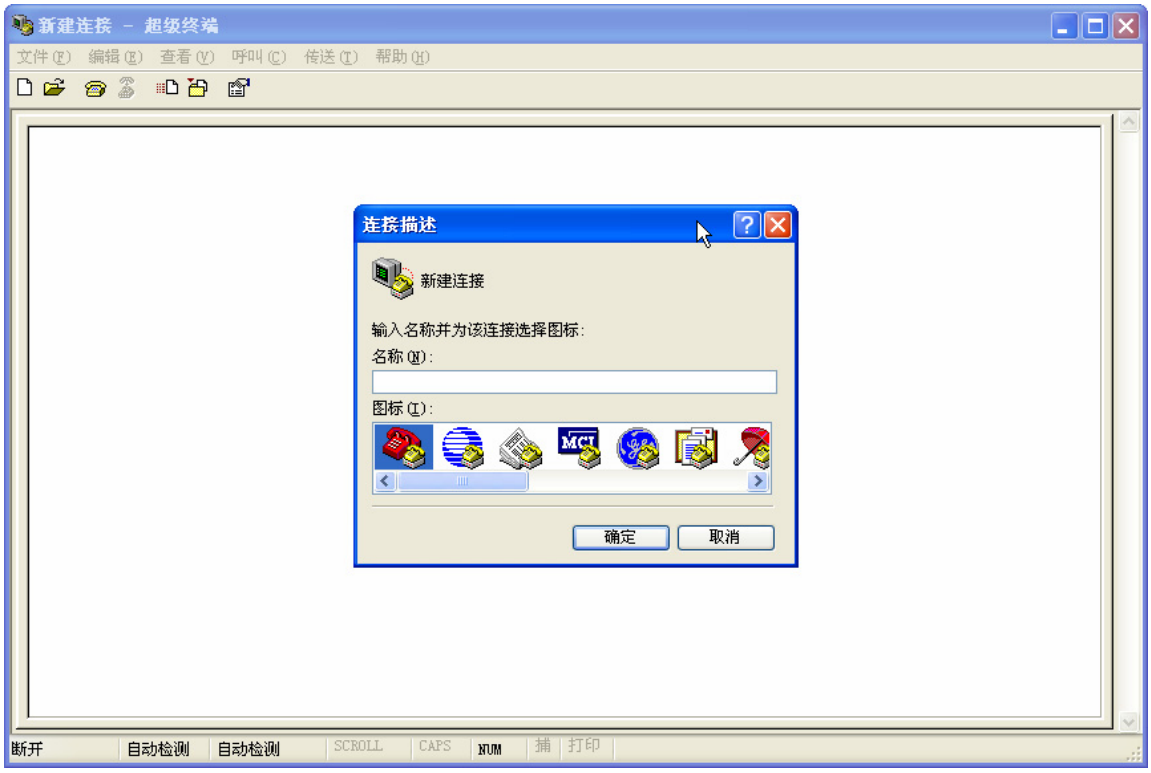

图 1-5 超级终端窗口

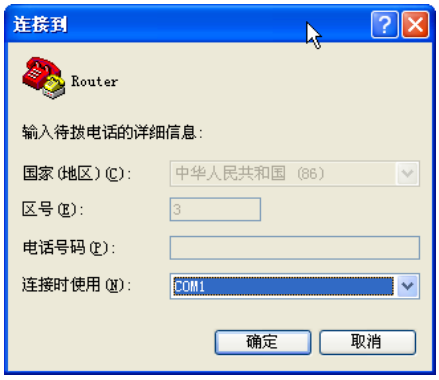

图 1-6 选择 COM 口

（3） 步骤 3：设置通信参数

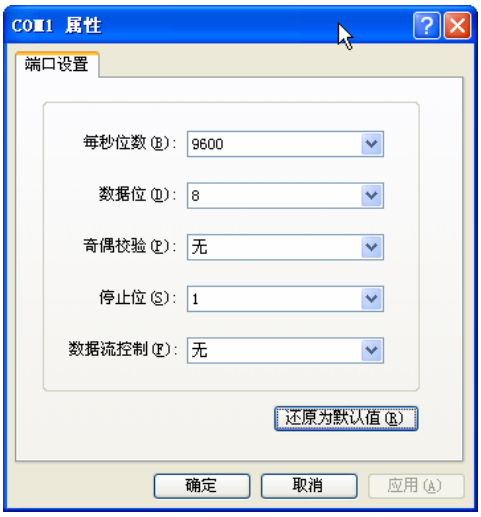

图 1-7 设置通信参数

通常路由器出厂时，波特率为 9600bps，因此在图 1-7 窗口中，点击【还原为默认值】按钮设置超级终端的通信参数；再点击【确定】按钮。按【回车】键，看看超级终端窗口上是否出现路由器提示符或其他字符，如果出现提示符或者其他字符则说明计算机已经连接到路由器了，我们可以开始配置路由器了。

（4） 步骤 4：路由器开机

关闭路由器电源，稍后重新打开电源，观察路由器的开机过程，如下：

```
System Bootstrap, Version 12.4(1r) [hqluong 1r], RELEASE SOFTWARE (fc1)
//以上显示 BOOT ROM 的版本
Copyright (c) 2005 by CISCO Systems, Inc.
Initializing memory for ECC
c2821 processor with 262144 Kbytes of main memory
Main memory is configured to 64 bit mode with ECC enabled
//以上显示路由器的内存大小
Readonly ROMMON initialized
program load complete, entry point: 0x8000f000, size: 0x274bf4c
Self                decompressing                the                image                :
################################################################################################
################################################################################################
################################################################################################
################################################################################################
################################################################################################ [OK]
//以上是 IOS 解压过程
Smart Init is enabled
smart init is sizing iomem
  ID            MEMORY_REQ                TYPE
0003E8          0X003DA000 C2821 Mainboard
                0X00264050 Onboard VPN
                0X000021B8 Onboard USB
                0X002C29F0 public buffer pools
```

```
                0X00211000 public particle pools
TOTAL:          0X00B13BF8

（此处省略）
A summary of U.S. laws governing CISCO cryptographic products may be found at:
http://www.CISCO.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@CISCO.com.

Installed image archive
CISCO 2821 (revision 49.46) with 249856K/12288K bytes of memory.  //内存大小
Processor board ID FHK1039F21Q
2 Gigabit Ethernet interfaces  //两个千兆以太网接口
2 Low-speed serial(sync/async) interfaces  //两个低速串行口（同步/异步）
1 Virtual Private Network (VPN) Module  //一个 VPN 网络模块
DRAM configuration is 64 bits wide with parity enabled.
239K bytes of non-volatile configuration memory.  //NVRAM 的大小
62720K bytes of ATA CompactFlash (Read/Write)  //FLASH 卡的大小

         --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]:
```

//以上提示是否进入配置对话模式？我们回答“n”结束该模式

4. 实验调试

图 1-8 逐一测试通信速率

如果超级终端无法连接到路由器，请按照以下顺序检查：

（1） 检查计算机和路由器之间的连接是否松动，并确保路由器已经开机；

（2） 图 1-6 中，是否选择正确的计算机 COM 口；

（3） 是否按照图 1-7 设置了正确的通信参数；

（4） 如果仍无法排除故障，而路由器非出厂设置，可能是路由器的通信波特率被修改为

非 9600bps，则如图 1-8，逐一测试通信速率；

（5） 用计算机的另一 COM 口和路由器的 console 口连接，或者确保计算机的 COM 口正常；

（6） 和供应商联系。

1.3 实验 2：通过 telnet 访问路由器

要通过 telnet 访问路由器，需要先通过 console 口对路由器进行基本配置，例如：IP 地址、密码等。

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 配置路由器以太网接口的 IP 地址，并打开接口；

（2） 配置路由器的 enable 密码和 vty 密码；

（3） telnet 程序的使用。

2. 实验拓扑

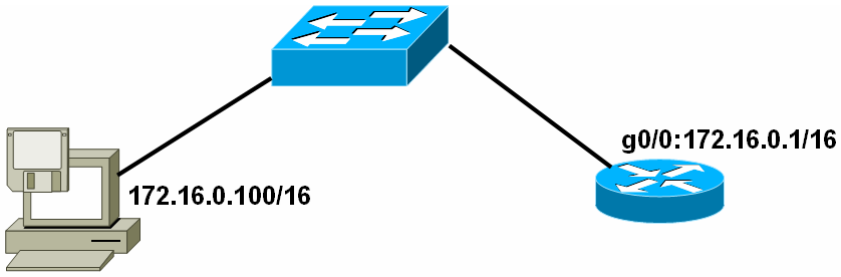

图 1-9 实验 2 拓扑

3. 实验步骤

(1) 步骤 1：配置路由器以太网接口 IP 地址

```
Router>enable
Router#
//以上是进入路由器的特权模式
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router (config)#
//以上是进入路由器的配置模式
Router (config)#interface g0/0
Router (config-if)#
//以上是进入路由器的以太网口 g0/0 接口，g0/0 中 g 表示是 GigabitEthernet，0/0 表示是第 0 个插槽中的第 0 个接口。S0/0/0 则表示为第 0 个插槽中的第 0 个模块上的第 0 个串行接口。
Router (config-if)#ip address 172.16.0.1 255.255.0.0
//以上是配置接口的 IP 地址
Router (config-if)#no shutdown
//以上是打开接口，默认时路由器的所有接口都是关闭的，这一点和交换机有很大差别
Router (config-if)#end
//退出配置模式
```

(2) 步骤 2：配置路由器密码

```
Router#conf terminal
```

Router(config)#line vty 0 4
//以上是进入路由器的VTY虚拟终端下，“vty 0 4”表示vty 0到vty 4，共5个虚拟终端
Router(config-line)#password CISCO
Router(config-line)#login
//以上是配置vty的密码，即telnet密码
Router(config-line)#exit
Router(config)#enable password CISCO
//以上是配置进入到路由器特权模式的密码
Router(config)#end
(3) 步骤3：通过telnet访问路由器
在计算机上配置网卡的IP地址为172.16.0.100/255.255.0.0,并打开DOS命令行窗口。首先测试计算机和路由器的IP连通性，再进行telnet远程登录。如下：
C:\>ping 172.16.0.1
Pinging 172.16.0.1 with 32 bytes of data:
Reply from 172.16.0.1: bytes=32 time<1ms TTL=255
Reply from 172.16.0.1: bytes=32 time<1ms TTL=255
Reply from 172.16.0.1: bytes=32 time<1ms TTL=255
Reply from 172.16.0.1: bytes=32 time<1ms TTL=255
Ping statistics for 172.16.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% lo
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
//以上表明计算机能ping通路由器

C:\>telnet 172.16.0.1
//telnet 路由器以太网卡上的IP地址
User Access Verification
Password:
Router>enable
Password:
Router#exit
//输入vty的密码CISCO、输入enable的密码CISCO，能正常进入路由器的特权模式。

4. 实验调试

如果无法从计算机上ping通路由器，依照以下步骤进行
（1） 检查计算机、交换机、路由器之间的连接是否松动；
（2） 检查连接线应该是否是直通线；
（3） 检查计算机的网卡和IP地址是否正常；
（4） 在路由器上，检查以太网接口是否正常
Router#show int g0/0
GigabitEthernet0/0 is up, line protocol is up
Hardware is MV96340 Ethernet, address is 0019.5535.b828 (bia 0019.5535.b828)
Internet address is 172.16.0.1/16
应该看到两个“up”，否则检查路由器和交换机之间的连接。

1.4　实验 3：配置终端访问服务器

使用终端访问服务器（就是插有异步模块的路由器）可以避免我们在同时配置多台路由器时频繁拔插 console 线，为了方便我们使用终端服务器，我们可以制作一个简单的菜单。

1. 实验目的

通过本实验，读者可以掌握如下技能：

(1)　配置终端访问服务器，并制作一个简单的菜单；

(2)　使用终端访问服务器控制路由器；

2. 实验拓扑

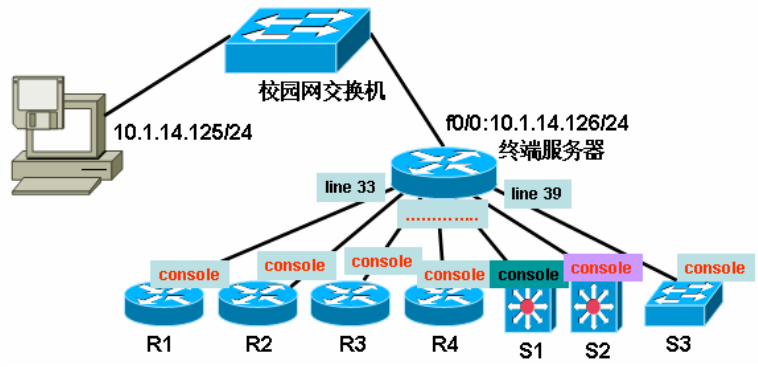

图 1-10 终端服务器和各路由器、交换机连接图

3. 实验步骤

(1)　步骤 1：终端服务器的基本配置

```
Router(config)#hostname Terminal-Server
//以上是配置终端服务器的主机名
Terminal-Server(config)#enable secret ccielab
//以上是配置进入特权模式的密码，防止他人修改终端服务器的配置
Terminal-Server(config)#no ip domain-lookup
//以上禁止路由器查找 DNS 服务器，防止我们输入错误命令时的长时间等待
Terminal-Server(config)#line vty 0 ?
  <1-15>  Last Line number
  <cr>
//查看该路由器支持多少 vty 虚拟终端，可以看到支持 0-15
Terminal-Server(config)#line vty 0 15
Terminal-Server(config-line)#no login
Terminal-Server(config-line)#logging synchronous
Terminal-Server(config-line)#no exec-timeout
Terminal-Server(config-line)#exit
//以上允许任何人不需密码就可以 telnet 该终端服务器，并且即使长时间不输入命令也不超时自动 logout 出来

Terminal-Server#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Terminal-Server(config)#interface f0/0
Terminal-Server(config-if)#ip address 10.1.14.126 255.255.255.0
Terminal-Server(config-if)#no shutdown
```

Terminal-Server(config-if)#exit
//以上配置以太网接口的 ip 地址为 10.1.14.126/255.255.255.0，并打开接口
Terminal-Server(config)#no ip routing
//由于终端服务器不需要路由功能，所以关闭路由功能，这时终端服务器相当于一台计算机
Terminal-Server(config)#ip default-gateway 10.1.14.254
//配置网关，允许他人从别的网段 telnet 该终端服务器
(2) 步骤 2：配置线路、制作简易菜单
Terminal-Server#show line

	Tty	Typ	Tx/Rx	A	Modem	Roty	AccO	AccI	Uses	Noise	Overruns	Int
*	0	CTY		-	-	-	-	-	0	0	0/0	-
*	33	TTY	9600/9600	-	-	-	-	-	6	3	238/717	-
*	34	TTY	9600/9600	-	-	-	-	-	1	0	274/823	-
*	35	TTY	9600/9600	-	-	-	-	-	1	0	244/736	-
*	36	TTY	9600/9600	-	-	-	-	-	5	57	255/767	-
*	37	TTY	9600/9600	-	-	-	-	-	1	0	1128/3388	-
*	38	TTY	9600/9600	-	-	-	-	-	0	7	1289/3864	-
	39	TTY	9600/9600	-	-	-	-	-	1	15	1175/3524	-
	40	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	41	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	42	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	43	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	44	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	45	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	46	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	47	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	48	TTY	9600/9600	-	-	-	-	-	0	0	0/0	-
	65	AUX	9600/9600	-	-	-	-	-	0	0	0/0	-
*	66	VTY		-	-	-	-	-	6	0	0/0	-
*	67	VTY		-	-	-	-	-	2	0	0/0	-
*	68	VTY		-	-	-	-	-	2	0	0/0	-
*	69	VTY		-	-	-	-	-	5	0	0/0	-
*	70	VTY		-	-	-	-	-	12	0	0/0	-
	71	VTY		-	-	-	-	-	2	0	0/0	-
	72	VTY		-	-	-	-	-	0	0	0/0	-
	73	VTY		-	-	-	-	-	0	0	0/0	-
	74	VTY		-	-	-	-	-	0	0	0/0	-
	75	VTY		-	-	-	-	-	0	0	0/0	-
	76	VTY		-	-	-	-	-	0	0	0/0	-
	77	VTY		-	-	-	-	-	0	0	0/0	-
	78	VTY		-	-	-	-	-	0	0	0/0	-
	79	VTY		-	-	-	-	-	0	0	0/0	-
	80	VTY		-	-	-	-	-	0	0	0/0	-
	81	VTY		-	-	-	-	-	0	0	0/0	-

//以上是查看终端服务器上异步模块的各异步口所在的线路编号，tty 表示的就是异步模

块，该终端服务器模块有 16 个接口，线路编号为 33-48，我们这里实际上只用了 33-39。记住线路的编号，后面需要根据这些编号进行配置。

```
Terminal-Server#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Terminal-Server(config)#line 33 48
Terminal-Server(config-line)#transport input all
//进入线路模式下，线路允许所有传入，实际上我们只允许 telnet 进入即可
Terminal-Server(config-line)#exit
Terminal-Server(config)#int loopback0
Terminal-Server(config-if)#ip address 1.1.1.1 255.255.255.255
//以上配置 loopback0 接口的 ip 地址，loopback 接口是一个逻辑上的接口，路由器上可以任意创建几乎无穷多的 loopback 接口，该接口可以永远是 UP 的。loopback 接口经常用于测试等。
Terminal-Server(config-if)#exit
Terminal-Server(config)#ip host R1 2033 1.1.1.1
Terminal-Server(config)#ip host R2 2034 1.1.1.1
Terminal-Server(config)#ip host R3 2035 1.1.1.1
Terminal-Server(config)#ip host R4 2036 1.1.1.1
Terminal-Server(config)#ip host S1 2037 1.1.1.1
Terminal-Server(config)#ip host S2 2038 1.1.1.1
Terminal-Server(config)#ip host S3 2039 1.1.1.1
Terminal-Server(config)#exit
//从终端服务器控制各路由器，是通过反向 telnet 实现的，此时 telnet 的端口号为线路编号加上 2000，例如 line 33，其端口号为 2033，如果要控制 line 33 线路上连接的路由器，我们可以采用：“telnet 1.1.1.1 2033”命令。然而这样命令很长，为了方便，所以我们使用“ip host”命令定义一系列的主机名，这样可以之间输入“R1”控制 line 33 线路上连接的路由器了。

Terminal-Server(config)#alias exec cr1 clear line 33
Terminal-Server(config)#alias exec cr2 clear line 34
Terminal-Server(config)#alias exec cr3 clear line 35
Terminal-Server(config)#alias exec cr4 clear line 36
Terminal-Server(config)#alias exec cs1 clear line 37
Terminal-Server(config)#alias exec cs2 clear line 38
Terminal-Server(config)#alias exec cs3 clear line 39
Terminal-Server(config)#
//以上是定义了一系列的命令别名，例如“cr1”=“clear line 33”，“clear line”命令的作用是清除线路

Terminal-Server(config)#privilege exec level 0 clear line
Terminal-Server(config)#privilege exec level 0 clear
//以上是使得我们在用户模式下也能使用“clear line”和“clear”命令
```

```
Terminal-Server(config)#banner motd #
Enter TEXT message.  End with the character '#'.
        ****************************************
        R1-------R1     cr1------clear line 33
        R2-------R2     cr2------clear line 34
        R3-------R3     cr3------clear line 35
        R4-------R4     cr4------clear line 36
        S1-------s1     cs1------clear line 37
        S2-------s2     cs2------clear line 38
        S3-------s3     cs3------clear line 39
        ****************************************
#
```

//以上是制作一个简单的菜单，提醒用户：要控制 R1 路由器可以使用“R1”命令（大小写不敏感）；要清除 R1 路由器所在的线路，可以使用“cr1”命令。我们是利用路由器的 banner motd 功能实现的，该功能使得我们 telnet 到路由器后，就显示以上简易菜单。

(3) 步骤 3：测试能否从终端服务器控制个路由器和交换机

在计算机上配置网卡的 IP 地址为 10.1.14.125/255.255.255.0，并打开 DOS 命令行窗口。首先测试计算机和路由器的 IP 连通性，再进行 telnet 远程登录。如下：

```
C:\Documents and Settings\longkey>ping 10.1.14.126

Pinging 10.1.14.126 with 32 bytes of data:
Reply from 10.1.14.126: bytes=32 time<1ms TTL=255
Reply from 10.1.14.126: bytes=32 time<1ms TTL=255
Reply from 10.1.14.126: bytes=32 time=1ms TTL=255
Reply from 10.1.14.126: bytes=32 time=18ms TTL=25

Ping statistics for 10.1.14.126:
    Packets: Sent = 4, Received = 4, Lost = 0 (0%
Approximate round trip times in milli-seconds:
     Minimum = 0ms, Maximum = 18ms, Average = 4ms
```

//以上表明计算机能 ping 通终端服务器

```
C:\Documents and Settings\longkey>telnet 10.1.14.126
        ****************************************
        R1-------R1     cr1------clear line 33
        R2-------R2     cr2------clear line 34
        R3-------R3     cr3------clear line 35
        R4-------R4     cr4------clear line 36
        S1-------s1     cs1------clear line 37
        S2-------s2     cs2------clear line 38
        S3-------s3     cs3------clear line 39
        ****************************************
```

//telnet 到 10.1.14.126，出现简易菜单

```
Terminal-Server>cr1
[confirm]
 [OK]
Terminal-Server>
//先用“cr1”命令清除线路 33，该线路上连接了路由器 R1
Terminal-Server>r1
Trying R1 (1.1.1.1, 2033)... Open

          ****************************************
          R1-------R1      cr1------clear line 33
          R2-------R2      cr2------clear line 34
          R3-------R3      cr3------clear line 35
          R4-------R4      cr4------clear line 36
          S1-------s1      cs1------clear line 37
          S2-------s2      cs2------clear line 38
          S3-------s3      cs3------clear line 39
          ****************************************

R1>
//输入“r1”命令，如果出现“R1>”或者“Router>”等，表明可以控制 R1 路由器了。如果出现以下情况：
Terminal-Server>r1
Trying R1 (1.1.1.1, 2033)...
% Connection refused by remote host
请执行几次“cr1”命令后，重新执行“r1”命令。
```

(4) 步骤 4：测试能否从终端服务器控制各路由器和交换机

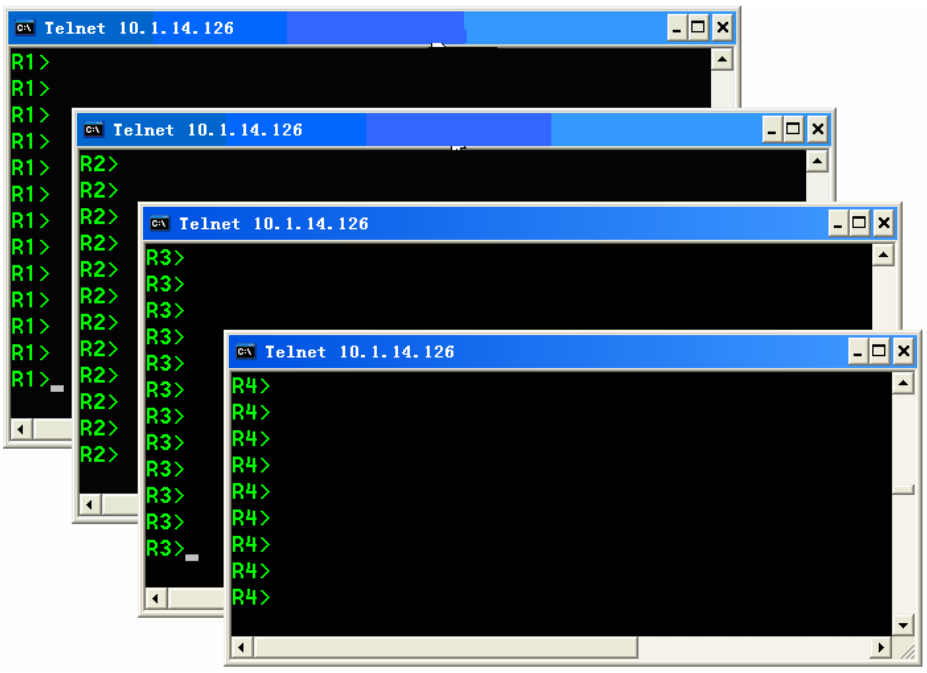

图 1-11 打开多个路由器或者交换机的控制窗口

重复步骤（3），可以打开不同路由器或者交换机的控制窗口，这样我们就可以在一台计算机上同时配置不同的路由器和交换机了，如图 1-11。当然，一台路由器只能被一台计算机所控制。

【提示】实际应用中如果需要配置多台设备，不建议使用 Windows 自带的 telnet 程序，可以选用 SecureCRT 等专业终端软件，这些软件的功能完善，更方便我们的配置。

1.5 本章小结

本章介绍了如何把计算机上的串口和路由器的 console 进行连接，来配置路由器。也介绍了如何配置路由器以使管理员能够通过 telnet 远程配置路由器。还介绍如何配置终端访问服务器，方便我们同时配置多台路由器或者交换机。本章给出了一直贯穿本书的网络拓扑。表 1-1 是本章出现的命令。

表 1-1 本章命令汇总

命令	作用
enable	从用户模式进入特权模式
configure terminal	进入配置模式
interface g0/0	进入千兆以太网接口模式
ip address 172.16.0.1 255.255.0.0	配置接口的 ip 地址
no shutdown	打开接口
line vty 0 4	进入虚拟终端 vty 0 - vty 4
password CISCO	配置密码
login	用户要进入路由器，需要先进行登录
exit	退回到上一级模式
enable password CISCO	配置进入特权模式的密码，密码不加密
end	直接回到特权模式
show int g0/0	显示 g0/0 接口的信息
hostname Terminal-Server	配置路由器的主机名
enable secret ccielab	配置进入特权模式的密码，密码加密
no ip domain-lookup	路由器不使用 DNS 服务器解析主机的 IP 地址
logging synchronous	路由器上的提示信息进行同步，防止信息干扰我们输入命令
no ip routing	关闭路由器的路由功能
ip default-gateway 10.1.14.254	配置路由器访问其他网段时所需的网关
show line	显示各线路的状态
line 33 48	进入 33-48 线路模式
transport input all	允许所有协议进入线路
int loopback0	进入 loopback0 接口
ip host R1 2033 1.1.1.1	为 1.1.1.1 主机起一个主机名
alias exec cr1 clear line 33	为命令起一个别名
privilege exec level 0 clear line	把命令 clear line 的等级改为 0,在用户模式下也可以执行它
banner motd	设置用户登录路由器时的提示信息

第 2 章 路由器基本配置

本章将先简要介绍路由器的硬件组成，而重点介绍路由器中最重要的部分 IOS，对路由器的配置实际上就是对 IOS 软件进行配置。IOS 提供了大量的命令，熟悉这些命令才能很好地发挥路由器的功能，本章介绍的是路由器的一些基础性命令。

2.1 路由器简介和 IOS 简介

2.1.1 路由器简介

路由器能起到隔离广播域的作用，还能在不同网络间转发数据包。路由器实际上是一台特殊用途的计算机，和常见的 PC 机一样，路由器有 CPU、内存、BOOT ROM。路由器没有键盘、硬盘、显示器；然而比起计算机，路由器多了 NVRAM、FLASH 以及各种各样的接口。路由器各个部件的作用如下：

(1) CPU：中央处理单元，和计算机一样，它是路由器的控制和运算部件。

(2) RAM/DRAM：内存，用于存储临时的运算结果，如：路由表、ARP 表、快速交换缓存、缓冲数据包、数据队列、当前配置。众所周知，RAM 中的数据在路由器断电后是会丢失的。

(3) FLASH：可擦除、可编程的 ROM，用于存放路由器的 IOS，FLASH 的可擦除特性允许我们更新、升级 IOS 而不用更换路由器内部的芯片。路由器断电后，FLASH 的内容不会丢失。FLASH 容量较大时，就可以存放多个 IOS 版本。

(4) NVRAM：非易失性 RAM，用于存放路由器的配置文件，路由器断电后，NVRAM 中的内容仍然保持。

(5) ROM：只读存储器，存储了路由器的开机诊断程序、引导程序和特殊版本的 IOS 软件（用于诊断等有限用途），ROM 中软件升级时需要更换芯片。

(6) 接口（Interface）：用于网络连接，路由器就是通过这些接口和不同的网络进行连接的。

2.1.2 IOS 简介

路由器也有自己的操作系统，通常称为 IOS（Internetwork Operating System）。和计算机上的 Windows 一样，IOS 是路由器的灵魂，所有配置是通过 IOS 完成的。Cisco 的 IOS 是命令行界面（称为 CLI，Command Line Interface），CLI 有两种基本工作模式：

(1) 用户模式（User mode）：通常用来查看路由器的状态。在此状态下，无法对路由器进行配置，可以查看的路由器信息也是有限的。

(2) 特权模式（Privilege mode）：可以更改路由器的配置，当然也可以查看路由器的所有信息。

表 2-1 常用的编辑命令

编辑键	命令功能
【Crtl+A】	移动光标到命令行开头
【Crtl+E】	移动光标到命令行末尾
【Crtl+P】(或【↑】)	重用前一条命令
【Crtl+N】(或【↓】)	重用下一条命令
【Esc+F】	光标前移一个词

【Esc+B】	光标后移一个词
【Crtl+F】	光标前移一个字母
【Crtl+B】	光标后移一个字母

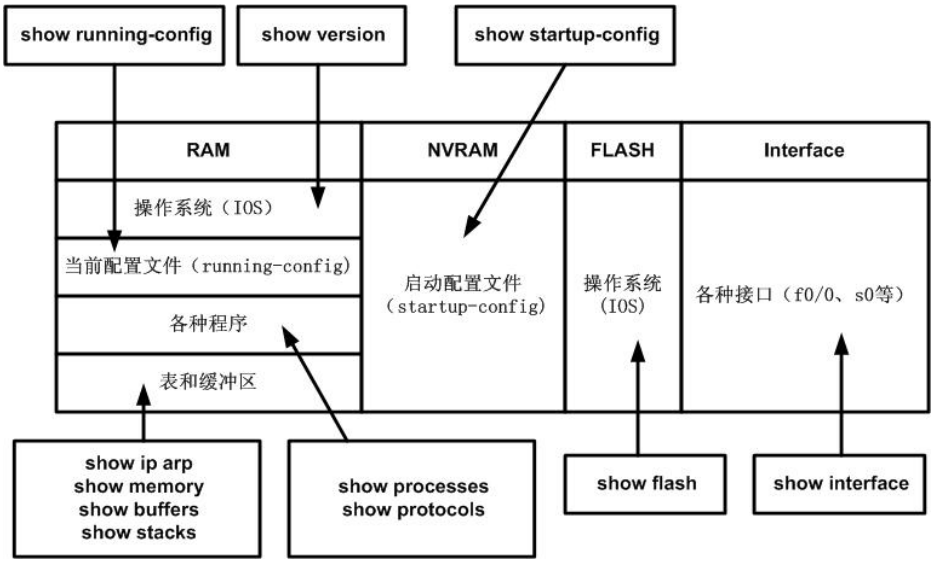

图 2-1 “show”命令显示路由器的各种信息

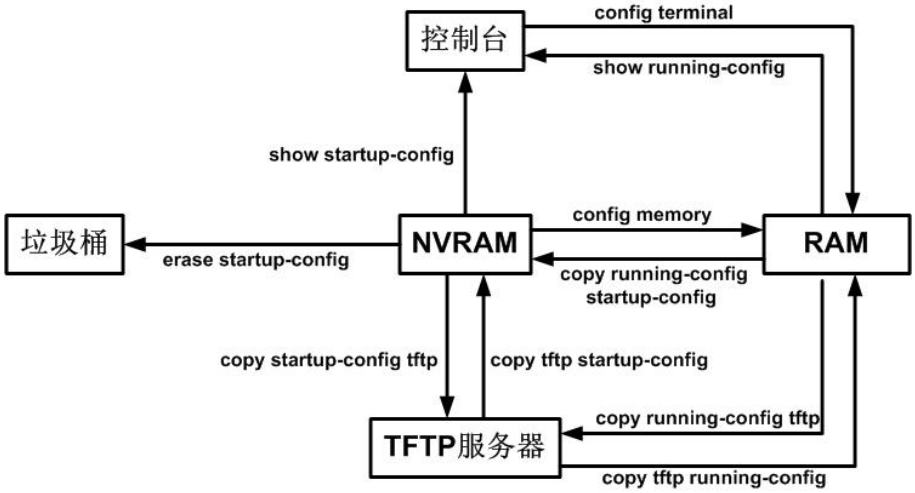

图 2-2 配置文件的流动

虽然是命令行，CLI 提供简单、丰富的编辑功能，如表 2-1，熟悉它们是熟练配置路由器的基础。在 CLI 下可以使用“show”命令查看存放在路由中不同部件中的信息，如图 2-1。CLI 下，我们可以在路由器的各种模式间进行切换来对路由器进行配置。

我们对路由器进行配置后，可以把配置保存在 NVRAM 中，路由器开机时会自动读取。为了安全，可以通过 TFTP 服务器把配置文件备份在计算机上。路由器的配置文件可以在不同的部件间流动，流动如图 2-2 所示。

路由器的 IOS 是如此重要，因此我们也需要通过 TFTP 服务器把 IOS 备份到计算机上。由于各种各样的原因，我们可能会不小心破坏了 IOS，造成路由器无法开机，可以通过 TFTP 把之前备份出的 IOS 进行恢复。如果不慎忘记了路由器的密码，也可以进行恢复。

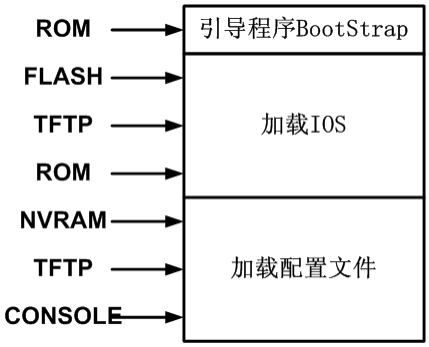

图 2-3　路由器启动过程示意图

Cisco 路由器开机后，首先执行一个开机自检过程（Power On Self Test，POST），诊断验证 CPU、内存及各个端口是否正常，紧接路由器进入软件初始化过程。如图 2-3，其步骤如下：

(1) 执行 ROM 中的引导程序加载（Bootstrap Loader），它和计算机中的 BIOS 很类似，Bootstrap 会把 IOS 装到 RAM 中；

(2) IOS 可以存放在许多地方（FLASH、TFTP 服务器上或 ROM 中），路由器寻找 IOS 映像的顺序，取决于配置寄存器的启动域以及其他的设置。配置寄存器（configuration register）是一个 16 位（2 进制）的寄存器，低 4 位就是启动域，不同的值代表从不同的位置查找 IOS，如表 2-2。详细的 IOS 查找过程如图 2-4 所示。

表 2-2　配置寄存器中启动于的值

配置寄存器的值（16 进制）	描述
0	使用 ROM 模式
1	自动从 ROM 中启动
2-F	从 FLASH 或 TFTP 服务器启动

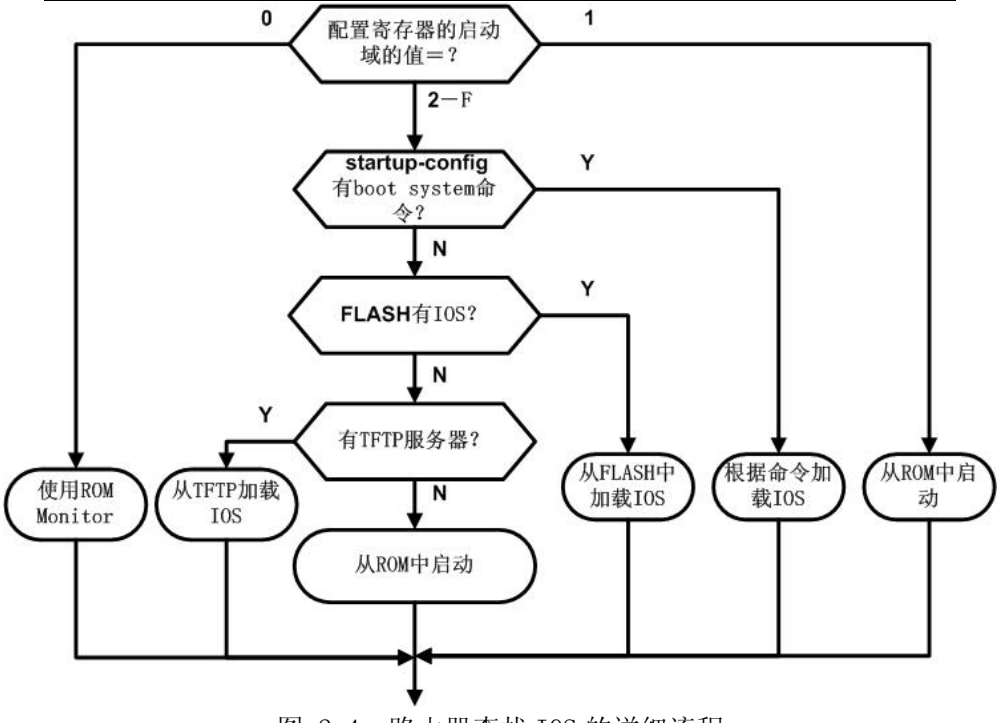

图 2-4　路由器查找 IOS 的详细流程

(3) 加载 IOS 到 RAM 中：如果 IOS 是压缩过的，就先解压。
(4) 在 NVRAM 中查找配置文件，并把配置文件加载到 RAM 中运行。
(5) 如果在 NVRAM 中没有找到配置文件，就进入 setup 配置模式（也称为配置对话模式）。

2.1.3 CDP 协议介绍

CDP（Cisco Discovery Protocol）协议是 Cisco 专有的协议，是使 Cisco 网络设备能够发现相邻的、直连的其他 Cisco 设备的协议。CDP 是数据链路层的协议，因此使用不同的网络层协议的 Cisco 设备也可以获得对方的信息。CDP 协议默认是启动的。

2.2 实验 1:CLI 的使用与 IOS 基本命令

1. 实验目的

通过本实验，读者可以掌握如下技能：
（1） 熟悉路由器 CLI 的各种模式
（2） 熟悉路由器 CLI 各种编辑命令
（3） 掌握路由器的 IOS 基本命令
（4） 查看路由器的有关信息

2. 实验拓扑

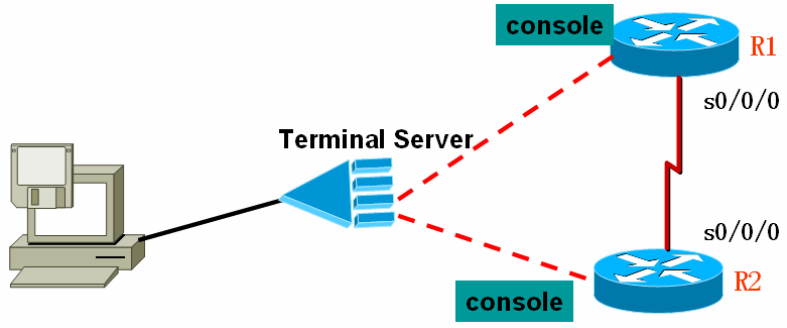

图 2-5 实验 1 拓扑图

3. 实验步骤

（1） 步骤 1：用户模式和特权模式的切换

```
Router>
Router>enable
Router#
Router#disable
Router>
```

//“Router”是路由器的名字，而“>”代表是在用户模式。“enable”命令可以使路由器从用户模式进入到特权模式，“disable”命令则相反，在特权模式下的提示符为“#”。

（2） 步骤 2：“?”和【Tab】键的使用，以配置路由器时钟为例

```
Router>enable
Router#clok
Translating "clok"...domain server (255.255.255.255)
 (255.255.255.255)
```

Translating "clok"...domain server (255.255.255.255)
% Unknown command or computer name, or unable to find computer address
//以上表明输入了错误的命令
Router#cl?
clear clock
//路由器列出了当前模式下可以使用的以“cl”开头的所有命令
Router#clock
% Incomplete command.
//路由器提示命令输入不完整，
Router#clock ?
set Set the time and date
//要注意的是“?”和“clock”之间要有空格，否则得到将不同的结果，如果不加空格路由器以为你是想列出以“clock”字母开头的命令，而不是想列出“clock”命令的子命令或参数。

Router#clock set ?
hh:mm:ss Current Time
Router#clock set 11:36:00
% Incomplete command.
Router#clock set 11:36:00 ?
<1-31> Day of the month
MONTH Month of the year
Router#clock set 11:36:00 12 ?
MONTH Month of the year
//以上多次使用“？”帮助命令，获得了“clock”命令的格式
Router#clock set 11:36:00 12 08
                             ^
% Invalid input detected at '^' marker.
//路由器提示输入了无效的参数，并用“^”指示错误的所在
Router#clock set 11:36:00 12 august
% Incomplete command.
Router#clock set 11:36:00 12 august 2003
Router#show clock
11:36:03.149 UTC Tue Aug 12 2003
//到此成功配置了路由器的时钟，通常如果命令成功，路由器不会有任何提示。在CLI下，可以直接使用“?”命令获得当前模式下的全部命令。如下：
Router# ?
Exec commands:
  access-enable   Create a temporary Access-List entry
  access-profile   Apply user-profile to interface
  access-template Create a temporary Access-List entry
  ....//为了节约篇幅，此处省略了部分输出
  erase   Erase a filesystem
  exit   Exit from the EXEC

```
  help            Description of the interactive help system
 --More--
```
//有多于一屏的内容时，按【回车】键显示下一行，按【空格】显示下一页，其他键则退出

```
Router#disable
Router>en
Router#
```
//在 CLI 中，命令是可以缩写的，但前提是路由器要能够区分得出，如下：

```
Router#dis
% Ambiguous command:  "dis"
Router#dis?
disable  disconnect
```
//使用“dis”不能退出特权模式的原因是路由器无法区分出“dis”代表“disable”还是“disconnect”。若再加多一个字母“a”就可以区分了。

```
Router#disa
Router>en【Tab】
Router>enable
Router#conf【Tab】 t【Tab】
Router#configure terminal
Router(config)#
```
//可以使用【Tab】键帮助我们自动完成命令

（3） 步骤 3：IOS 编辑命令与历史命令缓存大小

```
Router#show history
  en
  conf t
  show history
  dis
  disable
  enable
  conf t
  show history
```
//以上是显示历史命令

```
Router#terminal editing
```
//以上是打开编辑功能，实际上这是默认的。用上下左右光标键试试移动光标，也可以试试使用表 2-1 的编辑键移动光标

```
Router#terminal history size 50
```
//以上把缓存的历史命令数改为 50，默认值为 10

```
Router#terminal no editing
```
//以上关闭 terminal 的编辑功能，则表 2-1 的编辑键失效

```
Router#terminal editing
```
（4） 步骤 4：基本 IOS 命令

先连接到 R1 路由器上：

```
Router>enable
Router#configure terminal
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R1
//以上改变路由器的名称为“R1”，设置立即生效。
R1(config)#enable password cisco
//以上改变了 enable 的密码为“cisco”，这个密码是从用户模式进入到特权模式的密码。
R1(config)#interface g0/0
//以上进入到接口模式，这里是千兆以太网口（第 0 个插槽的第 0 个接口，编号从 0 开始）。
R1(config-if)#ip address 10.1.1.1 255.255.255.0
//以上给以太接口配置一个 IP 地址 10.1.1.1，掩码为 255.255.255.0 。
R1(config-if)#no shutdown
//以上开启以太网口，因为默认时路由器的各个接口是关闭的。
R1(config-if)#exit
//退回到上一级模式
R1(config)#interface s0/0/0
//以上进入到接口模式，这里是串行接口
R1(config-if)#ip address 10.12.12.1 255.255.255.0
//以上给串行接口配置一个 IP 地址
R1(config-if)#no shutdown
//以上开启接口
R1(config-if)#end（或【Ctrl+Z】）
//以上结束配置直接回到特权模式下。
R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
//以上把内存中的配置保存到 NVRAM 中，路由器开机时会读取它。
```

连接到 R2 路由器上，进行以下操作：

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R2
R2(config)#enable password cisco
R2(config)#interface g0/0
R2(config-if)#ip address 10.2.2.2 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#interface s0/0/0
R2(config-if)#ip address 10.12.12.2 255.255.255.0
R2(config-if)#clock rate 128000
//R2 这一端是 DCE，需要配置时钟
R2(config-if)#no shutdown
R2(config-if)#end
R2#copy running-config startup-config
```

Destination filename [startup-config]?
Building configuration...
[OK]

R2#ping 10.12.12.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.14.126, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
//从 R2 ping R1 的串行接口，可以 ping 通
（5） 步骤 5：各种“show”命令
R2#show version
Cisco IOS Software, 2800 Software (C2800NM-ADVENTERPRISEK9-M), Version 12.4(11)T1, RELEASE SOFTWARE (fc5)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 25-Jan-07 12:50 by prod_rel_team
//以上是 IOS 的版本信息
ROM: System Bootstrap, Version 12.4(1r) [hqluong 1r], RELEASE SOFTWARE (fc1)
//以上是 ROM 的版本信息
R2 uptime is 4 hours, 10 minutes //注：路由器的开机时间
System returned to ROM by power-on //路由器是如何启动的，例如：开电或者热启动
System image file is "flash:c2800nm-adventerprisek9-mz.124-11.T1.bin"
//注：以上是当前正在使用的 IOS 文件名

（此处省略）
If you require further assistance please contact us by sending email to export@cisco.com.

Cisco 2821 (revision 53.50) with 249856K/12288K bytes of memory.
//注：以上是路由型号、RAM 大小(249856K+12288K)
Processor board ID FHK1039F1FG //主板系列号
2 Gigabit Ethernet interfaces
4 Low-speed serial(sync/async) interfaces
1 Virtual Private Network (VPN) Module
//注：以上是各种接口的数量
DRAM configuration is 64 bits wide with parity enabled.
239K bytes of non-volatile configuration memory.
62720K bytes of ATA CompactFlash (Read/Write)
//注：以上是 NVRAM、FLASH 的大小情况

Configuration register is 0x2142
//以上是配置寄存器的值

```
R2#show running-config
Building configuration...

Current configuration : 1238 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname R2
.....（此处省略）
//以上显示路由正在使用的配置文件（存放在 RAM 中），通常配置文件为几百到几千字节

R2#show startup-config
Building configuration...

Current configuration : 1238 bytes
!
version 12.4
//以上显示路由 NVRAM 中的配置文件

R2#show interface s0/0/0
Serial0/0/0 is up, line protocol is up   //注：该接口的状态
  Hardware is GT96K Serial
  Internet address is 10.12.12.2/24  //注：该接口的 IP 地址
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  //注：以上是该接口的 MTU、带宽、延时、可靠性、负载大小
  Encapsulation HDLC, loopback not set  //注：串口的封装类型为 HDLC
  Keepalive set (10 sec)
.....（此处省略）

R2#show flash
CompactFlash directory:
File  Length   Name/status
  1   41205996  c2800nm-adventerprisek9-mz.124-11.T1.bin
[41206060 bytes used, 23019216 available, 64225276 total]
62720K bytes of ATA CompactFlash (Read/Write)
//显示了 flash 中存放的 IOS 情况，flash 的总大小，可用空间

R2#show controllers s0/0/0
Interface Serial0/0/0
Hardware is GT96K
```

```
DCE V.35, clock rate 128000
idb at 0x4728A8C0, driver data structure at 0x4728CBEC
wic_info 0x4728D218
Physical Port 1, SCC Num 1
//显示 s0/0/0 接口为 v.35 接口，且为 DCE，已经配置了时钟

R2#show ip arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  172.16.0.2             -   0019.5566.6320  ARPA   GigabitEthernet0/0
Internet  172.16.0.100            3   000c.7650.df17  ARPA   GigabitEthernet0/0
//以上显示路由中缓存的 ARP 表
```

2.3 实验 2:配置文件的备份和 IOS 的备份

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 熟悉 TFTP 服务器的使用

（2） 熟悉备份路由器的配置文件

（3） 掌握备份路由器的 IOS

2. 实验拓扑

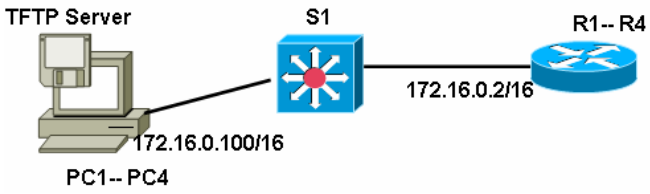

图 2-6 实验 2 拓扑图

3. 实验步骤

（1） 步骤 1：TFTP SERVER 软件的安装、准备

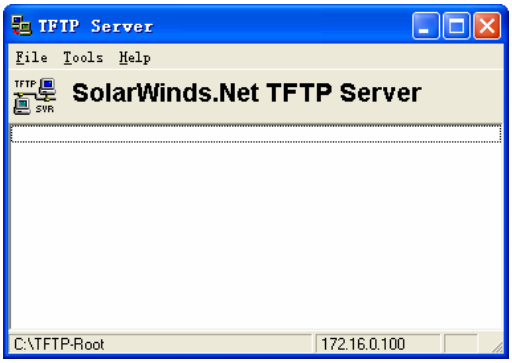

图 2-7 TFTP Server 主窗口

Tftp 服务器软件有各种各样，本书以 SolarWinds TFTP Server 软件为例，该软件可以从 http://solarwinds.net 上免费下载。下载后安装后，运行该软件，如图 2-7。从【File】

→【Configure】菜单打开配置窗口，如图 2-8。在“TFTP Root Directory”选项卡中，可以看到 TFTP 的主目录为 c:\tftp-root，TFTP Server 接收到的文件将存放在该目录，也从该目录查找要发送的文件。如图 2-9，在“Security”选项卡中，配置该 TFTP 可以接收和发送文件。

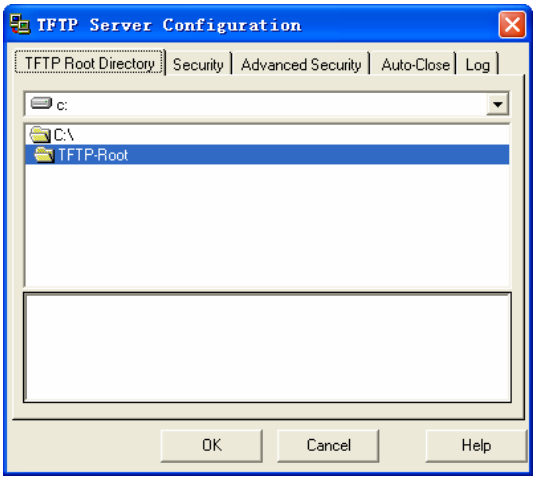

图 2-8 查看 TFTP 服务器的主目录

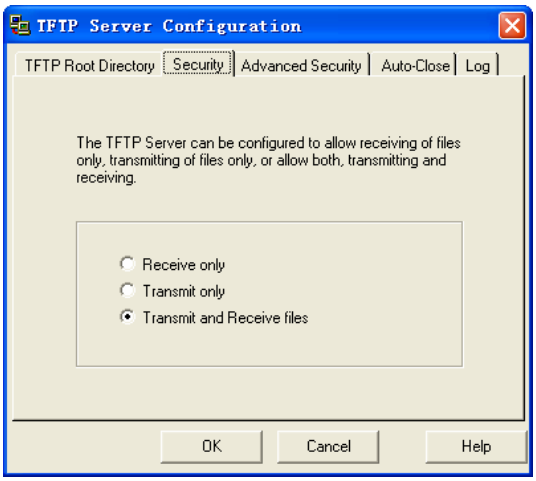

图 2-9 配置 TFTP Server 可以接收和发送文件

（2） 步骤 2：路由器和计算机间的 IP 可达

首先确保 S1 交换机为出厂配置，如果不是的话，请执行以下命令：

```
Switch>enable
Switch#delete flash:vlan.dat
Switch#erase startup-config
```

其次在 PC 机上配置 IP 地址为 172.16.0.100/16。

```
R2(config)#int g0/0
R2(config-if)#no shutdown
R2(config-if)#ip address 172.16.0.2 255.255.0.0
R2(config-if)#exit
//以上在路由器的以太网接口配置 IP 地址，并启用接口
R2#ping 172.16.0.100
```

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.100, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
//测试从 R2 到 PC（TFTP 服务器）的 IP 可达
（3） 步骤 3：备份配置文件到 TFTP 服务器
R2#**copy running-config tftp:**
//把内存中的配置文件备份到 TFTP 服务器上
Address or name of remote host []? 172.16.0.100 //回答 TFTP 服务器的 IP 地址
Destination filename [r2-confg]? //回答文件名，默认时为“路由器名-confg”
!!
1381 bytes copied in 0.452 secs (3055 bytes/sec)
//备份成功，共 1381 字节，可以在 c:\tftp-root 目录下找到该文件，是一个纯文本的文件。可以用写字板打开，而用记事本打开则格式会出现问题
（4） 步骤 4：采用“复制、粘贴”备份配置文件

使用 TFTP 服务器备份配置文件很是麻烦，我们也可以简单地在终端窗口中，执行“**show running-config**”命令显示当前的配置，在终端窗口中复制全部配置，粘贴到某文本文件中。

【提示】如果是在 Windows 自带的超级终端窗口中复制、粘贴配置，会有“---more---”等字样，要记得删除这些字符。

（5） 步骤 5：备份配置 IOS 到 TFTP 服务器
R2#**show flash:**
CompactFlash directory:
File Length Name/status
  1 41205996 c2800nm-adventerprisek9-mz.124-11.T1.bin
[41206060 bytes used, 23019216 available, 64225276 total]
62720K bytes of ATA CompactFlash (Read/Write)
//先查看 flash 中的 IOS 大小，文件名等
R2#**copy flash:c2800nm-adventerprisek9-mz.124-11.T1.bin tftp:**
//把 IOS 备份到 TFTP 服务器上
Address or name of remote host []? 172.16.0.100 //回答 TFTP 服务器的 IP 地址
Destination filename [c2800nm-adventerprisek9-mz.124-11.T1.bin]?
//回答文件名。默认时和源文件名是一样的，不建议修改文件名，因为 IOS 文件名包含了 IOS 的版本、特征等信息
!!（此处省略）
//备份成功可以在 c:\tftp-root 目录下找到该文件。

2.4 实验 3:密码恢复和 IOS 的恢复

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 熟悉路由器的密码恢复步骤

（2） 熟悉路由器的 IOS 恢复步骤

2. 实验拓扑

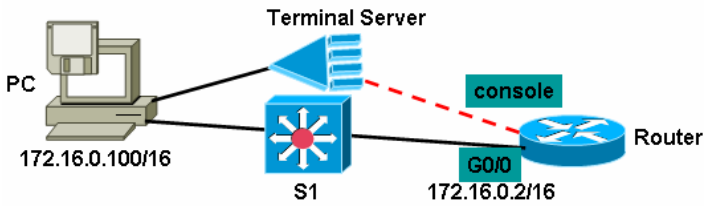

图 2-10 实验 3 拓扑图

3. 实验步骤

（1） 步骤 1：在路由器上配置密码

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname R2
R2(config)#enable password 309nasfdndf12
```

//故意配置一个自己也记不住的密码，以供密码恢复使用

（2） 步骤 2：路由器密码恢复

关闭路由器电源并重新开机，当控制台出现启动过程，赶快按【Ctrl+Break】键中断路由器的启动过程，进入 rommon 模式，如下：

```
System Bootstrap, Version 12.4(1r) [hqluong 1r], RELEASE SOFTWARE (fc1)
Copyright (c) 2005 by cisco Systems, Inc.
Initializing memory for ECC
c2821 processor with 262144 Kbytes of main memory
Main memory is configured to 64 bit mode with ECC enabled
Readonly ROMMON initialized
rommon 1>confreg 0x2142
```

//改变配置寄存器的值为 0x2142，这会使得路由器开机时不读取 NVRAM 中的配置文件 。

```
rommon 2 > i
```

//重启路由器。路由器重启后会直接进入到 setup 配置模式，用【Ctrl+C】或者回答“n”，退出 setup 模式。

```
Router>enable
Router#copy startup-config running-config
Destination filename [running-config]?
661 bytes copied in 0.625 secs
```

//把配置文件从 NVRAM 中拷贝到 RAM 中，在此基础上修改密码。

```
R2#configure terminal
R2(config)#enable password cisco
```

//以上把密码改为自己的密码，如果还配置别的密码则一起把它们修改了。

```
R2(config)#config-register 0x2102
```

//以上把寄存器的值恢复为正常值 0x2102

```
R2(config)#exit
R2#copy running-config startup-config
```

```
Destination filename [startup-config]?
Building configuration...
[OK]
R2#reload
//以上是保存配置，重启路由器，检查路由器是否正常
```

【提示】在保存配置前，还需要把路由器的各个接口一一打开

（3） 步骤 3：故意删除 flash 中的 IOS，我们要恢复 IOS

```
R2#show flash:
CompactFlash directory:
File  Length   Name/status
  1   41205996  c2800nm-adventerprisek9-mz.124-11.T1.bin
[41206060 bytes used, 23019216 available, 64225276 total]
62720K bytes of ATA CompactFlash (Read/Write)
//显示 flash 中的 IOS
R2#delete flash:c2800nm-adventerprisek9-mz.124-11.T1.bin
Delete filename [c2800nm-adventerprisek9-mz.124-11.T1.bin]?
Delete flash:c2800nm-adventerprisek9-mz.124-11.T1.bin? [confirm]
//以上是删除 flash 的 IOS，模拟 FLASH 中的 IOS 丢失或者 IOS 升级失败
```

【提示】请慎重进行该步骤。如果工作中不慎误删 IOS，请不要将路由器关机，可以直接使用“copy tftp flash”命令从 TFTP 服务器恢复 IOS，这比起我们下面介绍的方法简单得多。除了从 TFTP 恢复 IOS，还可以用 Xmodem 方式通过 console 口恢复 IOS，然而由于 console 的速度很慢，很少有人采用。

（4） 步骤 4：恢复 IOS

请确认 IOS 已经放在 c:\ TFTP-Root 目录下。路由器丢失了 IOS 后，开机将自动进入 rommon 模式。

```
rommon 2 > IP_ADDRESS=172.16.0.2
rommon 3 > IP_SUBNET_MASK=255.255.0.0
rommon 4 > DEFAULT_GATEWAY=172.16.0.100
rommon 5 > TFTP_SERVER=172.16.0.100
rommon 6 > TFTP_FILE=c2800nm-adventerprisek9-mz.124-11.T1.bin
```

//要恢复 IOS，需要配置一些变量的值，主要是路由器的 IP 地址、掩码等。由于路由器和 TFTP 服务器在同一网段，是不需要网关的，但是不能不配置该值，所以我们把 DEFAULT_GATEWAY 胡乱地指向了 TFTP 服务器。请注意变量名的大小写。

```
rommon 8 > tftpdnld
//开始从 tftp 恢复 IOS
           IP_ADDRESS: 172.16.0.2
      IP_SUBNET_MASK: 255.255.0.0
     DEFAULT_GATEWAY: 172.16.0.100
         TFTP_SERVER: 172.16.0.100
           TFTP_FILE: c2800nm-adventerprisek9-mz.124-11.T1.bin
        TFTP_VERBOSE: Progress
```

```
    TFTP_RETRY_COUNT: 18
        TFTP_TIMEOUT: 7200
       TFTP_CHECKSUM: Yes
        TFTP_MACADDR: 00:19:55:66:63:20
             GE_PORT: Gigabit Ethernet 0
       GE_SPEED_MODE: Auto

Invoke this command for disaster recovery only.
WARNING: all existing data in all partitions on flash will be lost!
Do you wish to continue? y/n:  [n]:  y
//回答“y”开始从 tftp 服务器上恢复 IOS，根据 IOS 的大小，通常需要十几分钟
Receiving              c2800nm-adventerprisek9-mz.124-11.T1.bin              from
172.16.0.100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
（此处省略）
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
File reception completed.
Validating checksum.
Copying file c2800nm-adventerprisek9-mz.124-11.T1.bin to flash.
Eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
//从 tftp 服务器接收了 IOS 后，会进行校验。
rommon 9 > i
//重启路由器
```

2.5 实验 4:CDP

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 查找 CDP 邻居

（2） 熟悉 CDP 的配置

2. 实验拓扑

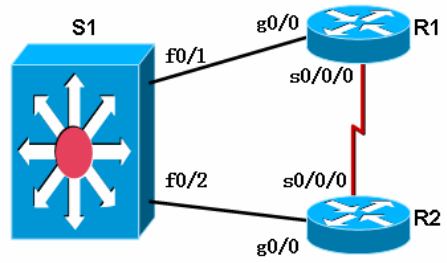

图 2-11 实验 4 拓扑图

3. 实验步骤

（1） 步骤 1：打开接口

```
R1(config)#int g0/0
R1(config-if)#no shutdown
R1(config-if)#int s0/0/0
R1(config-if)#no shutdown
```

R2(config)#int g0/0
R2(config-if)#no shutdown
R2(config-if)#int s0/0/0
R2(config-if)#no shutdown
R2(config-if)#clock rate 128000
//以上是打开路由器间互连的各个接口，而默认时交换机 S1 的所有接口就是打开的。
（2） 步骤：查看 CDP 配置
R1#show cdp
Global CDP information: Global CDP information:
Sending CDP packets every 60 seconds
Sending a holdtime value of 180 seconds
Sending CDPv2 advertisements is enabled
//默认时 CDP 是运行的，每 60 秒从接口发送 cdp 消息；发送出的 CDP 消息，邻居会为它保存 180 秒

R1#show cdp interface
GigabitEthernet0/0 is up, line protocol is up
Encapsulation ARPA
Sending CDP packets every 60 seconds
Holdtime is 180 seconds
GigabitEthernet0/1 is administratively down, line protocol is down
Encapsulation ARPA
Sending CDP packets every 60 seconds
Holdtime is 180 seconds
Serial0/0/0 is down, line protocol is down
Encapsulation HDLC
Sending CDP packets every 60 seconds
Holdtime is 180 seconds
Serial0/0/1 is administratively down, line protocol is down
Encapsulation HDLC
Sending CDP packets every 60 seconds
Holdtime is 180 seconds
//以上显示在哪些接口运行 CDP 协议
（3） 步骤 3：查看 CDP 邻居
R1#show cdp neighbors

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
R2	Ser 0/0/0	137	R S I	2821	Ser 0/0/0
S1	Gig 0/0	172	S I	WS-C3560-	Fas 0/1

//以上显示R1路由器有两个邻居:R2和S1。“Device ID”表示邻居的主机名;“Local Intrfce”表明 R1 通过该接口和邻居连接，注意是 R1 上的接口；“Holdtme”指收到邻居发送的 CDP 消息的时间，采用倒计时;“Capability”表明邻居是什么设备，第一、二行 Capability Codes 对各符号进行了说明；“Platform”指明了邻居的硬件型号；“Port ID”指明了 R1 是连接对方的哪个接口上。

R1#show cdp entry R2

Device ID: R2

Entry address(es):

Platform: Cisco 2821, Capabilities: Router Switch IGMP

Interface: Serial0/0/0, Port ID (outgoing port): Serial0/0/0

Holdtime : 158 sec

Version :

Cisco IOS Software, 2800 Software (C2800NM-ADVENTERPRISEK9-M), Version 12.4(11)T1, RELEASE SOFTWARE (fc5)

Technical Support: http://www.cisco.com/techsupport

Copyright (c) 1986-2007 by Cisco Systems, Inc.

Compiled Thu 25-Jan-07 12:50 by prod_rel_team

advertisement version: 2

VTP Management Domain: ''

//以上是显示邻居 R2 的详细信息，甚至可以知道邻居的 IOS 版本

R1#clear cdp table

//清除 CDP 表

（4） 步骤 4：关、启 CDP；调整 CDP 参数

R1(config)#int g0/0

R1(config-if)#no cdp enable

//以上是在 g0/0 接口上关闭 cdp，其他接口还运行 CDP

R1(config-if)#exit

R1(config)#no cdp run //以上是在整个路由器上关闭 cdp

R1(config)#cdp run //在整个路由器上打开 cdp

R1(config)#cdp timer 30 //调整 CDP 消息发送时间为 30 秒

R1(config)#cdp holdtime 120

//调整 CDP 消息的 holdtime 为 120 秒,对方收到该路由器发送的 CDP 消息后将保持 120 秒

R1#show cdp

Global CDP information:

Sending CDP packets every 30 seconds

Sending a holdtime value of 120 seconds

Sending CDPv2 advertisements is enabled

2.6 本章小结

本章介绍了路由器的硬件组成，介绍了 CLI 界面的几种模式、各种编辑命令。IOS 有着大量的命令，本章主要介绍路由器的基本初始化命令。还介绍了路由器配置文件保存、备份，密码的恢复，IOS 的备份和恢复。表 2-3 是本章出现的命令。

表 2-3 本章命令汇总

命令	作用
clock set	设置路由器的时间
show clock	显示路由器的时间
show history	显示历史命令
terminal no editing	关闭 CLI 的编辑功能
terminal editing	打开 CLI 的编辑功能
terminal history size 50	修改历史命令缓冲区的大小
copy running-config startup-config	把内存中的配置文件保存到 NVRAM 中
clock rate 128000	配置串口上的时钟(DCE 端)
show version	显示路由器的 IOS 版本等信息
show running-config	显示内存中的配置文件
show startup-config	显示 NVRAM 中的配置文件
show interface s0/0/0	显示接口的信息
show flash	显示 flash 的有关信息
show controllers s0/0/0	显示 s0/0/0 的控制器信息
show ip arp	显示路由器中的 arp 表
copy running-config tftp	把内存中的配置文件拷贝到 tftp 服务器上
copy tftp running-config	把 tftp 服务器上的配置文件拷贝到内存中
copy flash:c2800nm-adventerprisek9-mz.124-11.T1.bin tftp	把 flash 中的 IOS 拷贝到 tftp 服务器上
confreg 0x2142	在 rommon 模式下修改配置寄存器值
i	在 rommon 模式下重启路由器
copy startup-config running-config	把 NVRAM 中的配置文件拷贝到内存中
config-register 0x2102	修改配置寄存器值
reload	重启路由器
delete flash:c2800nm-adventerprisek9-mz.124-11.T1.bin	删除 flash 中的 IOS
copy tftp flash	从 tftp 服务器上拷贝 IOS 到 flash 中
tftpdnld	rommon 模式下，从 tftp 服务器下载 IOS
show cdp	显示 CDP 运行信息
show cdp interface	显示 CDP 在各接口的运行情况
show cdp neighbors	显示 CDP 邻居信息
show cdp entry R2	显示 CDP 邻居 R2 的详细信息
clear cdp table	清除 CDP 邻居表
no cdp enable	接口下关闭 CDP
no cdp run/ cdp run	关闭/打开整个路由器的 CDP
cdp timer 30	CDP 每 30 秒发送一次
cdp holdtime 120	让邻居为本设备发送的 CDP 消息保持 120 秒

第 3 章 静态路由

转发数据包是路由器的最主要功能。路由器转发数据包时需要查找路由表，管理员可以通过手工的方法在路由器中直接配置路由表，这就是静态路由。虽然静态路由不适合于在大的网络中使用，但是由于静态路由简单、路由器负载小、可控性强等原因，在许多场合中还经常被使用。本章将介绍静态路由的配置，同时为以后配置动态路由奠定基础。

3.1 静态路由与默认路由

3.1.1 静态路由介绍

路由器在转发数据时，要先在路由表（routing table）中查找相应的路由。路由器有这么三种途径建立路由：

（1） 直连网络：路由器自动添加和自己直接连接的网络的路由

（2） 静态路由：管理员手动输入到路由器的路由

（3） 动态路由：由路由协议（routing protocol）动态建立的路由

静态路由的缺点是不能动态反映网络拓扑，当网络拓扑发生变化时，管理员就必须手工改变路由表；然而静态路不会占用路由器太多的 CPU 和 RAM 资源，也不占用线路的带宽。如果出于安全的考虑想隐藏网络的某些部分或者管理员想控制数据转发路径，也会使用静态路由。在一个小而简单的网络中，也常使用静态路由，因为配置静态路由会更为简捷。

配置静态路由的命令为“ip route”，命令的格式如下：

ip route 目的网络 掩码 { 网关地址 | 接口 }

例子：**ip route 192.168.1.0 255.255.255.0 s0/0**

例子：**ip route 192.168.1.0 255.255.255.0 12.12.12.2**

在写静态路由时，如果链路是点到点的链路（例如 PPP 封装的链路），采用网关地址和接口都是可以的；然而如果链路是多路访问的链路（例如以太网），则只能采用网关地址，即不能：**ip route 192.168.1.0 255.255.255.0 f0/0** 。

【提示】有的 IOS 版本中，采用 **ip route 192.168.1.0 255.255.255.0 f0/0** 时，路由器也是正常工作的，然而这是代理 ARP 的功劳，建议不要采用该形式。

在路由器上，可以使用“**show ip route**”命令查看路由表。如下：

```
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set
R    172.16.0.0/16 [120/2] via 10.1.0.2, 00:00:21, Serial0/0
```

```
                [120/2] via 10.3.0.2, 00:00:06, Serial0/1
     10.0.0.0/16 is subnetted, 4 subnets
R       10.2.0.0 [120/1] via 10.1.0.2, 00:00:21, Serial0/0
C       10.3.0.0 is directly connected, Serial0/1
C       10.1.0.0 is directly connected, Serial0/0
R       10.4.0.0 [120/1] via 10.3.0.2, 00:00:06, Serial0/1
C    192.168.1.0/24 is directly connected, FastEthernet0/0
```

在输出中，首先显示路由条目各种类型的的简写，如：“C”为直连网络，“S”为静态路由。以上面带有下划的路由为例，“R”表示这条路由是“RIP”协议学习得到的；“10.2.0.0”是目的网络；“[120/1]”是管理距离（Administrative Distance ，AD）/ 度量值（Metric）；“via 10.1.0.2”是指到达目的网络的下一跳路由器的 IP 地址；“00:00:21”是指路由器最近一次得知路由到现在的时间；“Serials 0/0”是指到达下一跳应从哪个端口出去。

【技术要点】管理距离（AD）：用来表示路由的可信度，路由器可能从多种途径获得同一路由，例如：一个路由器要获得“10.2.0.0/24”网络的路由，可以来自 RIP，也可以是静态路由。不同途径获得的路由可能采取不同的路径到达目的网络，为了区别它们的可信度，用管理距离加以表示。表 3-1 是通过各种路由协议获得的路由的默认管理距离。路由表中管理距离值越小说明路由的可靠程度越高，静态路由的管理距离为 1，说明手工输入的路由优先级高于其他的路由。

表 3-1　路由协议的默认管理距离

路由协议	管理距离
直连接口	0
静态路由	1
外部 BGP	20
内部 EIGRP	90
IGRP	100
OSPF	110
RIP	120
外部 EIGRP	170
内部 BGP	200

【技术要点】度量值（Metric）：某一个路由协议判别到目的网络的最佳路径的方法。当一路由器有多条路径到达某一目的网络时，路由协议必须判断其中的哪一条是最佳的并把它放到路由表中，路由协议会给每一条路径计算出一个数，这个数就是度量值，通常这个值是没有单位的。度量值越小，这条路径越佳。然而不同的路由协议定义度量值的方法是不一样的，所以不同的路由协议选择出的最佳距离可能是不一样的。具体请参见路由协议的章节。

3.1.2 默认路由介绍

所谓的默认路由，是指路由器在路由表中如果找不到到达目的网络的具体路由时，最后会采用的路由。默认路由通常会在存根网络（Stub network，即只有一个出口的网络）中使用。如图 3-1，图中左边的网络到 Internet 上只有一个出口，因此可以在 R2 上配置默认路由。命令为：**ip route 0.0.0.0 0.0.0.0 { 网关地址 | 接口 }**

例子：**ip route 0.0.0.0 0.0.0.0 s0/0**

例子：**ip route 0.0.0.0 0.0.0.0 12.12.12.2**

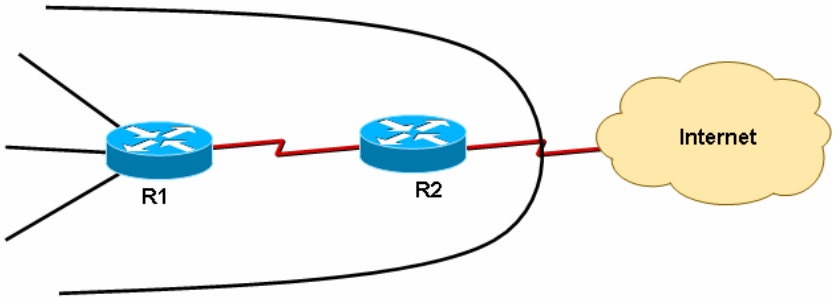

图 3-1 桩网络（Stub network）

3.1.3 ip classless

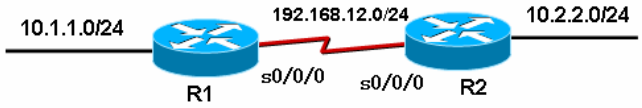

图 3-2 ip classless 示例

图 3-2 中，如果在 R1 上配置了默认路由：**ip route 0.0.0.0 0.0.0.0 s0/0/0**，则 R1 路由器是否会把到达 10.2.2.0/24 网络的数据从 s0/0/0 接口发送出去？这取决于是否执行了“**ip classless**”命令。如果执行了“**ip classless**”命令（实际上这是默认值），则路由器存在默认路由时，所有在路由表中查不到具体路由的数据包将通过默认路由发送。

如果执行了“**no ip classless**”命令，当路由器存在一主类网络的某一子网路由时，路由器将认为自己已经知道该主类网络的全部子网的路由，这时即使存在默认路由，到达该主类任一子网的数据包不会通过默认路由发送。图 3-2 中，执行了“**no ip classless**”后，由于 R1 路由器上有 10.0.0.0 的子网 10.1.1.0/24（这是直连路由），因此 R1 路由器收到到达 10.2.2.0/24 子网的数据包不会使用默认路由进行发送。然而如果数据包是要到达 20.2.2.0/24，默认路由会被采用，因为 R1 没有任何 20.0.0.0 子网的路由。

3.2 实验 1:静态路由

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 路由表的概念

（2） ip route 命令的使用

（3） 根据需求正确配置静态路由

2. 实验拓扑

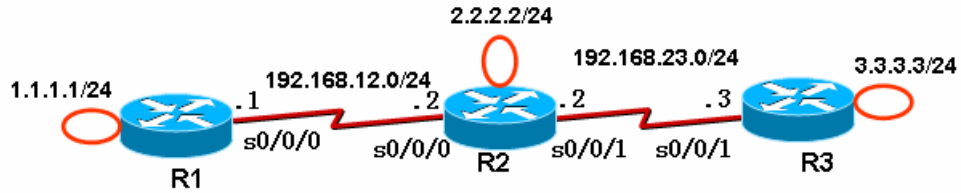

图 3-3 实验 1、实验 2 拓扑图

3. 实验步骤

我们要使得 1.1.1.0/24、2.2.2.0/24、3.3.3.0/24 网络之间能够互相通信。

（1） 步骤 1：在各路由器上配置 IP 地址、保证直连链路的连通性

```
R1(config)#int loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config)#int s0/0/0
R1(config-if)#ip address 192.168.12.1 255.255.255.0
R1(config-if)#no shutdown

R2(config)#int loopback0
R2(config-if)#ip address 2.2.2.2 255.255.255.0
R2(config)#int s0/0/0
R2(config-if)#clock rate 128000
R2(config-if)#ip address 192.168.12.2 255.255.255.0
R2(config-if)#no shutdown
R2(config)#int s0/0/1
R2(config-if)#clock rate 128000
R2(config-if)#ip address 192.168.23.2 255.255.255.0
R2(config-if)#no shutdown

R3(config)#int loopback0
R3(config-if)#ip address 3.3.3.3 255.255.255.0
R3(config)#int s0/0/1
R3(config-if)#ip address 192.168.23.3 255.255.255.0
R3(config-if)#no shutdown
```

（2） 步骤 2：R1 上配置静态路由

```
R1(config)#ip route 2.2.2.0 255.255.255.0 s0/0/0
```

//下一跳为接口形式，s0/0/0 是点对点的链路，注意应该是 R1 上的 s0/0/0 接口

```
R1(config)#ip route 3.3.3.0 255.255.255.0 192.168.12.2
```

//下一跳为 IP 地址形式，192.168.12.2 是 R2 上的 IP 地址

（3） 步骤 3：R2 上配置静态路由

```
R2(config)#ip route 1.1.1.0 255.255.255.0 s0/0/0
R2(config)#ip route 3.3.3.0 255.255.255.0 s0/0/1
```

（4） 步骤 4：R3 上配置静态路由

```
R3(config)#ip route 1.1.1.0 255.255.255.0 s0/0/1
R3(config)#ip route 2.2.2.0 255.255.255.0 s0/0/1
```

4. 实验调试

（1） 在 R1、R2、R3 上查看路由表

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     2.0.0.0/24 is subnetted, 1 subnets
S       2.2.2.0 is directly connected, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
S       3.3.3.0 [1/0] via 192.168.12.2
```

R2#show ip route

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
S       1.1.1.0 is directly connected, Serial0/0/0
     2.0.0.0/24 is subnetted, 1 subnets
C       2.2.2.0 is directly connected, Loopback0
     3.0.0.0/24 is subnetted, 1 subnets
S       3.3.3.0 is directly connected, Serial0/0/1
C    192.168.23.0/24 is directly connected, Serial0/0/1
```

R3#show ip route

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
     1.0.0.0/24 is subnetted, 1 subnets
S       1.1.1.0 is directly connected, Serial0/0/1
     2.0.0.0/24 is subnetted, 1 subnets
S       2.2.2.0 is directly connected, Serial0/0/1
     3.0.0.0/24 is subnetted, 1 subnets
```

```
C       3.3.3.0 is directly connected, Loopback0
C    192.168.23.0/24 is directly connected, Serial0/0/1
```

（2） 从各路由器的环回口 ping 其他路由器的环回口：

R1#ping

//不带任何参数的 ping 命令，允许我们输入更多的参数

```
Protocol [ip]:
Target IP address: 2.2.2.2        //目标 IP 地址
Repeat count [5]:                 //发送的 ping 次数
Datagram size [100]:              //ping 包的大小
Timeout in seconds [2]:           //超时时间
Extended commands [n]: y          //是否进一步扩展命令
Source address or interface: 1.1.1.1       //源 IP 地址
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/14/16 ms
```

//以上说明从 R1 的 loopback0 可以 ping 通 R2 上的 loopback0。也可以直接使用命令：

R1#ping 2.2.2.2 source loopback 0

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/14/16 ms
```

R2#ping 1.1.1.1 source loopback 0

R2#ping 3.3.3.3 source loopback 0

//从 R2 的 loopback0 应该可以 ping 通 R1 和 R3 的 lopback0 接口。

R3#ping 1.1.1.1 source loopback 0

R3#ping 2.2.2.2 source loopback 0

//从 R3 的 loopback0 也应该可以 ping 通 R1 和 R2 的 lopback0 接口。

【提示】虽然从 R1 的 loopback0 可以 ping 通 R3 的 loopback0，数据需要经过 192.168.23.0/24 网络，但是在 R1 上我们并没有添加 192.168.23.0/24 的路由。路由器转发数据包完成是根据路由表的，并且数据是一跳一跳地被转发的，就像接力赛似的。从 R1 的 loopback0 口 ping R3 的 loopback0 口时，IP 数据包的源 IP 为 1.1.1.1，目的 IP 为 3.3.3.3。R1 路由器首先查路由表，数据包被发到了 R2；R2 路由器也查路由表（3.3.3.0/24 路由），

数据包被发到了 R3；R3 知道这是直连路由。R3 响应 R1 的数据包进行类似的过程。

（3） 从 R1 上 ping 2.2.2.2、从 R1 上 ping 3.3.3.3

```
R1#ping 2.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/14/16 ms
```

//可以 ping 通。

```
R1#ping 3.3.3.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

//以上无法ping通，原因在于使用ping命令时，如果不指明源接口，则R1路由器使用s0/0/0接口的 IP 地址（192.168.12.1）作为 IP 数据包的源 IP 地址了。R3 上响应 R1 的数据包时，数据包是发向 192.168.12.1 的，然而由于 R3 没有 192.168.12.0/24 的路由，数据包无法发送。即：数据包从 R1 到了 R3 后，无法返回 R1。

3.3 实验 2:默认路由

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 默认路由的使用场合

（2） 默认路由的配置

2. 实验拓扑

如图 3-3。

3. 实验步骤

在实验 1 的基础上进行实验 2。

（1） 步骤 1：R1、R3 上删除原有静态路由

```
R1(config)#no ip route 2.2.2.0 255.255.255.0 Serial0/0/0
```

//要删除路由，在原有命令前面加 no 即可

```
R1(config)#no ip route 3.3.3.0 255.255.255.0 192.168.12.2

R3(config)#no ip route 1.1.1.0 255.255.255.0 Serial0/0/1
R3(config)#no ip route 2.2.2.0 255.255.255.0 Serial0/0/1
```

（2） 步骤 2： R1、R3 上配置默认路由

```
R1(config)#ip route 0.0.0.0 0.0.0.0 s0/0/0
R3(config)#ip route 0.0.0.0 0.0.0.0 s0/0/1
```

4. 实验调试

从各路由器的环回口 ping 其他路由器的环回口。请读者比较两个实验 ping 的结果，

仔细分析原因。

3.4 实验 3:ip classless

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） ip classless 命令的含义

（2） 配置 ip classless

2. 实验拓扑

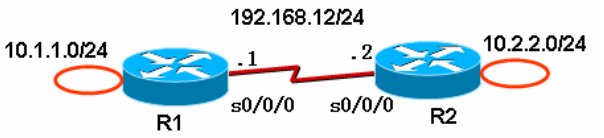

图 3-4 实验 3 拓扑图

3. 实验步骤

（1） 步骤 1：执行“ip classless”

```
R1(config)#interface Loopback0
R1(config-if)#ip address 10.1.1.1 255.255.255.0
R1(config)#interface Serial0/0/0
R1(config-if)#no shutdown
R1(config-if)#ip address 192.168.12.1 255.255.255.0
R1(config)#ip classless
R1(config)#ip route 0.0.0.0 0.0.0.0 Serial0/0/0
```

//以上我们配置了默认路由；同时打开“ip classless”，默认就是打开的。

```
R2(config)#interface Loopback0
R2(config-if)#ip address 10.2.2.2 255.255.255.0
R2(config)#interface Serial0/0/0
R1(config-if)#no shutdown
R2(config-if)#ip address 192.168.12.2 255.255.255.0
R2(config-if)#clock rate 128000
R1(config)#ip classless
R2(config)#ip route 10.1.1.0 255.255.255.0 Serial0/0/0
```

测试从 R1 ping R2 的 loopback0 接口

```
R1#ping 10.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/13/16 ms
```

可以 ping 通。

（2） 步骤 2：执行“no ip classless”

```
R1(config)#no ip cef
//关闭 ip cef，防止影响我们的测试，CEF
R1(config)#no ip classless

R1#ping 10.2.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

可以看到 R1 虽然存在默认路由，也不能可以 ping 通 R2 的 loopback0 接口。

3.5 本章小结

本章介绍了静态路由的配置方法，命令虽然简单，但是静态路由的配置却有一定的技巧性。我们也介绍了路由表的查看方法以及路由表中各字段的含义。默认路由常常用于桩网络中，处于局域网和 Internet 边界的路由器通常会使用它。表 3-2 是本章出现的命令。

表 3-2 本章命令汇总

命令	作用
ip route	配置静态路由
show ip route	查看路由表
ip classless/ no ip classless	打开/关闭有类路由功能
ping 2.2.2.2 source loopback 0	指定源端口进行 ping 测试

第 4 章　RIP

动态路由协议包括距离向量路由协议和链路状态路由协议。RIP（Routing Information Protocols，路由信息协议）是使用最广泛的距离向量路由协议。RIP 是为小型网络环境设计的，因为这类协议的路由学习及路由更新将产生较大的流量，占用过多的带宽。

4.1　RIP 概述

RIP 是由 Xerox 在 70 年代开发的,最初定义在 RFC1058 中。RIP 用两种数据包传输更新：更新和请求，每个有 RIP 功能的路由器默认情况下每隔 30 秒利用 UDP 520 端口向与它直连的网络邻居广播（RIP v1）或组播（RIP v2）路由更新。因此路由器不知道网络的全局情况，如果路由更新在网络上传播慢，将会导致网络收敛较慢，造成路由环路。为了避免路由环路，RIP 采用水平分割、毒性逆转、定义最大跳数、闪式更新、抑制计时 5 个机制来避免路由环路。

RIP 协议分为版本 1 和版本 2。不论是版本 1 或版本 2，都具备下面的特征：

1. 是距离向量路由协议；
2. 使用跳数（Hop Count）作为度量值；
3. 默认路由更新周期为 30 秒；
4. 管理距离（AD）为 120；
5. 支持触发更新；
6. 最大跳数为 15 跳；
7. 支持等价路径,默认 4 条,最大 6 条；
8. 使用 UDP520 端口进行路由更新。

而 RIPv1 和 RIPv2 的区别如表 4-1。

表 4-1 RIPv1 和 RIPv2 的区别

RIPv1	RIPv2
在路由更新的过程中不携带子网信息	在路由更新的过程中携带子网信息
不提供认证	提供明文和 MD5 认证
不支持 VLSM 和 CIDR	支持 VLSM 和 CIDR
采用广播更新	采用组播（224.0.0.9）更新
有类别（Classful）路由协议	无类别（Classless）路由协议

4.2　RIPv1

4.2.1 实验 1：RIPv1 基本配置

1.实验目的

通过本实验可以掌握：

（1）在路由器上启动 RIPv1 路由进程

（2）启用参与路由协议的接口，并且通告网络

（3）理解路由表的含义

（4）查看和调试 RIPv1 路由协议相关信息

2. 拓扑结构

实验拓扑如图 4-1 所示。

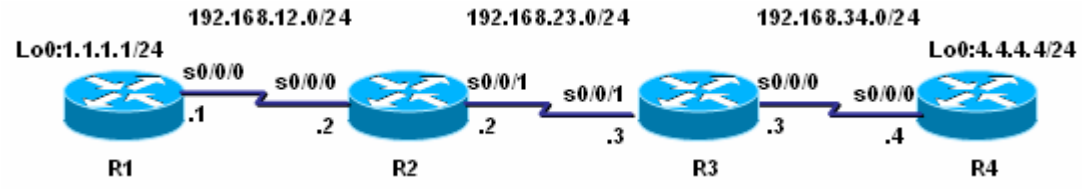

图 4-1 RIPv1 的基本配置

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router rip                          //启动 RIP 进程
R1(config-router)#version 1                    //配置 RIP 版本 1
R1(config-router)#network 1.0.0.0              //通告网络
R1(config-router)#network 192.168.12.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router rip
R2(config-router)#version 1
R2(config-router)#network 192.168.12.0
R2(config-router)#network 192.168.23.0
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router rip
R3(config-router)#version 1
R3(config-router)#network 192.168.23.0
R3(config-router)#network 192.168.34.0
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router rip
R4(config-router)#version 1
R4(config-router)#network 192.168.34.0
R4(config-router)#network 4.0.0.0
```

4. 实验调试

(1) show ip route

该命令用来查看路由表。

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
```

```
C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
R    4.0.0.0/8 [120/3] via 192.168.12.2, 00:00:03, Serial0/0/0
R    192.168.23.0/24 [120/1] via 192.168.12.2, 00:00:03, Serial0/0/0
R    192.168.34.0/24 [120/2] via 192.168.12.2, 00:00:03, Serial0/0/0
```

以上输出表明路由器 R1 学到了 3 条 RIP 路由，其中路由条目“R 4.0.0.0/8 [120/3] via 192.168.12.2, 00:00:03, Serial0/0/0”的含义如下：

① R：路由条目是通过 RIP 路由协议学习来的；

② 4.0.0.0/8：目的网络；

③ 120：RIP 路由协议的默认管理距离；

④ 3：度量值，从路由器 R1 到达网络 4.0.0.0/8 的度量值为 3 跳；

⑤ 192.168.12.2：下一跳地址；

⑥ 00:00:03：距离下一次更新还有 27（30-3）秒；

⑦ Serial0/0/0：接收该路由条目的本路由器的接口。

同时通过该路由条目的掩码长度可以看到，RIPv1 确实不传递子网信息。

（2）show ip protocols

该命令查看 IP 路由协议配置和统计信息。

```
R1#show ip protocols
```

【注意】

“//”后的信息表示注释，不是输出内容。

```
Routing Protocol is "rip"
//路由器上运行的路由协议是 RIP
  Outgoing update filter list for all interfaces is not set
//在出方向上没有设置过滤列表
  Incoming update filter list for all interfaces is not set
//在入方向上没有设置过滤列表
  Sending updates every 30 seconds, next due in 23 seconds
//更新周期是 30 秒，距离下次更新还有 23 秒
```

【注意】

为了防止更新同步,RIP 会以 15%的误差发送更新,即实际发送更新的周期的范围是 25.5-30 秒。

```
  Invalid after 180 seconds, hold down 180, flushed after 240
//invalid after：路由条目如果在 180 秒还没有收到更新，则被标记为无效
```

【技术要点】

被标记为无效的路由条目类似如下所示：

```
R    4.0.0.0/8 is possibly down, routing via 192.168.12.2, Serial0/0/0
```

可以通过很多方式使路由条目进入无效周期，例如在接口上加拒绝接收 UDP520 端口的

ACL，还比如将接口设置为被动接口等。

//**hold down**：抑制计时器的时间为 180 秒
//**flushed after**：路由条目如果在 240 秒还没有收到更新,则从路由表中删除此路由条目

【提示】
可以通过下面的命令来调整以上三个时间参数：
R1(config-router)#**timers basic** *update invalid holddown flushed*

```
  Redistributing: rip
//只运行 RIP 协议，没有其它的协议重分布进来
  Default version control: send version 1, receive version 1
//默认发送版本 1 的路由更新，接收本版 1 的路由更新
    Interface             Send  Recv  Triggered RIP  Key-chain
    Serial0/0/0           1     1
    Loopback0             1     1
//以上三行显示了运行 RIP 协议的接口，以及可以接收和发送的 RIP 路由更新的版本
  Automatic network summarization is in effect
//RIP 路由协议默认开启自动汇总功能
  Maximum path: 4
//RIP 路由协议可以支持 4 条等价路径，最大为 6 条
```

【提示】
可以通过下面的命令来修改 RIP 路由协议支持等价路径的条数：
R1(config-router)#**maximum-paths** *number-paths*

```
  Routing for Networks:
    1.0.0.0
    192.168.12.0
//以上三行表明 RIP 通告的网络
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.12.2         120      00:00:03
//以上三行表明路由信息源，其中：
//gateway：学习路由信息的路由器的接口地址，也就是下一跳地址
//distance：管理距离
//last update：更新发生在多长时间以前
  Distance: (default is 120)
//默认管理距离是 120
```

(3) debug ip rip

该命令可以查看 RIP 路由协议的动态更新过程。

```
R1#clear ip route *
R1#debug ip rip
Feb  9 12:43:13.311: RIP: sending request on Serial0/0/0 to 255.255.255.255
```

```
Feb  9 12:43:13.315: RIP: sending request on Loopback0 to 255.255.255.255
Feb  9 12:43:13.323: RIP: received v1 update from 192.168.12.2 on Serial0/0/0
Feb  9 12:43:13.323:      4.0.0.0 in 3 hops
Feb  9 12:43:13.323:      192.168.23.0 in 1 hops
Feb  9 12:43:13.323:      192.168.34.0 in 2 hops
Feb  9 12:43:15.311: RIP: sending v1 flash update to 255.255.255.255 via Loopback0 (1.1.1.1)
Feb  9 12:43:15.311: RIP: build flash update entries
Feb  9 12:43:15.311:    network 4.0.0.0 metric 4
Feb  9 12:43:15.311:    network 192.168.12.0 metric 1
Feb  9 12:43:15.311:    network 192.168.23.0 metric 2
Feb  9 12:43:15.311:    network 192.168.34.0 metric 3
Feb  9 12:43:15.311: RIP: sending v1 flash update to 255.255.255.255 via Serial0/0/0
(192.168.12.1)
Feb  9 12:43:15.311: RIP: build flash update entries
Feb  9 12:43:15.311:    network 1.0.0.0 metric 1
```

通过以上输出，可以看到 RIPv1 采用广播更新（255.255.255.255），分别向 Loopback0 和 s0/0/0 发送路由更新，同时从 s0/0/0 接收三条路由更新，分别是 4.0.0.0，度量值是 3 跳；192.168.34.0，度量值是 2 跳；192.168.23.0，度量值是 1 跳。

【技术要点】

flash update（闪式更新）指的是当网络上某个路径的度量值发生变化，路由器立即发出更新信息，而不管是否到达常规路由信息更新的周期。

4.2.2 实验 2:被动接口与单播更新

1. 实验目的

通过本实验可以掌握：

（1）被动接口的含义、配置和应用场合

（2）单播更新的应用场合和配置

2. 拓扑结构

实验拓扑如图 4-2 所示。

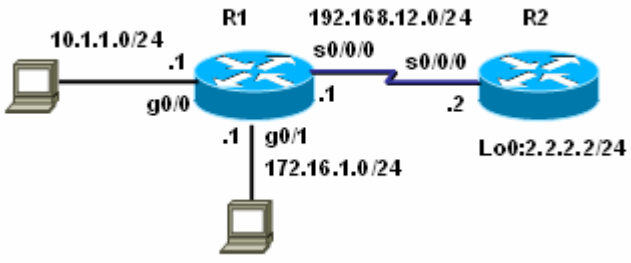

图 4-2 配置被动接口

由于以太口 g0/0 和 g0/1 连接主机，不需要向这些接口发送路由更新，所以可以考虑将路由器的该接口设置为被动接口。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router rip
R1(config-router)#version 1
R1(config-router)#network 10.0.0.0
R1(config-router)#network 172.16.0.0
R1(config-router)#network 192.168.12.0
R1(config-router)#passive-interface GigabitEthernet0/0
R1(config-router)#passive-interface GigabitEthernet0/1
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router rip
R2(config-router)#version 1
R2(config-router)#network 192.168.12.0
R2(config-router)#network 2.0.0.0
```

4.实验调试

```
R1#debug ip rip
R1#clear ip route *
Feb  9 13:24:41.275: RIP: sending request on Serial0/0/0 to 255.255.255.255
Feb  9 13:24:41.283: RIP: received v1 update from 192.168.12.2 on Serial0/0/0
Feb  9 13:24:41.283:      2.0.0.0 in 1 hops
Feb  9 13:24:43.275: RIP: sending v1 flash update to 255.255.255.255 via Serial0/0/0
(192.168.12.1)
Feb  9 13:24:43.275: RIP: build flash update entries
Feb  9 13:24:43.275:    network 10.0.0.0 metric 1
Feb  9 13:24:43.275:    network 172.16.0.0 metric 1
```

从以上输出可以看出，路由器 R1 确实不向被动接口 g0/0 和 g0/1 发送路由更新。

【技术要点】

被动接口只能接收路由更新，不能以广播或组播方式发送更新，但是可以以单播的方式发送更新，配置单播更新的命令如下：

```
R1(config-router)#neighbor A.B.C.D
```

【实例】

如图 4-3 所示，路由器 R1 只想把路由更新送到路由器 R3 上，由于 RIPv1 路由协议采用广播更新，默认情况下，路由更新将发送给以太网上任何一个设备，为了防止这种情况发生，把路由器 R1 的 g0/0 配置成被动接口，然而路由器 R1 还想把路由更新发送给 R3，这时候必须采用单播更新，为指定的相邻路由器 R3 发送路由更新。路由器 R1 具体的配置如下：

```
R1(config)#router rip
R1(config-router)#passive-interface GigabitEthernet0/0
R1(config-router)#neighbor 172.16.1.3
```

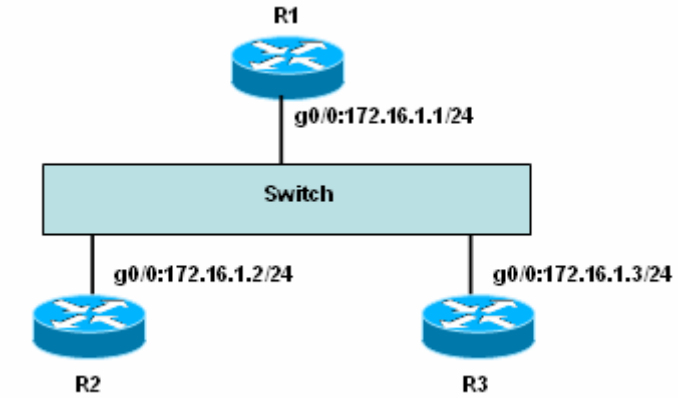

图 4-3 配置单播更新

4.2.3 实验 3：使用子网地址

1.实验目的

通过本实验可以掌握：

（1）RIPv1 使用子网地址的条件

（2）RIPv1 接收子网路由的原则

2.拓扑结构

实验拓扑如图 4-4 所示。

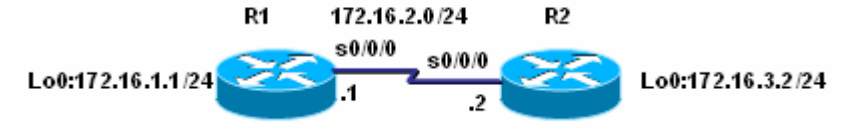

图 4-4 RIPv1 使用子网地址

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router rip
R1(config-router)#version 1
R1(config-router)#network 172.16.0.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router rip
R2(config-router)#version 1
R2(config-router)#network 172.16.0.0
```

4.实验调试

分别查看 R1、R2 的路有表：

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, Loopback0
C       172.16.2.0 is directly connected, Serial0/0/0
R       172.16.3.0 [120/1] via 172.16.2.2, 00:00:03, Serial0/0/0
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 3 subnets
R       172.16.1.0 [120/1] via 172.16.2.1, 00:00:21, Serial0/0/0
C       172.16.2.0 is directly connected, Serial0/0/0
C       172.16.3.0 is directly connected, Loopback0
```

从路由器 R1 和 R2 的路由表输出可以看出，它们互相学习到了 24 位的路由条目，从而可以说明，某些情况下 RIPv1 更新确实可以携带子网信息。

【技术要点】

RIPv1 路由更新可以携带子网信息必须同时满足两个条件：

（1）整个网络所有地址在同一个主类网络；

（2）子网掩码长度必须相同。

【思考】

假如在图 4-4 中，路由器 R2 的 s0/0/0 接口的 IP 地址的掩码长度为 25 位，那么，R2 的路由表是怎样的呢？结果如下：

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
```

```
Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
R       172.16.1.0/25 [120/1] via 172.16.2.1, 00:00:17, Serial0/0/0
C       172.16.2.0/25 is directly connected, Serial0/0/0
C       172.16.3.0/24 is directly connected, Loopback0
```

由此得出 RIP v1 接收子网路由的原则：如果路由器收到的是子网路由条目,那么就以接收该路由条目的接口的掩码长度作为该子网路由条目的掩码长度。

4.3 RIPv2

4.3.1 实验 4：RIPv2 基本配置

1.实验目的

通过本实验可以掌握：

（1）在路由器上启动 RIPv2 路由进程

（2）启用参与路由协议的接口，并且通告网络

（3）auto-summary 的开启和关闭

（4）查看和调试 RIPv2 路由协议相关信息

2.拓扑结构

实验拓扑如图 4-1 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 1.0.0.0
R1(config-router)#network 192.168.12.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.12.0
R2(config-router)#network 192.168.23.0
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#no auto-summary
R3(config-router)#network 192.168.23.0
R3(config-router)#network 192.168.34.0
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router rip
R4(config-router)#version 2
R4(config-router)#no auto-summary
```

R4(config-router)#network 192.168.34.0

R4(config-router)#network 4.0.0.0

4.实验调试

(1) show ip route

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     4.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
R       4.4.4.0/24 [120/3] via 192.168.12.2, 00:00:22, Serial0/0/0
R    192.168.23.0/24 [120/1] via 192.168.12.2, 00:00:22, Serial0/0/0
R    192.168.34.0/24 [120/2] via 192.168.12.2, 00:00:22, Serial0/0/0
```

从上面输出的路由条目“4.4.4.0/24”，可以看到RIPv2路由更新是携带子网信息的。

(2) show ip protocols

```
R1#show ip protocols
Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 19 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    Serial0/0/0           2     2
    Loopback0             2     2
// RIPv2默认情况下只接收和发送版本2的路由更新
```

【提示】

可以通过命令“ip rip send version”和“ip rip receive version”来控制在路由器接口上接收和发送的版本，例如在s0/0/0接口上接收版本1和2的路由更新，但是只发送版本2的路由更新，配置如下：

R1(config-if)#ip rip send version 2

R1(config-if)#ip rip receive version 1 2

【注意】

接口特性是优于进程特性的，对于本实验，虽然在 RIP 进程中配置了“version 2”，但是如果在接口上配置了“ip rip receive version 1 2”，则该接口可以接收版本 1 和 2 的路由更新。

```
Automatic network summarization is not in effect
Maximum path: 4
Routing for Networks:
  1.0.0.0
  192.168.12.0
Routing Information Sources:
  Gateway         Distance      Last Update
  192.168.12.2         120      00:00:26
Distance: (default is 120)
```

4.3.2 实验 5：RIPv2 手工汇总

1. 实验目的

通过本实验可以掌握：

（1）RIPv2 路由的手工汇总

（2）RIPv2 不支持 CIDR 汇总

（3）RIPv2 可以传递 CIDR 汇总

2. 拓扑结构

实验拓扑如图 4-5 所示。

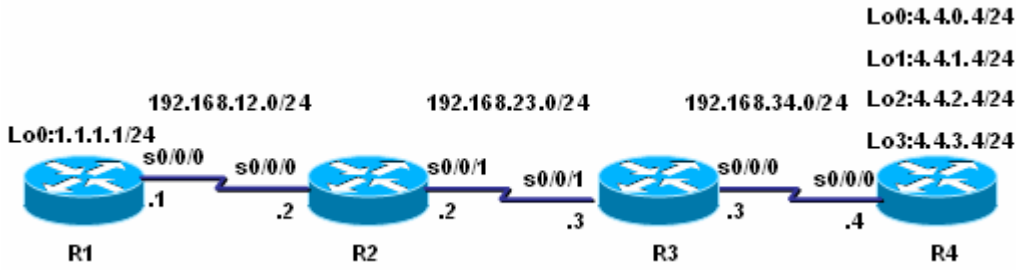

图 4-5 RIPv2 路由手工汇总

3. 实验步骤

路由器 R1、R2 和 R3 的配置和 **4.3.1 实验 4** 相同，R4 的配置如下：

```
R4(config)#router rip
R4(config-router)#version 2
R4(config-router)#no auto-summary
R4(config-router)#network 192.168.34.0
R4(config-router)#network 4.0.0.0
R4(config)#interface s0/0/0
R4(config-if)#ip summary-address rip 4.4.0.0 255.255.252.0//RIP 手工路由汇总
```

4. 实验调试

（1）在没有执行汇总之前路由器 R1 的路由表如下：

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
```

```
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     4.0.0.0/24 is subnetted, 4 subnets
R       4.4.0.0 [120/3] via 192.168.12.2, 00:00:21, Serial0/0/0
R       4.4.1.0 [120/3] via 192.168.12.2, 00:00:21, Serial0/0/0
R       4.4.2.0 [120/3] via 192.168.12.2, 00:00:12, Serial0/0/0
R       4.4.3.0 [120/3] via 192.168.12.2, 00:00:05, Serial0/0/0
R    192.168.23.0/24 [120/1] via 192.168.12.2, 00:00:21, Serial0/0/0
R    192.168.34.0/24 [120/2] via 192.168.12.2, 00:00:22, Serial0/0/0
```

从上面的输出看到路由器 R1 的路由表中有 R4 的 4 条环回接口的明细路由。

（2）在执行汇总以后路由器 R1 的路由表如下：

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     4.0.0.0/22 is subnetted, 1 subnets
R       4.4.0.0 [120/3] via 192.168.12.2, 00:00:21, Serial0/0/0
R    192.168.23.0/24 [120/1] via 192.168.12.2, 00:00:21, Serial0/0/0
R    192.168.34.0/24 [120/2] via 192.168.12.2, 00:00:22, Serial0/0/0
```

上面的输出表明在路由器 R1 的路由表中接收到汇总路由，当然 R2，R3 上也能收到汇总路由。

【思考】

现在将路由器 R4 上四个环回接口 lo0-lo4 的地址分别修改为 192.168.96.4/24,

192.168.97.4/24, 192.168.98.4/24, 192.168.99.4/24,在 s0/0/0 接口下还能够实现路由汇总吗？R4 上做如下的配置：

```
R4(config-if)#router rip
R4(config-router)#network 192.168.96.0
R4(config-router)#network 192.168.97.0
R4(config-router)#network 192.168.98.0
R4(config-router)#network 192.168.99.0
R4(config-if)#ip summary-address rip 192.168.96.0 255.255.252.0
```

路由器会提示如下信息：

“Summary mask must be greater or equal to major net”

显示的提示信息表明汇总后的掩码长度必须要大于或等于主类网络的掩码程度，因为“22<24”,所以不能汇总。

所以 **RIPv2 不支持 CIDR 汇总**，但是可以传递 CIDR 汇总。

解决方案如下：

（1）用静态路由发布被汇总的路由

```
R4(config)#ip route 192.168.96.0 255.255.252.0 null0
```

（2）将静态路由重分布到 RIP 网络中

```
R4(config)#router rip
R4(config-router)#redistribute static     //将静态路由重分布到 RIP 路由协议中
R4(config-router)#no network 192.168.96.0
R4(config-router)#no network 192.168.97.0
R4(config-router)#no network 192.168.98.0
R4(config-router)#no network 192.168.99.0
```

（3）在路由器 R1 上查看路由表

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
R    192.168.23.0/24 [120/1] via 192.168.12.2, 00:00:18, Serial0/0/0
R    192.168.34.0/24 [120/2] via 192.168.12.2, 00:00:18, Serial0/0/0
R    192.168.96.0/22 [120/3] via 192.168.12.2, 00:00:18, Serial0/0/0
```

通过输出不难看出 RIPv2 是可以传递 CIDR 汇总信息的。

4.3.3 实验 6：RIPv2 认证和触发更新

1.实验目的

通过本实验可以掌握：

（1）RIPv2 明文认证的配置和匹配原则

（2）RIPv2 MD5 认证的配置和匹配原则

（3）RIPv2 触发更新

2.拓扑结构

实验拓扑如图 4-1 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#key chain test  //配置钥匙链
R1(config-keychain)#key 1  //配置 KEY ID
R1(config-keychain-key)#key-string cisco   //配置 KEY ID 的密匙
R1(config)#interface s0/0/0
R1(config-if)#ip rip authentication mode text
//启用认证，认证模式为明文，默认认证模式就是明文，所以也可以不用指定
R1(config-if)#ip rip authentication key-chain test  //在接口上调用钥匙链
R1(config-if)#ip rip triggered                      //在接口上启用触发更新
```

（2）步骤 2：配置路由器 R2

```
R2(config)#key chain test
R2(config-keychain)#key 1
R2(config-keychain-key)#key-string cisco
R2(config)#interface s0/0/0
R2(config-if)#ip rip triggered
R2(config-if)#ip rip authentication key-chain test
R2(config-if)#interface s0/0/1
R2(config-if)#ip rip authentication key-chain test
R2(config-if)#ip rip triggered
```

（3）步骤 3：配置路由器 R3

```
R3(config)#key chain test
R3(config-keychain)#key 1
R3(config-keychain-key)#key-string cisco
R3(config)#interface s0/0/0
R3(config-if)#ip rip authentication key-chain test
R3(config-if)#ip rip triggered
R3(config-if)#interface s0/0/1
R3(config-if)#ip rip authentication key-chain test
R3(config-if)#ip rip triggered
```

（4）步骤 4：配置路由器 R4

```
R4(config)#key chain test
R4(config-keychain)#key 1
R4(config-keychain-key)#key-string cisco
R4(config)#interface s0/0/0
R4(config-if)#ip rip authentication key-chain test
R4(config-if)#ip rip triggered
```

4.实验调试

（1）show ip protocols

```
R2#show ip protocols
Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 4 seconds
  Invalid after 180 seconds, hold down 0, flushed after 240
// 由于触发更新，hold down 计时器自动为 0
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    Serial0/0/0           2     2          Yes       test
    Serial0/0/1           2     2          Yes       test
//以上两行表明 s0/0/0 和 s0/0/1 接口启用了认证和触发更新
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    192.168.12.0
    192.168.23.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.12.1         120      00:26:10
    192.168.23.3         120      00:26:01
  Distance: (default is 120)
```

（2）debug ip rip

```
R2#debug ip rip
RIP protocol debugging is on
R2#clear ip route *
*Feb 11 13:51:31.827: RIP: sending triggered request on Serial0/0/0 to 224.0.0.9
*Feb 11 13:51:31.831: RIP: sending triggered request on Serial0/0/1 to 224.0.0.9
*Feb 11 13:51:31.843: RIP: sending triggered request on Serial0/0/0 to 224.0.0.9
*Feb 11 13:51:31.847: RIP: sending triggered request on Serial0/0/1 to 224.0.0.9
*Feb 11 13:51:31.847: RIP: send v2 triggered flush update to 192.168.12.1 on Serial0/0/0 with
no route
*Feb 11 13:51:31.851: RIP: start retransmit timer of 192.168.12.1
*Feb 11 13:51:31.855: RIP: send v2 triggered flush update to 192.168.23.3 on Serial0/0/1 with
no route
*Feb 11 13:51:31.855: RIP: start retransmit timer of 192.168.23.3
*Feb 11 13:51:32.019: RIP: received packet with text authentication cisco
*Feb 11 13:51:32.019: RIP: received v2 triggered update from 192.168.12.1 on Serial0/0/0
*Feb 11 13:51:32.023: RIP: sending v2 ack to 192.168.12.1 via Serial0/0/0 (192.168.12.2),
     flush, seq# 1
*Feb 11 13:51:32.027:      1.1.1.0/24 via 0.0.0.0 in 1 hops
```

```
*Feb 11 13:51:32.031: RIP: received packet with text authentication cisco
*Feb 11 13:51:32.035: RIP: received v2 triggered update from 192.168.23.3 on Serial0/0/1
*Feb 11 13:51:32.035: RIP: sending v2 ack to 192.168.23.3 via Serial0/0/1(192.168.23.2),
     flush, seq# 2
*Feb 11 13:51:32.039:      192.168.34.0/24 via 0.0.0.0 in 1 hops
*Feb 11 13:51:32.043:      4.4.4.0/24 via 0.0.0.0 in 2 hops
*Feb 11 13:51:32.071: RIP: received packet with text authentication cisco
*Feb 11 13:51:32.071: RIP: received v2 triggered update from 192.168.23.3 on Serial0/0/1
*Feb 11 13:51:32.071: RIP: sending v2 ack to 192.168.23.3 via Serial0/0/1(192.168.23.2),
     flush, seq# 3
*Feb 11 13:51:32.075:      192.168.34.0/24 via 0.0.0.0 in 1 hops
*Feb 11 13:51:32.079:      4.4.4.0/24 via 0.0.0.0 in 2 hops
*Feb 11 13:51:32.083: RIP: received packet with text authentication cisco
*Feb 11 13:51:32.083: RIP: received v2 triggered ack from 192.168.23.3 on Serial0/0/1
     flush seq# 2
*Feb 11 13:51:32.087: RIP: send v2 triggered update to 192.168.23.3 on Serial0/0/1
*Feb 11 13:51:32.087: RIP: build update entries
*Feb 11 13:51:32.091:   route 176: 192.168.12.0/24 metric 1, tag 0
*Feb 11 13:51:32.091:   route 181: 1.1.1.0/24 metric 2, tag 0
*Feb 11 13:51:32.095: RIP: Update contains 2 routes, start 176, end 188
*Feb 11 13:51:32.095: RIP: start retransmit timer of 192.168.23.3
*Feb 11 13:51:32.099: RIP: received packet with text authentication cisco
*Feb 11 13:51:32.099: RIP: received v2 triggered update from 192.168.12.1 on Serial0/0/0
*Feb 11 13:51:32.103: RIP: sending v2 ack to 192.168.12.1 via Serial0/0/0 (192.168.12.2),
     flush, seq# 2
*Feb 11 13:51:32.107:      1.1.1.0/24 via 0.0.0.0 in 1 hops
*Feb 11 13:51:32.107: RIP: received packet with text authentication cisco
*Feb 11 13:51:32.111: RIP: received v2 triggered ack from 192.168.12.1 on Serial0/0/0
     flush seq# 3
*Feb 11 13:51:32.111: RIP: send v2 triggered update to 192.168.12.1 on Serial0/0/0
*Feb 11 13:51:32.115: RIP: build update entries
*Feb 11 13:51:32.115:   route 178: 192.168.23.0/24 metric 1, tag 0
*Feb 11 13:51:32.119:   route 184: 192.168.34.0/24 metric 2, tag 0
*Feb 11 13:51:32.123:   route 187: 4.4.4.0/24 metric 3, tag 0
*Feb 11 13:51:32.123: RIP: Update contains 3 routes, start 178, end 188
*Feb 11 13:51:32.123: RIP: start retransmit timer of 192.168.12.1
*Feb 11 13:51:32.263: RIP: received packet with text authentication cisco
*Feb 11 13:51:32.263: RIP: received v2 triggered ack from 192.168.23.3 on Serial0/0/1
     seq# 3
*Feb 11 13:51:32.267: RIP: received packet with text authentication cisco
*Feb 11 13:51:32.271: RIP: received v2 triggered ack from 192.168.12.1 on Serial0/0/0
     seq# 4
```

从上面的输出可以看出，在路由器 R2 上，虽然我们打开了 **debug ip rip**，但是由于采用触发更新，所以并没有看到每 30 秒更新一次的信息，而是清除了路由表这件事件触发了

路由更新。而且所有的更新中都有“triggered”的字样，同时在接收的更新中带有“text authentication”的字样，证明接口 s0/0/0 和 s0/0/1 启用了触发更新和明文认证。

（3）show ip rip database

该命令可以查看 RIP 数据库。

```
R2#show ip rip database
1.0.0.0/8    auto-summary
1.1.1.0/24
    [1] via 192.168.12.1, 00:12:22 (permanent), Serial0/0/0
  * Triggered Routes:
    - [1] via 192.168.12.1, Serial0/0/0
4.0.0.0/8    auto-summary
4.4.4.0/24
    [2] via 192.168.23.3, 00:12:22 (permanent), Serial0/0/1
  * Triggered Routes:
    - [2] via 192.168.23.3, Serial0/0/1
192.168.12.0/24    auto-summary
192.168.12.0/24    directly connected, Serial0/0/0
192.168.23.0/24    auto-summary
192.168.23.0/24    directly connected, Serial0/0/1
192.168.34.0/24    auto-summary
192.168.34.0/24
    [1] via 192.168.23.3, 00:12:22 (permanent), Serial0/0/1
  * Triggered Routes:
    - [1] via 192.168.23.3, Serial0/0/1
```

以上输出进一步说明了在 s0/0/0 和 s0/0/1 启用了触发更新。

（4）show run

```
R2#show run | begin router rip
router rip
 version 2
 timers basic 30 180 0 240
//由于触发更新，在配置中自动加入上面一行，且 hold down 计时器被设置为 0
 network 192.168.12.0
 network 192.168.23.0
 no auto-summary
```

关于 MD5 认证，只需要在接口下声明认证模式为 MD5 即可，例如在 R1 上的配置如下：

```
R1(config)#key chain test  //定义钥匙链
R1(config-keychain)#key 1
R1(config-keychain-key)#key-string cisco
R1(config)#interface s0/0/0
R1(config-if)#ip rip authentication mode md5 //认证模式为 MD5
R1(config-if)#ip rip authentication key-chain test
```

其他的配置和明文认证相同，这里不再赘述。当在 R2 上执行“debug ip rip”时显示

类似如下的信息：

*Feb 11 14:04:36.851: RIP: sending **triggered request** on Serial0/0/0 to 224.0.0.9

*Feb 11 14:04:36.855: RIP: sending **triggered request** on Serial0/0/1 to 224.0.0.9

*Feb 11 14:04:36.867: RIP: sending triggered request on Serial0/0/0 to 224.0.0.9

*Feb 11 14:04:36.871: RIP: sending triggered request on Serial0/0/1 to 224.0.0.9

*Feb 11 14:04:36.871: RIP: send v2 **triggered flush update** to 192.168.12.1 on Serial0/0/0 with no route

*Feb 11 14:04:36.875: RIP: start retransmit timer of 192.168.12.1

*Feb 11 14:04:36.875: RIP: send v2 triggered flush update to 192.168.23.3 on Serial0/0/1 with no route

*Feb 11 14:04:36.879: RIP: start retransmit timer of 192.168.23.3

*Feb 11 14:04:36.927: RIP: received packet with **MD5 authentication**

*Feb 11 14:04:36.931: RIP: received v2 triggered update from 192.168.23.3 on Serial0/0/1

*Feb 11 14:04:36.931: RIP: sending v2 **ack** to 192.168.23.3 via Serial0/0/1 (192.168.23.2), flush, seq# 4

*Feb 11 14:04:36.935: 192.168.34.0/24 via 0.0.0.0 in 1 hops

*Feb 11 14:04:36.943: 4.4.4.0/24 via 0.0.0.0 in 2 hops

*Feb 11 14:04:36.947: RIP: received packet with MD5 authentication

*Feb 11 14:04:36.947: RIP: received v2 triggered update from 192.168.12.1 on Serial0/0/0

*Feb 11 14:04:36.951: RIP: sending v2 ack to 192.168.12.1 via Serial0/0/0 (192.168.12.2), flush, seq# 3

*Feb 11 14:04:36.955: 1.1.1.0/24 via 0.0.0.0 in 1 hops

*Feb 11 14:04:36.959: RIP: received packet with MD5 authentication

*Feb 11 14:04:36.959: RIP: received v2 triggered update from 192.168.12.1 on Serial0/0/0

*Feb 11 14:04:36.963: RIP: sending v2 ack to 192.168.12.1 via Serial0/0/0 (192.168.12.2), flush, seq# 4

*Feb 11 14:04:36.967: 1.1.1.0/24 via 0.0.0.0 in 1 hops

*Feb 11 14:04:36.967: RIP: received packet with MD5 authentication

*Feb 11 14:04:36.971: RIP: received v2 triggered ack from 192.168.12.1 on Serial0/0/0 flush seq# 5

*Feb 11 14:04:36.971: RIP: send v2 triggered update to 192.168.12.1 on Serial0/0/0

*Feb 11 14:04:36.975: RIP: build update entries

*Feb 11 14:04:36.975: route 191: 192.168.23.0/24 metric 1, tag 0

*Feb 11 14:04:36.979: route 194: 192.168.34.0/24 metric 2, tag 0

*Feb 11 14:04:36.979: route 197: 4.4.4.0/24 metric 3, tag 0

*Feb 11 14:04:36.983: RIP: Update contains 3 routes, start 191, end 201

*Feb 11 14:04:36.983: RIP: start retransmit timer of 192.168.12.1

*Feb 11 14:04:36.991: RIP: received packet with MD5 authentication

*Feb 11 14:04:36.991: RIP: received v2 triggered update from 192.168.23.3 on Serial0/0/1

*Feb 11 14:04:36.991: RIP: sending v2 ack to 192.168.23.3 via Serial0/0/1 (192.168.23.2), flush, seq# 5

*Feb 11 14:04:36.999: 192.168.34.0/24 via 0.0.0.0 in 1 hops

*Feb 11 14:04:36.999: 4.4.4.0/24 via 0.0.0.0 in 2 hops

*Feb 11 14:04:37.003: RIP: received packet with MD5 authentication

```
*Feb 11 14:04:37.003: RIP: received v2 triggered ack from 192.168.23.3 on Serial0/0/1
     flush seq# 4
*Feb 11 14:04:37.007: RIP: send v2 triggered update to 192.168.23.3 on Serial0/0/1
*Feb 11 14:04:37.007: RIP: build update entries
*Feb 11 14:04:37.011:   route 189: 192.168.12.0/24 metric 1, tag 0
*Feb 11 14:04:37.015:   route 200: 1.1.1.0/24 metric 2, tag 0
*Feb 11 14:04:37.015: RIP: Update contains 2 routes, start 189, end 201
*Feb 11 14:04:37.019: RIP: start retransmit timer of 192.168.23.3
*Feb 11 14:04:37.059: RIP: received packet with MD5 authentication
*Feb 11 14:04:37.059: RIP: received v2 triggered ack from 192.168.12.1 on Serial0/0/0
     seq# 6
*Feb 11 14:04:37.067: RIP: received packet with MD5 authentication
*Feb 11 14:04:37.071: RIP: received v2 triggered ack from 192.168.23.3 on Serial0/0/1
     seq# 5
```

以上输出信息表明采用了 MD5 认证和触发更新。

【技术要点】

（1）在以太网接口下，不支持触发更新；

（2）触发更新需要协商，链路的两端都需要配置；

（3）在认证的过程中，如果定义多个 key ID，明文认证和 MD5 认证的匹配原则是不一样的：

① 明文认证的匹配原则是：

A. 发送方发送最小Key ID的密钥

B. 不携带Key ID号码

C. 接收方会和所有 Key Chain 中的密钥匹配，如果匹配成功，则通过认证。

【实例 1】

路由器 R1 有一个 Key ID,key1=cisco；

路由器 R2 有两个 Key ID,key1=ccie，key2=cisco

根据上面的原则，R1 认证失败，R2 认证成功，所以在 RIP 中，出现单边路由并不稀奇。

② MD5 认证的匹配原则是：

A. 发送方发送最小Key ID的密钥

B. 携带Key ID号码

C. 接收方首先会查找是否有相同的Key ID，如果有，只匹配一次，决定认证是否成功。如果没有该Key ID，只向下查找下一跳，匹配，认证成功；不匹配，认证失败。

【实例 2】

路由器 R1 有三个 Key ID,key1=cisco，key3=ccie,key5=cisco；

路由器 R2 有一个 Key ID,key2=cisco

根据上面的原则，R1 认证失败，R2 认证成功。

4.3.4 实验 7：浮动静态路由

1. 实验目的

通过本实验可以掌握浮动静态路由原理、配置以及备份应用。

2. 拓扑结构

实验拓扑如图 4-6 所示。

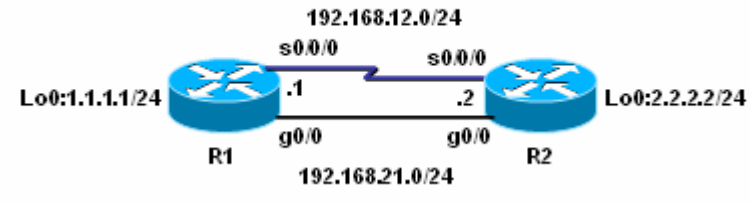

图 4-6 浮动静态路由

3. 实验步骤

本实验通过修改静态路由的管理距离为 130，使得路由器选路的时候优先选择 RIP，而静态路由作为备份。

（1）步骤 1：配置路由器 R1

```
R1(config)#ip route 2.2.2.0 255.255.255.0 192.168.12.2 130
//将静态路由的管理距离设置为 130
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 1.0.0.0
R1(config-router)#network 192.168.21.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#ip route 1.1.1.0 255.255.255.0 192.168.12.1 130
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.21.0
R2(config-router)#network 2.0.0.0
```

4. 实验调试

（1）在 R1 上查看路由表：

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
```

```
C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     2.0.0.0/24 is subnetted, 1 subnets
R       2.2.2.0 [120/1] via 192.168.21.2, 00:00:25, GigabitEthernet0/0
C    192.168.21.0/24 is directly connected, GigabitEthernet0/0
```

从以上输出可以看出，路由器将 RIP 的路由放入路由表中，因为 RIP 的管理距离为 120，小于在静态路由中设定的 130，而静态路由处于备份的地位。

（2）在 R1 上将 g0/0 接口 shutdown，然后查看路由表：

```
R1(config)#interface gigabitEthernet 0/0
R1(config-if)#shutdown
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     2.0.0.0/24 is subnetted, 1 subnets
S       2.2.2.0 [130/0] via 192.168.12.2
```

以上输出说明，当主路由中断后，备份的静态路由被放入到路由表中，也很好地解释了浮动静态路由作为备份的工作原理。

（3）在 R1 上将 g0/0 接口启动，然后查看路由表：

```
R1(config)#interface gigabitEthernet 0/0
R1(config-if)#no shutdown
R1#show ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
```

```
C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     2.0.0.0/24 is subnetted, 1 subnets
R       2.2.2.0 [120/1] via 192.168.21.2, 00:00:09, GigabitEthernet0/0
C    192.168.21.0/24 is directly connected, GigabitEthernet0/0
```

以上输出表明当主路由恢复后，浮动静态路由又恢复到备份的地位。

4.3.4 实验 8：ip default-network

1.实验目的

通过本实验可以掌握如何通过 ip default-network 向网络中注入一条默认路由。

2.拓扑结构

实验拓扑如图 4-7 所示。

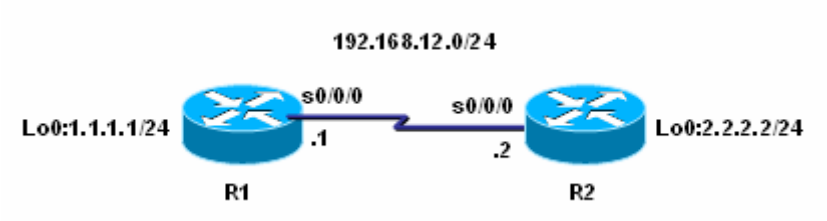

图 4-7 ip default-network 向 RIP 网络中注入默认路由

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 192.168.12.0
R1(config)#ip default-network 1.0.0.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.12.0
R2(config-router)#network 2.0.0.0
```

4.实验调试

（1）在 R2 上查看路由表：

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
```

```
Gateway of last resort is 192.168.12.1 to network 0.0.0.0
//表明默认路由的网关为 192.168.12.1
C    192.168.12.0/24 is directly connected, Serial0/0/0
     2.0.0.0/24 is subnetted, 1 subnets
C       2.2.2.0 is directly connected, Loopback0
R*   0.0.0.0/0 [120/1] via 192.168.12.1, 00:00:22, Serial0/0/0
```

从以上输出可以看出 R1 上的“**ip default-network**”命令确实向 RIP 网络中注入一条“**R***”的默认路由。

（2）在 R2 上 ping 1.1.1.1；

```
R2#ping 1.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/14/16 ms
```

以上输出表明在路由器 R2 上可以 ping 通地址 1.1.1.1，虽然在 R1 的 RIP 进程中没有通告该网络，也恰恰说明是默认路由起了作用。否则，因为在 R2 的路由表中没有 1.1.1.0 的路由条目，是不可能 ping 通的。

【技术要点】

（1）ip default-network 后面的网络一定要是主类网络；

（2）ip default-network 后面的网络可以是直连的或者通过其它协议学到的网络。

4.4 RIP 命令汇总

表 4-2 列出了本章涉及到的主要的命令。

表 4-2 本章命令汇总

命令	作用
show ip route	查看路由表
show ip protocols	查看 IP 路由协议配置和统计信息
show ip rip database	查看 RIP 数据库
debug ip rip	动态查看 RIP 的更新过程
clear ip route *	清除路由表
router rip	启动 RIP 进程
network	通告网络
version	定义 RIP 的版本
no auto-summary	关闭自动汇总
ip rip send version	配置 RIP 发送的版本
ip rip receive version	配置 RIP 接收的版本
passive-interface	配置被动接口
neighbor	配置单播更新的目标
ip summary-address rip	配置 RIP 手工汇总

key chain	定义钥匙链
key *key-id*	配置 Key ID
key-string	配置 Key ID 的密匙
ip rip triggered	配置触发更新
ip rip authentication mode	配置认证模式
ip rip authentication key-chain	配置认证使用的钥匙链
timers basic	配置更新的计时器
maximum-paths	配置等价路径的最大值
ip default-network	向网络中注入默认路由

第 5 章　EIGRP

EIGRP(Enhanced Interior Gateway Routing Protocol，增强型内部网关路由协议)是 Cisco 公司开发的一个平衡混合型路由协议，它融合了距离向量和链路状态两种路由协议的优点，支持 IP、IPX、AppleTalk 等多种网络层协议。由于 TCP / IP 是当今网络中最常用的协议，因此本书只讨论 IP 网络环境中的 EIGRP。

5.1　EIGRP 概述

EIGRP 是一个高效的路由协议，它的特点如下:

1. 通过发送和接收 Hello 包来建立和维持邻居关系，并交换路由信息；
2. 采用组播（224.0.0.10）或单播进行路由更新；
3. EIGRP 的管理距离为 90 或 170；
4. 采用触发更新，减少带宽占用；
5. 支持可变长子网掩码 (VLSM)，默认开启自动汇总功能；
6. 支持 IP、IPX、AppleTalk 等多种网络层协议；
7. 对每一种网络协议，EIGRP 都维持独立的邻居表、拓扑表和路由表；
8. EIGRP 使用 Diffusing Update 算法(DUAL)来实现快速收敛.并确保没有路由环路；
9. 存储整个网络拓扑结构的信息，以便快速适应网络变化；
10. 支持等价和非等价的负载均衡；
11. 使用可靠传输协议（RTP）保证路由信息传输的可靠性；
12. 无缝连接数据链路层协议和拓扑结构，EIGRP 不要求对 OSI 参考模型的 2 层协议做特别的配置。

5.2　实验 1：EIGRP 基本配置

1.实验目的

通过本实验可以掌握：

（1）在路由器上启动 EIGRP 路由进程

（2）启用参与路由协议的接口，并且通告网络

（3）EIGRP 度量值的计算方法

（4）可行距离（FD)、通告距离（RD）以及可行性条件（FC）

（5）邻居表、拓扑表以及路由表的含义

（6）查看和调试 EIGRP 路由协议相关信息

2.实验拓扑

本实验拓扑结构如图 5-1 所示。

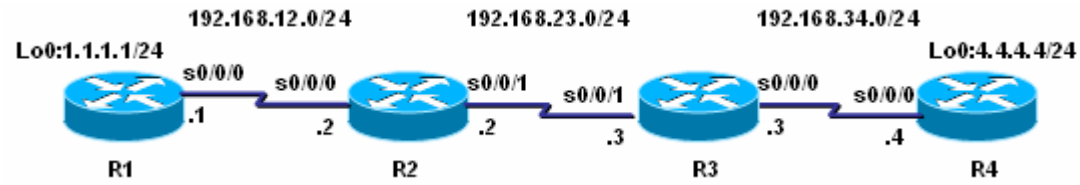

图 5-1 EIGRP 基本配置

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router eigrp 1
```

```
R1(config-router)#no auto-summary
R1(config-router)#network 1.1.1.0 0.0.0.255
R1(config-router)#network 192.168.12.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router eigrp 1
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.12.0
R2(config-router)#network 192.168.23.0
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router eigrp 1
R3(config-router)#no auto-summary
R3(config-router)#network 192.168.23.0
R3(config-router)#network 192.168.34.0
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router eigrp 1
R4(config-router)#no auto-summary
R4(config-router)#network 4.4.4.0 255.255.255.0
R4(config-router)#network 192.168.34.0
```

【说明】

EIGRP 协议在通告网段时，如果是主类网络(即标准 A、B、C 类的网络，或者说没有划分子网的网络)，只需输入此网络地址；如果是子网的话，则最好在网络号后面写子网掩码或者反掩码，这样可以避免将所有的子网都加入 EIGRP 进程中。

反掩码是用广播地址(255.255.255.255)减去子网掩码所得到。如掩码地址是 255.255.248.0，则反掩码地址是 0.0.7.255。在高级的 IOS 中也支持网络掩码的写法。

运行 EIGRP 的整个网络 AS 号码必须一致，其范围为 1-65535 之间。

4.实验调试

（1）show ip route

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
D       1.1.1.0 [90/20640000] via 192.168.12.1, 00:04:19, Serial0/0/0
     4.0.0.0/24 is subnetted, 1 subnets
D       4.4.4.0 [90/21152000] via 192.168.23.3, 00:00:06, Serial0/0/1
```

C   192.168.23.0/24 is directly connected, Serial0/0/1

D   192.168.34.0/24 [90/21024000] via 192.168.23.3, 00:05:34, Serial0/0/1

以上输出表明路由器 R2 通过 EIGRP 学到了 3 条 EIGRP 路由条目，管理距离是 90，注意 EIGRP 协议代码用字母“D”表示,如果通过重分布方式进入 EIGRP 网络的路由条目，默认管理距离为 170，路由代码用“D EX”表示，也说明 EIGRP 路由协议能够区分内部路由和外部路由。

对于 EIGRP 度量值的计算，不妨以“**D 1.1.1.0 [90/20640000] via 192.168.12.1, 00:04:19, Serial0/0/0**”路由条目为例来说明。

EIGRP 度量值的计算公式=**[K1 * Bandwidth + (K2 * Bandwidth)/(256-Load) + K3*Delay]*[K5/(Reliability + K4)] *256**

默认情况下，**K1 = K3 = 1 ，K2 = K4 = K5 = 0** 。

Bandwidth =10^7/所经由链路中入口带宽（单位为 Kbps）的最小值

Delay=所经由链路中入口的延迟之和（单位为 μs）/10

接下来看一下在路由器 R2 中的“1.1.1.0”路由条目的度量值是如何计算的？

首先看带宽应该是从 R1 的 Loopback0 到 R2 最小的，应该是 R2 的 s0/0/0 接口的带宽，为 128K,而延迟是路由器 R1 的 Loopback0 和路由器 R2 的 s0/0/0 接口的延迟之和，所以最后的度量值应该是[10^7/128+(5000+20000)/10]*256=20640000,和路由器计算的结果是一致的。

【提示】

接口的带宽和延迟可以通过“**show interface** ”查看。

（2）**show ip protocols**

```
R2#show ip protocols
Routing Protocol is "eigrp 1"
//AS 号码为 1
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
// 显示计算度量值所用的 K 值
  EIGRP maximum hopcount 100
//EIGRP 支持的最大跳数
  EIGRP maximum metric variance 1
// variance 值默认为 1，即默认时只支持等价路径的负载均衡
  Redistributing: eigrp 1
  EIGRP NSF-aware route hold timer is 240s
//不间断转发的持续时间
  Automatic network summarization is not in effect
//显示自动汇总已经关闭，默认自动汇总是开启的
  Maximum path: 4
  Routing for Networks:
    192.168.12.0
```

```
    192.168.23.0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.12.1          90      00:10:44
    192.168.23.3          90      00:10:15
  Distance: internal 90 external 170
```

（3）**show ip eigrp neighbors**

```
R2#show ip eigrp neighbors
IP-EIGRP neighbors for process 1
H   Address         Interface   Hold   Uptime   SRTT   RTO   Q    Seq
                                (sec)           (ms)       Cnt  Num
1   192.168.23.3    Se0/0/1      12    00:11:05    7    1140  0    5
0   192.168.12.1    Se0/0/0      12    00:11:29    7    1140  0    3
```

以上输出各字段的含义如下:

① H: 表示与邻居建立会话的顺序;

② Address: 邻居路由器的接口地址;

③ Interface: 本地到邻居路由器的接口;

④ Hold: 认为邻居关系不存在所能等待的最大时间;

⑤ Uptime: 从邻居关系建立到目前的时间;

⑥ SRTT: 是向邻居路由器发送一个数据包以及本路由器收到确认包的时间;

⑦ RTO: 路由器在重新传输包之前等待 ACK 的时间;

⑧ Q Cnt: 等待发送的队列;

⑨ Seq Num: 从邻居收到的发送数据包的序列号。

【技术要点】

运行 EIGRP 路由协议的路由器不能建立邻居关系的可能原因:

① EIGRP 进程的 AS 号码不同

② 计算度量值的 K 值不同

（4）**show ip eigrp topology**

```
R2#show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(192.168.23.2)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 1.1.1.0/24, 1 successors, FD is 20640000
        via 192.168.12.1 (20640000/128256), Serial0/0/0
P 4.4.4.0/24, 1 successors, FD is 21152000
        via 192.168.23.3 (21152000/20640000), Serial0/0/1
P 192.168.34.0/24, 1 successors, FD is 21024000
        via 192.168.23.3 (21024000/20512000), Serial0/0/1
P 192.168.12.0/24, 1 successors, FD is 20512000
        via Connected, Serial0/0/0
```

```
P 192.168.23.0/24, 1 successors, FD is 20512000
        via Connected, Serial0/0/1 (5)show ip eigrp interface
```

以上输出可以清楚地看到每条路由条目的 FD 和 RD 的值。而拓扑结构数据库中状态代码最常见的是“P”,“A”和“s”,含义如下：

① P: 代表 passive,表示网络处于收敛的稳定状态；

② A: 代表 active,当前网络不可用,正处于发送查询状态；

③ s：在 3 分钟内，如果被查询的路由没有收到回应，查询的路由就被置为“stuck in active”状态。

【术语】

可行距离（FD)：到达一个目的网络的最小度量值；

通告距离（RD)：邻居路由器所通告的它自己到达目的网络的最小的度量值；

可行性条件（FC)：是 EIGRP 路由器更新路由表和拓扑表的依据。可行性条件可以有效地阻止路由环路，实现路由的快速收敛。可行性条件的公式为：**AD<FD**。

（5）**show ip eigrp interfaces**

```
R2#show ip eigrp interfaces
IP-EIGRP interfaces for process 1

                   Xmit Queue   Mean   Pacing Time   Multicast    Pending
Interface   Peers  Un/Reliable  SRTT   Un/Reliable   Flow Timer   Routes
Se0/0/0       1        0/0       7       5/190        218           0
Se0/0/1       1        0/0       7       5/190        222           0
```

以上输出各字段的含义如下:

① Interface：运行 EIGRP 协议的接口；

② Peers：该接口的邻居的个数；

③ Xmit Queue Un/Reliable：在不可靠/可靠队列中存留的数据包的数量；

④ Mean SRTT：平均的往返时间，单位是秒；

⑤ Pacing Time Un/Reliable：用来确定不可靠/可靠队列中数据包被送出接口的时间间隔；

⑥ Multicast Flow Timer：组播数据包被发送前最长的等待时间；

⑦ Pending Routes：在传送队列中等待被发送的数据包携带的路由条目。

（6）**show ip eigrp traffic**

```
R2#show ip eigrp traffic
IP-EIGRP Traffic Statistics for AS 1
  Hellos sent/received: 364/361
  Updates sent/received: 10/8
  Queries sent/received: 0/0
  Replies sent/received: 0/0
  Acks sent/received: 4/5
  Input queue high water mark 1, 0 drops
  SIA-Queries sent/received: 0/0
  SIA-Replies sent/received: 0/0
  Hello Process ID: 187
```

PDM Process ID: 167

以上输出显示了 EIGRP 发送和接收到的数据包的统计情况。

（7）**debug eigrp neighbors**

该命令可以动态查看 EIGRP 邻居关系的情况。在路由器 R1 先将 s0/0/0 接口 shutdown 掉，然后再 no shutdown，可以看到 EIGRP 邻居建立的过程。

```
R2#debug eigrp neighbors
EIGRP Neighbors debugging is on
*Feb 10 02:59:31.199: %LINK-3-UPDOWN: Interface Serial0/0/0, changed state to down
*Feb 10 02:59:31.199: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 192.168.12.1 (Serial0/0/0) is down: interface down
*Feb 10 02:59:31.199: Going down: Peer 192.168.12.1 total=1 stub 0 template=1, iidb-stub=0 iid-all=0
*Feb 10 02:59:31.199: EIGRP: Neighbor 192.168.12.1 went down on Serial0/0/0
*Feb 10 02:59:32.199: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to down
*Feb 10 02:59:48.199: %LINK-3-UPDOWN: Interface Serial0/0/0, changed state to up
*Feb 10 02:59:49.199: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
*Feb 10 02:59:49.687: EIGRP: New peer 192.168.12.1 total=2 stub 0 template=1 idbstub=0 iidball=1
*Feb 10 02:59:49.687: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 192.168.12.1 (Serial0/0/0) is up: new adjacency
```

（8）**debug eigrp packets**

该命令可以显示 EIGRP 发送和接收的数据包。

```
R2#debug eigrp packets
EIGRP Packets debugging is on
    (UPDATE, REQUEST, QUERY, REPLY, HELLO, IPXSAP, PROBE, ACK, STUB, SIAQUERY, SIAREPLY)
*Feb 10 03:01:08.107: EIGRP: Received HELLO on Serial0/0/0 nbr 192.168.12.1
*Feb 10 03:01:08.107:   AS 1, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
*Feb 10 03:01:08.843: EIGRP: Received HELLO on Serial0/0/1 nbr 192.168.23.3
*Feb 10 03:01:08.843:   AS 1, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0 peerQ un/rely 0/0
*Feb 10 03:01:10.927: EIGRP: Sending HELLO on Serial0/0/0
*Feb 10 03:01:10.927:   AS 1, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
*Feb 10 03:01:12.471: EIGRP: Sending HELLO on Serial0/0/1
*Feb 10 03:01:12.471:   AS 1, Flags 0x0, Seq 0/0 idbQ 0/0 iidbQ un/rely 0/0
```

以上输出显示 R2 发送和接收的 EIGRP 数据包，由于当前网络是收敛的，所以只有 HELLO 数据包发送和接收的报告。

【术语】

在 EIGRP 中，有五种类型的数据包：

① Hello：以组播的方式定期发送，用于建立和维持邻居关系；

② 更新：当路由器收到某个邻居路由器的第一个 Hello 包时，以单播传送方式回送一

个包含它所知道的路由信息的更新包。当路由信息发生变化时，以组播的方式发送只包含变化信息的更新包；

③ 查询：当一条链路失效，路由器重新进行路由计算，但在拓扑表中没有可行的后继路由时，路由器就以组播的方式向它的邻居发送一个查询包，以询问它们是否有一条到目的地的后继路由；

④ 答复：以单播的方式回传给查询方，对查询数据包进行应答；

⑤ 确认：以单播的方式传送，用来确认更新、查询、答复数据包。

（9）在路由器 R1 上通过“ip default-network”向 EIGRP 网络注入一条默认路由，具体配置如下：

```
R1(config-if)#ip address 1.1.1.1 255.0.0.0
R1(config)#router eigrp 1
R1(config-router)#no network 1.1.1.0 0.0.0.255
R1(config-router)#network 1.0.0.0
R1(config)#ip default-network 1.0.0.0
```

在 R2 上查看路由表：

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.12.1 to network 1.0.0.0

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/8 is subnetted, 1 subnets
D*      1.0.0.0 [90/20640000] via 192.168.12.1, 00:00:08, Serial0/0/0
     4.0.0.0/24 is subnetted, 1 subnets
D       4.4.4.0 [90/21152000] via 192.168.23.3, 00:05:35, Serial0/0/1
C    192.168.23.0/24 is directly connected, Serial0/0/1
D    192.168.34.0/24 [90/21024000] via 192.168.23.3, 00:05:35, Serial0/0/1
```

以上输出表明路由器 R2 收到一条默认路由，同理，在 R3,R4 上也会收到一条默认路由。

5.3 EIGRP 负载均衡、汇总和认证

5.3.1 实验 2: EIGRP 负载均衡

1.实验目的

通过本实验可以掌握：

（1）EIGRP 等价负载均衡的实现方法

（2）EIGRP 非等价负载均衡的实现方法

（3）修改 EIGRP 度量值的方法

（4）可行距离（FD）、通告距离（RD）以及可行性条件（FC）的深层含义

2. 实验拓扑

本实验拓扑结构如图 5-2 所示。

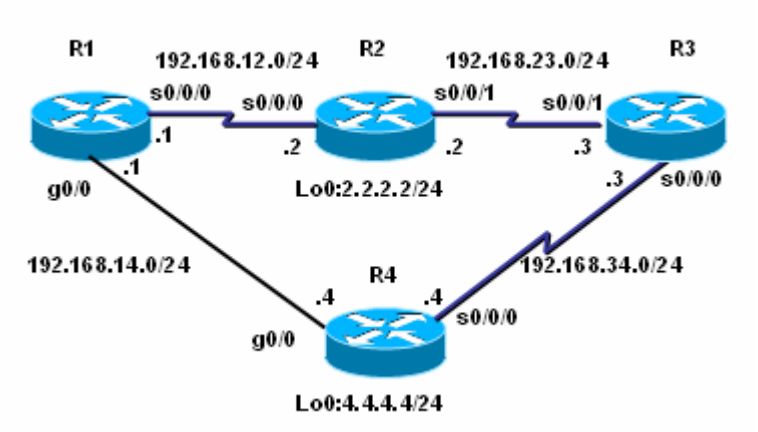

图 5-2 EIGRP 负载均衡

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router eigrp 1
R1(config-router)#no auto-summary
R1(config-router)#network 192.168.14.0
R1(config-router)#network 192.168.12.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router eigrp 1
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.12.0
R2(config-router)#network 192.168.23.0
R2(config-router)#network 2.2.2.0 255.255.255.0
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router eigrp 1
R3(config-router)#no auto-summary
R3(config-router)#network 192.168.23.0
R3(config-router)#network 192.168.34.0
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router eigrp 1
R4(config-router)#no auto-summary
R4(config-router)#network 4.4.4.0 255.255.255.0
R4(config-router)#network 192.168.34.0
R4(config-router)#network 192.168.14.0
```

4. 实验调试

（1）按照上面的配置，在 R4 查看路由表：

```
R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
```

```
        D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

D    192.168.12.0/24
           [90/20514560] via 192.168.14.1, 00:00:15, GigabitEthernet0/0
     2.0.0.0/24 is subnetted, 1 subnets
D       2.2.2.0 [90/20642560] via 192.168.14.1, 00:00:15, GigabitEthernet0/0
C    192.168.14.0/24 is directly connected, GigabitEthernet0/0
     4.0.0.0/24 is subnetted, 1 subnets
C       4.4.4.0 is directly connected, Loopback0
D    192.168.23.0/24 [90/21024000] via 192.168.34.3, 00:00:15, Serial0/0/0
C    192.168.34.0/24 is directly connected, Serial0/0/0
```

本实验只关注路由器 R2 的 Loopback 0，虽然路由器 R4 到达路由器 R2 的 Loopback 0 有两条路径，但是路由器会将 FD 最小的放入路由表，选择走 g0/0 接口。那么另外一条路径是不是可行后继路由呢？在路由器 R4 上查看拓扑表如下：

```
R4#show ip eigrp topology
IP-EIGRP Topology Table for AS(1)/ID(4.4.4.4)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 2.2.2.0/24, 1 successors, FD is 20642560
        via 192.168.14.1 (20642560/20640000), GigabitEthernet0/0
        via 192.168.34.3 (21152000/20640000), Serial0/0/0
P 4.4.4.0/24, 1 successors, FD is 128256
        via Connected, Loopback0
P 192.168.34.0/24, 1 successors, FD is 20512000
        via Connected, Serial0/0/0
P 192.168.12.0/24, 1 successors, FD is 20514560
        via 192.168.14.1 (20514560/20512000), GigabitEthernet0/0
P 192.168.14.0/24, 1 successors, FD is 28160
        via Connected, GigabitEthernet0/0
P 192.168.23.0/24, 1 successors, FD is 21024000
        via 192.168.34.3 (21024000/20512000), Serial0/0/0
```

从上面的输出中可以看到，第二条路径（走 s0/0/0 接口）的 AD 为 **20640000**，而最优路由(走 g0/0 接口)的 FD 为 **20642560**，**AD<FD**,满足可行性条件，所以第二条路径(走 s0/0/0 接口）是最优路由（走 g0/0 接口）的可行后继。

【术语】

后继：是一个直接连接的邻居路由器，通过它到达目的网络的路由最优；

可行后继：是一个邻居路由器，但是通过它到达目的地的度量值比其它路由器高，但它的通告距离小于通过后继路由器到达目的网络的可行距离，因而被保存在拓扑表中，用做备份路由。

（2）通过适当的配置，使得在路由器 R4 上看 R2 的 Loopback 0 的路由条目为等价路由，从而实现等价负载均衡。根据前面讲的 EIGRP 度量值的计算公式，这两条路径的最小带宽是相同的，只要它们的延迟之和相同，就是等价路由，为此，在路由器 R4 上做如下的配置：

```
R4(config)#interface gigabitEthernet 0/0
R4(config-if)#delay 2000
```

【提示】

在接口下用 delay 命令修改的延迟，在计算度量值时，不需要再除以 10。

在 R4 上查看路由表：

```
R4#show ip route eigrp
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

D    192.168.12.0/24
           [90/21024000] via 192.168.14.1, 00:00:15, GigabitEthernet0/0
     2.0.0.0/24 is subnetted, 1 subnets
D       2.2.2.0 [90/21152000] via 192.168.34.3, 00:00:15, Serial0/0/0
                [90/21152000] via 192.168.14.1, 00:00:15, GigabitEthernet0/0
D    192.168.23.0/24 [90/21024000] via 192.168.34.3, 00:00:15, Serial0/0/0
```

以上输出表明路由条目“2.2.2.0”确实有两条等价路径，表明 EIGRP 是支持等价负载均衡的。

（3）将 R4 的以太口 g0/0 的 delay 恢复到原来的值，通过 “variance”命令来研究 EIGRP 的非等价负载均衡。在（1）的结果中发现，对于“2.2.2.0”路由条目，在路由器 R4 的拓扑结构数据库中存在如下的记录：

```
P 2.2.2.0/24, 1 successors, FD is 20642560
        via 192.168.14.1 (20642560/20640000), GigabitEthernet0/0
        via 192.168.34.3 (21152000/20640000), Serial0/0/0
```

现在只需要在 R4 的路由器上调整 variance 的值，使得这两条路径在路由表中都可见和

可用，R4 上的配置如下：

```
R4(config)#router eigrp 1
R4(config-router)#variance 2
```

在 R4 上查看路由表：

```
R4#show ip route eigrp
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

D    192.168.12.0/24
           [90/20514560] via 192.168.14.1, 00:00:02, GigabitEthernet0/0
     2.0.0.0/24 is subnetted, 1 subnets
D       2.2.2.0 [90/21152000] via 192.168.34.3, 00:00:02, Serial0/0/0
                [90/20642560] via 192.168.14.1, 00:00:02, GigabitEthernet0/0
D    192.168.23.0/24 [90/21024000] via 192.168.34.3, 00:00:02, Serial0/0/0
```

以上输出表明路由条目“2.2.2.0”有两条路径可达，但是它们的度量值不同，这就是所说的非等价路由，从而证明 EIGRP 是支持非等价负载均衡的。

【技术要点】

EIGRP 非等价负载均衡是通过“variance”命令实现的，“variance”默认是 1(即代表等价路径的负载均衡)，variance 值的范围是 1-128.这个参数代表了可以接受的不等价路径的度量值的倍数,在这个范围内的链路都将被接受,并且被放入路由表中。

5.3.2 实验 3：EIGRP 路由汇总

1.实验目的

通过本实验可以掌握：

（1）路由汇总的目的

（2）EIGRP 自动汇总

（3）EIGRP 手工汇总

（4）指向 null0 路由的含义

2.实验拓扑

本实验拓扑结构如图 5-3 所示。

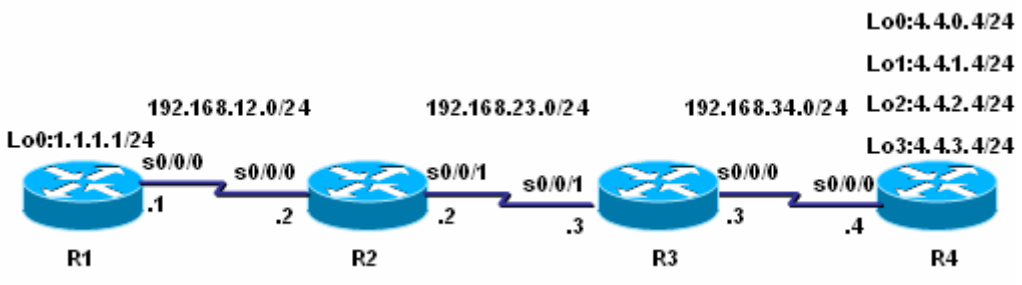

图 5-3 EIGRP 路由汇总

3. 实验步骤

本实验只给出路由器 R4 的配置，路由器 R1、R2 和 R3 的配置同 5.2 的**实验 1** 完全相同。默认的时候 EIGRP 的自动汇总是开启的，自动汇总只对本地产生的 EIGRP 路由汇总，可以通过“no auto-summary”命令关闭自动汇总，然后进行手工汇总，R4 的配置如下：

```
R4(config)#router eigrp 1
R4(config-router)#no auto-summary
R4(config-router)#network 4.4.0.0 255.255.252.0
R4(config-router)#network 192.168.34.0
R4(config)#interface s0/0/0
R4(config-if)#ip summary-address eigrp 1 4.4.0.0 255.255.252.0
//配置 EIGRP 手工路由汇总
```

4. 实验调试

（1）在 R4 s0/0/0 执行汇总之前，在 R3 上查看路由表：

```
R3#show ip route eigrp
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

D    192.168.12.0/24 [90/21024000] via 192.168.23.2, 00:23:31, Serial0/0/1
     1.0.0.0/24 is subnetted, 1 subnets
D       1.1.1.0 [90/21152000] via 192.168.23.2, 00:00:18, Serial0/0/1
     4.0.0.0/24 is subnetted, 4 subnets
D       4.4.0.0 [90/20640000] via 192.168.34.4, 00:01:02, Serial0/0/0
D       4.4.1.0 [90/20640000] via 192.168.34.4, 00:01:02, Serial0/0/0
D       4.4.2.0 [90/20640000] via 192.168.34.4, 00:01:02, Serial0/0/0
D       4.4.3.0 [90/20640000] via 192.168.34.4, 00:01:02, Serial0/0/0
```

以上输出表明路由器 R3 的路由表中有 4 条明细路由。

（2）在路由器 R4 s0/0/0 接口执行汇总，在 R3 和 R4 上查看路由表：

```
R4#show ip route eigrp
```

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

D    192.168.12.0/24 [90/21536000] via 192.168.34.3, 00:04:36, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
D       1.1.1.0 [90/21664000] via 192.168.34.3, 00:03:41, Serial0/0/0
     4.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
D       4.4.0.0/22 is a summary, 00:01:33, Null0
D    192.168.23.0/24 [90/21024000] via 192.168.34.3, 00:26:55, Serial0/0/0
R3#show ip route eigrp
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

D    192.168.12.0/24 [90/21024000] via 192.168.23.2, 00:27:30, Serial0/0/1
     1.0.0.0/24 is subnetted, 1 subnets
D       1.1.1.0 [90/21152000] via 192.168.23.2, 00:04:17, Serial0/0/1
     4.0.0.0/22 is subnetted, 1 subnets
D       4.4.0.0 [90/20640000] via 192.168.34.4, 00:02:09, Serial0/0/0
```

以上输出说明，在路由器 R4 的 s0/0/0 执行手工汇总后，会在自己路由表产生一条指向“null0”的 EIGRP 路由，主要是为了防止路由环路的；在路由器 R3 上收到被汇总的路由条目“4.4.0.0/22”。

【提示】

当被汇总的明细路由全部 down 掉以后,汇总路由才自动从路由表里被删除,从而可以有效避免路由抖动。

【思考】

如果把上面的实验的 R4 的环回接口 lo0-lo4 的地址改为 192.168.96.4/24,192.168.97.4/24, 192.168.98.4/24, 192.168.99.4/24,那么在路由器 R4 的 s0/0/0 接口还能够实现汇

总吗？在路由器 R4 上实施的配置如下：

```
R4(config)#router eigrp 1
R4(config-router)#network 192.168.96.0 255.255.252.0
R4(config)#interface s0/0/0
R4(config-if)#ip summary-address eigrp 1 192.168.96.0 255.255.252.0
```

分别在 R4 和 R3 上查看路由表：

```
R4#show ip route eigrp
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
D    192.168.12.0/24 [90/21536000] via 192.168.34.3, 00:04:36, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
D       1.1.1.0 [90/21664000] via 192.168.34.3, 00:03:41, Serial0/0/0
D    192.168.96.0/22 is a summary, 00:01:40, Null0
D    192.168.23.0/24 [90/21024000] via 192.168.34.3, 00:26:55, Serial0/0/0
R3#show ip route eigrp
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set
D    192.168.12.0/24 [90/21024000] via 192.168.23.2, 00:27:30, Serial0/0/1
     1.0.0.0/24 is subnetted, 1 subnets
D       1.1.1.0 [90/21152000] via 192.168.23.2, 00:04:17, Serial0/0/1
D    192.168.96.0/22 [90/20640000] via 192.168.34.4, 00:02:09, Serial0/0/0
```

从 R3 和 R4 的路由表的输出可以看出 EIGRP 支持 CIDR 汇总，这一点和 RIPv2 是不相同的。

5.3.3 实验 4：EIGRP 认证

1.实验目的

通过本实验可以掌握 EIGRP 路由协议认证的配置和调试。

2.实验拓扑

本实验拓扑结构如图 5-1 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#key chain ccnp
R1(config-keychain)# key 1
R1(config-keychain-key)#key-string cisco
R1(config)#interface s0/0/0
R1(config-if)#ip authentication mode eigrp 1 md5 //认证模式为 MD5
R1(config-if)#ip authentication key-chain eigrp 1 ccnp //在接口上调用钥匙链
```

（2）步骤 2：配置路由器 R2

```
R2(config)#key chain ccnp
R2(config-keychain)# key 1
R2(config-keychain-key)#key-string cisco
R2(config)#interface s0/0/0
R2(config-if)#ip authentication mode eigrp 1 md5
R2(config-if)#ip authentication key-chain eigrp 1 ccnp
R2(config)#interface s0/0/1
R2(config-if)#ip authentication mode eigrp 1 md5
R2(config-if)#ip authentication key-chain eigrp 1 ccnp
```

（3）步骤 3：配置路由器 R3

```
R3(config)#key chain ccnp
R3(config-keychain)# key 1
R3(config-keychain-key)#key-string cisco
R3(config)#interface s0/0/0
R3(config-if)#ip authentication mode eigrp 1 md5
R3(config-if)#ip authentication key-chain eigrp 1 ccnp
R3(config)#interface s0/0/1
R3(config-if)#ip authentication mode eigrp 1 md5
R3(config-if)#ip authentication key-chain eigrp 1 ccnp
```

（4）步骤 4：配置路由器 R4

```
R4(config)#key chain ccnp
R4(config-keychain)# key 1
R4(config-keychain-key)#key-string cisco
R4(config)#interface s0/0/0
R4(config-if)#ip authentication mode eigrp 1 md5
R4(config-if)#ip authentication key-chain eigrp 1 ccnp
```

4.实验调试

（1）如果链路的一端启用了认证，另外一端没有起用认证，则出现下面的提示信息：

*Feb 10 05:46:11.119: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 192.168.12.2 (Serial0/0/0) is down: **authentication mode changed**

（2）如果钥匙链的密匙不正确，则出现下面的提示信息：

*Feb 10 05:47:08.122: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 192.168.12.1 (Serial0/0/0) is down: **Auth failure**

5.4 EIGRP 命令汇总

表 5-1 列出了本章涉及到的主要的命令。

表 5-1 本章命令汇总

命令	作用
show ip eigrp neighbors	查看 EIGRP 邻居表
show ip eigrp topology	查看 EIGRP 拓扑结构数据库
show ip eigrp interface	查看运行 EIGRP 路由协议的接口的状况
show ip eigrp traffic	查看 EIGRP 发送和接收到的数据包的统计情况
debug eigrp neighbors	查看 EIGRP 动态建立邻居关系的情况
debug eigrp packets	显示发送和接收的 EIGRP 数据包
ip hello-interval eigrp	配置 EIGRP 的 HELLO 发送周期
ip hold-time eigrp	配置 EIGRP 的 HELLO hold 时间
router eigrp	启动 EIGRP 路由进程
no auto-summary	关闭自动汇总
ip authentication mode eigrp	配置 EIGRP 的认证模式
ip authentication key-chain eigrp	在接口上调用钥匙链
variance	配置非等价负载均衡
delay	配置接口下的延迟
bandwidth	配置接口下的带宽
ip summary-address eigrp	手工路由汇总

第 6 章 单区域 OSPF

OSPF（Open Shortest Path First,开放最短链路优先）路由协议是典型的链路状态路由协议。OSPF 由 IETF 在 20 世纪 80 年代末期开发,OSPF 是 SPF 类路由协议中的开放式版本。最初的 OSPF 规范体现在 RFC1131 中，被称为 OSPF 版本 1，但是版本 1 很快被进行了重大改进的版本所代替，这个新版本体现在 RFC1247 文档中。RFC1247 被称为 OSPF 版本 2,是为了明确指出其在稳定性和功能性方面的实质性改进。这个 OSPF 版本有许多更新文档，每一个更新都是对开放标准的精心改进。接下来的一些规范出现在 RFC1583 和 2328 中。OSPF 版本 2 的最新版体现在 RFC 2328 中。而 OSPF 版本 3 是关于 IPv6 的。OSPF 的内容多而复杂，所以本书分了多个章节来介绍。本章只讨论单区域的 OSPF。

6.1 OSPF 概述

OSPF 作为一种内部网关协议（Interior Gateway Protocol，IGP），用于在同一个自治系统（AS）中的路由器之间交换路由信息。OSPF 的特性如下:

1. 可适应大规模网络；
2. 收敛速度快；
3. 无路由环路；
4. 支持 VLSM 和 CIDR；
5. 支持等价路由；
6. 支持区域划分，构成结构化的网络；
7. 提供路由分级管理；
8. 支持简单口令和 MD5 认证；
9. 以组播方式传送协议报文；
10. OSPF 路由协议的管理距离是 110；
11. OSPF 路由协议采用 cost 作为度量标准；
12. OSPF 维护邻居表、拓扑表和路由表。

另外，OSPF 将网络划分为四种类型：广播多路访问型（BMA）、非广播多路访问型（NBMA）、点到点型（Point-to-Point）、点到多点型（Point-to-MultiPoint）。不同的二层链路的类型需要 OSPF 不同的网络类型来适应。

下面的几个术语是学习 OSPF 要掌握的：

1. 链路：链路就是路由器用来连接网络的接口；
2. 链路状态：用来描述路由器接口及其与邻居路由器的关系。所有链路状态信息构成链路状态数据库；
3. 区域：有相同的区域标志的一组路由器和网络的集合。在同一个区域内的路由器有相同的链路状态数据库；
4. 自治系统：采用同一种路由协议交换路由信息的路由器及其网络构成一个自治系统；
5. 链路状态通告（LSA）：LSA 用来描述路由器的本地状态，LSA 包括的信息有关于路由器接口的状态和所形成的邻接状态；
6. 最短路经优先(SPF)算法：是 OSPF 路由协议的基础。SPF 算法有时也被称为 Dijkstra 算法，这是因为最短路径优先算法(SPF)是 Dijkstra 发明的。OSPF 路由器利用 SPF，独立地计算出到达任意目的地的最佳路由。

6.2　实验 1:点到点链路上的 OSPF

1. 实验目的

通过本实验可以掌握：

（1）在路由器上启动 OSPF 路由进程

（2）启用参与路由协议的接口，并且通告网络及所在的区域

（3）度量值 cost 的计算

（4）hello 相关参数的配置

（5）点到点链路上的 OSPF 的特征

（6）查看和调试 OSPF 路由协议相关信息

2. 实验拓扑

本实验的拓扑结构如图 6-1 所示。

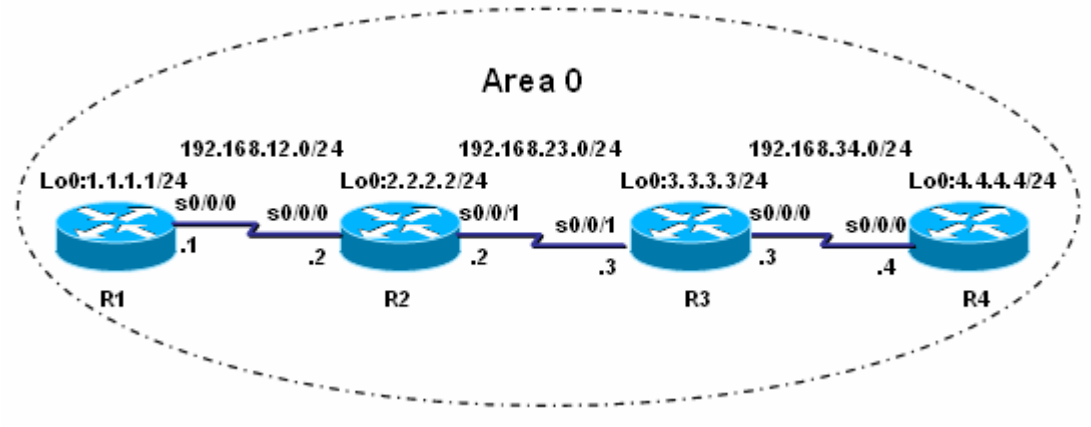

图 6-1　点到点链路上的 OSPF

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 255.255.255.0 area 0
R1(config-router)#network 192.168.12.0 255.255.255.0 area 0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 192.168.12.0 255.255.255.0 area 0
R2(config-router)#network 192.168.23.0 255.255.255.0 area 0
R2(config-router)#network 2.2.2.0 255.255.255.0 area 0
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 192.168.23.0 255.255.255.0 area 0
R3(config-router)#network 192.168.34.0 255.255.255.0 area 0
R3(config-router)#network 3.3.3.0 255.255.255.0 area 0
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 0.0.0.255 area 0
R4(config-router)#network 192.168.34.0 0.0.0.255 area 0
```

【技术要点】

（1）OSPF 路由进程 ID 的范围必须在 1-65535 之间，而且只有本地含义，不同路由器的路由进程 ID 可以不同。如果要想启动 OSPF 路由进程，至少确保有一个接口是 up 的；

（2）区域 ID 是在 0-4294967295 内的十进制数，也可以是 IP 地址的格式 A.B.C.D。当网络区域 ID 为 0 或 0.0.0.0 时称为主干区域；

（3）在高版本的 IOS 中通告 OSPF 网络的时候，网络号的后面可以跟网络掩码，也可以跟反掩码，都是可以的；

（4）确定 Router ID 遵循如下顺序：

① 最优先的是在 OSPF 进程中用命令“router-id”指定了路由器 ID；

② 如果没有在 OSPF 进程中指定路由器 ID，那么选择 IP 地址最大的环回接口的 IP 地址为 Router ID；

③ 如果没有环回接口，就选择最大的活动的物理接口的 IP 地址为 Router ID。

建议用命令“router-id”来指定路由器 ID,这样可控性比较好。

4.实验调试

（1）show ip route

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/782] via 192.168.12.1, 00:18:40, Serial0/0/0
     2.0.0.0/24 is subnetted, 1 subnets
C       2.2.2.0 is directly connected, Loopback0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/782] via 192.168.23.3, 00:18:40, Serial0/0/1
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/1563] via 192.168.23.3, 00:18:40, Serial0/0/1
C    192.168.23.0/24 is directly connected, Serial0/0/1
O    192.168.34.0/24 [110/1562] via 192.168.23.3, 00:18:41, Serial0/0/1
```

输出结果表明同一个区域内通过 OSPF 路由协议学习的路由条目用代码“O”表示。

【说明】

① 环回接口 OSPF 路由条目的掩码长度都是 32 位，这是环回接口的特性，尽管通告了 24 位，解决的办法是在环回接口下修改网络类型为“Point-to-Point”，操作如下：

R2(config)#interface loopback 0

R2(config-if)#ip ospf network point-to-point

这样收到的路由条目的掩码长度和通告的一致。

② 路由条目“4.4.4.4”的度量值为 1563，计算过程如下：

cost 的计算公式为 10^8/带宽（bps），然后取整，而且是所有链路入口的 cost 之和，环回接口的 cost 为 1，路由条目“4.4.4.4”到路由器 R2 经过的入接口包括路由器 R4 的 loopback0，路由器 R3 的 s0/0/0，路由器 R2 的 s0/0/1，所以计算如下：$1+10^8/128000+10^8/128000=1563$。也可以直接通过命令“ip ospf cost”设置接口的 cost 值，并且它是优先计算的 cost 值的。

（2）show ip protocols

```
R2#show ip protocols
Routing Protocol is "ospf 1"
 //当前路由器运行的 OSPF 进程 ID
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 2.2.2.2
 //本路由器 ID
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 //本路由器参与的区域数量和类型
  Maximum path: 4
 //支持等价路径最大数目
  Routing for Networks:
    2.2.2.0 0.0.0.255 area 0
    192.168.12.0 0.0.0.255 area 0
    192.168.23.0 0.0.0.255 area 0
 //以上四行表明 OSPF 通告的网络以及这些网络所在的区域
 Reference bandwidth unit is 100 mbps
//参考带宽为 10^8
  Routing Information Sources:
    Gateway         Distance      Last Update
    4.4.4.4              110      00:08:36
    3.3.3.3              110      00:08:36
    1.1.1.1              110      00:08:36
 //以上 5 行表明路由信息源
  Distance: (default is 110)
 //OSPF 路由协议默认的管理距离
```

（3）show ip ospf

该命令显示 OSPF 进程及区域的细节，如路由器运行 SPF 算法的次数等。

```
R2#show ip ospf 1
```

Routing Process "ospf 1" with ID 2.2.2.2
Start time: 00:50:57.156, Time elapsed: 00:42:41.880
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
Supports area transit capability
Router is not originating router-LSAs with maximum metric
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPFs 10000 msecs
Maximum wait time between two consecutive SPFs 10000 msecs
Incremental-SPF disabled
Minimum LSA interval 5 secs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Number of external LSA 0. Checksum Sum 0x000000
Number of opaque AS LSA 0. Checksum Sum 0x000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Number of areas transit capable is 0
External flood list length 0
IETF NSF helper support enabled
Cisco NSF helper support enabled
Area BACKBONE(0)
Number of interfaces in this area is **3**
Area has no authentication
SPF algorithm last executed 00:15:07.580 ago
SPF algorithm executed 9 times
Area ranges are
Number of LSA 4. Checksum Sum 0x02611A
Number of opaque link LSA 0. Checksum Sum 0x000000
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0

（4）**show ip ospf interface**

R2#**show ip ospf interface s0/0/0**
Serial0/0/0 is up, line protocol is up
Internet Address 192.168.12.2/24, Area 0
//该接口的地址和运行的 OSPF 区域
Process ID 1, Router ID 2.2.2.2, Network Type POINT_TO_POINT, Cost: 781
//进程 ID，路由器 ID，网络类型，接口 Cost 值

```
  Transmit Delay is 1 sec, State POINT_TO_POINT
//接口的延迟和状态
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
oob-resync timeout 40
//显示几个计时器的值
    Hello due in 00:00:05
//距离下次发送 Hello 包的时间
  Supports Link-local Signaling (LLS)
//支持 LLS
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
//以上两行表示启用了 IETF 和 Cisco 的 NSF 功能
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
//邻居的个数以及已建立邻接关系的邻居的个数
    Adjacent with neighbor 1.1.1.1
//已经建立邻接关系的邻居路由器 ID
  Suppress hello for 0 neighbor(s)
//没有进行 Hello 抑制
```

（5）**show ip ospf neighbor**

```
R2#show ip ospf neighbor
Neighbor ID   Pri   State           Dead Time   Address         Interface
3.3.3.3       0     FULL/  -        00:00:35    192.168.23.3    Serial0/0/1
1.1.1.1       0     FULL/  -        00:00:38    192.168.12.1    Serial0/0/0
```

以上输出表明路由器 R2 有两个邻居，它们的路由器 ID 分别为 1.1.1.1 和 3.3.3.3,其它参数解释如下：

① **Pri**：邻居路由器接口的优先级；

② **State**：当前邻居路由器接口的状态；

③ **Dead Time**：清除邻居关系前等待的最长时间；

④ **Address**：邻居接口的地址；

⑤ **Interface**：自己和邻居路由器相连接口；

⑥ “-”： 表示点到点的链路上 OSPF 不进行 DR 选举。

【技术要点】

OSPF 邻居关系不能建立的常见原因：

① hello 间隔和 dead 间隔不同；

同一链路上的 hello 包间隔和 dead 间隔必须相同才能建立邻接关系。默认情况下，hello 包发送间隔如表 6-1 所示。

表 6-1 OSPF hello 间隔和 dead 间隔

网络类型	Hello 间隔（秒）	Dead 间隔（秒）

广播多路访问	10	40
非广播多路访问	30	120
点到点	10	40
点到多点	30	120

默认时 Dead 间隔是 Hello 间隔的四倍。可以在接口下通过“ip ospf hello-interval”和“ip ospf dead-interval”命令调整。

② 区域号码不一致；

③ 特殊区域（如stub，nssa等）区域类型不匹配；

④ 认证类型或密码不一致；

⑤ 路由器 ID 相同；

⑥ Hello 包被 ACL deny；

⑦ 链路上的MTU 不匹配；

⑧ 接口下OSPF网络类型不匹配。

（6）show ip ospf database

```
R2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         240         0x80000005 0x00BA35 3
2.2.2.2         2.2.2.2         1308        0x80000008 0x00D7C0 5
3.3.3.3         3.3.3.3         1310        0x80000007 0x00282D 5
4.4.4.4         4.4.4.4         44          0x80000004 0x009AFE 3
```

以上输出是 R2 的区域 0 的拓扑结构数据库的信息，标题行的解释如下：

① Link ID：是指 Link State ID,代表整个路由器，而不是某个链路；

② ADV Router：是指通告链路状态信息的路由器 ID；

③ Age：老化时间；

④ Seq#：序列号；

⑤ Checksum：校验和；

⑥ Link count：通告路由器在本区域内的链路数目。

6.3 实验 2:广播多路访问链路上的 OSPF

1.实验目的

通过本实验可以掌握：

（1）在路由器上启动 OSPF 路由进程

（2）启用参与路由协议的接口，并且通告网络及所在的区域

（3）修改参考带宽

（4）DR 选举的控制

（5）广播多路访问链路上的 OSPF 的特征

2.实验拓扑

本实验的拓扑结构如图 6-2 所示。

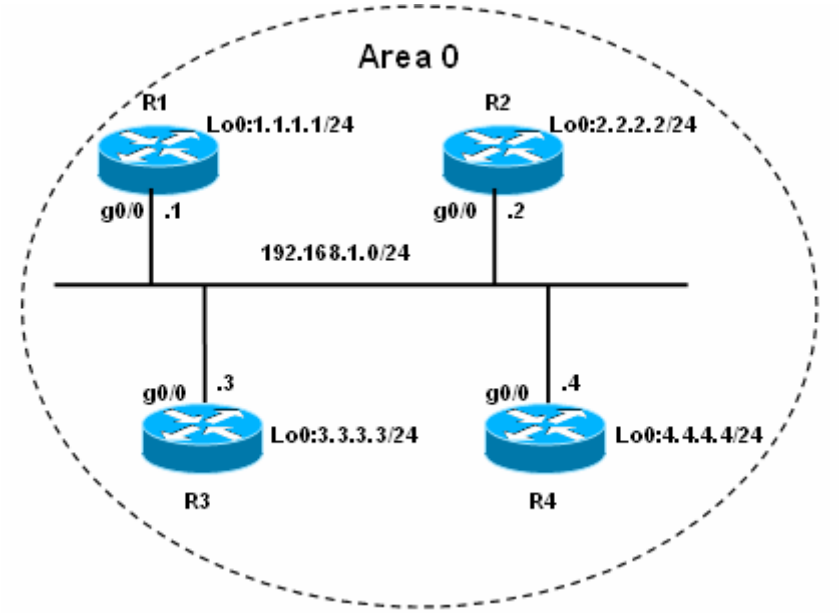

图 6-2 广播多路访问链路上的 OSPF

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 255.255.255.0 area 0
R1(config-router)#network 192.168.1.0 255.255.255.0 area 0
R1(config-router)#auto-cost reference-bandwidth 1000
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 2.2.2.0 255.255.255.0 area 0
R2(config-router)#network 192.168.1.0 255.255.255.0 area 0
R2(config-router)#auto-cost reference-bandwidth 1000
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 255.255.255.0 area 0
R3(config-router)#network 192.168.1.0 255.255.255.0 area 0
R3(config-router)#auto-cost reference-bandwidth 1000
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 255.255.255.0 area 0
R4(config-router)#network 192.168.1.0 255.255.255.0 area 0
R4(config-router)#auto-cost reference-bandwidth 1000
```

【说明】

“auto-cost reference-bandwidth”命令是修改参考带宽的，因为本实验中的以太口的带宽为千兆，如果采用默认的百兆参考带宽，计算出来的 cost 是 0.1，这显然是不合理的。修改参考带宽要在所有的 OSPF 路由器上配置，目的是确保参考标准是相同的。另外，当执行命令“auto-cost reference-bandwidth”的时候，系统也会提示如下信息：

```
% OSPF: Reference bandwidth is changed.
        Please ensure reference bandwidth is consistent across all routers.
```

4.实验调试

（1）show ip ospf neighbor

```
R1#show ip ospf neighbor

Neighbor ID  Pri   State           Dead Time   Address         Interface
2.2.2.2        1   FULL/BDR        00:00:37    192.168.1.2     GigabitEthernet0/0
3.3.3.3        1   FULL/DROTHER    00:00:37    192.168.1.3     GigabitEthernet0/0
4.4.4.4        1   FULL/DROTHER    00:00:34    192.168.1.4     GigabitEthernet0/0
```

以上输出表明在该广播多路访问网络中，R1 是 DR,R2 是 BDR,R3 和 R4 为 DROTHER。

【技术要点】

（1）为了避免路由器之间建立完全邻接关系而引起的大量开销，OSPF 要求在多路访问的网络中选举一个 DR，每个路由器都与之建立邻接关系。选举 DR 的同时也选举出一个 BDR，在 DR 失效的时候，BDR 担负起 DR 的职责，而且所有其它路由器只与 DR 和 BDR 建立邻接关系；

（2）DR 和 BDR 有它们自己的组播地址 224.0.0.6；

（3）DR 和 BDR 的选举是以各个网络为基础的，也就是说 DR 和 BDR 选举是一个路由器的接口特性，而不是整个路由器的特性；

（4）DR 选举的原则：

① 首要因素是时间，最先启动的路由器被选举成 DR；

② 如果同时启动，或者重新选举，则看接口优先级（范围为 0-255），优先级最高的被选举成 DR，默认情况下，多路访问网络的接口优先级为 1，点到点网络接口优先级为 0，修改接口优先级的命令是“ **ip ospf priority**”，如果接口的优先级被设置为 0，那么该接口将不参与 DR 选举；

③ 如果前两者相同，最后看路由器 ID，路由器 ID 最高的被选举成 DR；

（5）DR 选举是非抢占的，除非人为地重新选举。重新选举 DR 的方法有两种，一是路由器重新启动，二是执行“clear ip ospf process”命令。

（2）show ip ospf interface

分别在路由器 R1 和 R4 上执行该命令：

```
R1#show ip ospf interface g0/0
GigabitEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type BROADCAST, Cost: 10
```

```
  Transmit Delay is 1 sec, State DR, Priority 1
//自己 state 是 DR
  Designated Router (ID) 1.1.1.1, Interface address 192.168.1.1
//DR 的路由器 ID 以及接口地址
  Backup Designated router (ID) 2.2.2.2, Interface address 192.168.1.2
//BDR 的路由器 ID 以及接口地址
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:09
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 3, Adjacent neighbor count is 3
//R1 是 DR,有 3 个邻居，并且全部形成邻接关系
    Adjacent with neighbor 2.2.2.2  (Backup Designated Router)   //R2 是 BDR
    Adjacent with neighbor 3.3.3.3
    Adjacent with neighbor 4.4.4.4
  Suppress hello for 0 neighbor(s)
R4#show ip ospf interface g0/0
GigabitEthernet0/0 is up, line protocol is up
  Internet Address 192.168.1.4/24, Area 0
  Process ID 1, Router ID 4.4.4.4, Network Type BROADCAST, Cost: 10
//网络类型为 BROADCAST
  Transmit Delay is 1 sec, State DROTHER, Priority 1
//自己的 state 是 DROTHER
  Designated Router (ID) 1.1.1.1, Interface address 192.168.1.1
//DR 的路由器 ID 和接口地址
  Backup Designated router (ID) 2.2.2.2, Interface address 192.168.1.2
//BDR 的路由器 ID 和接口地址
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:06
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 3, Adjacent neighbor count is 2
```

//有 3 个邻居，只与 R1 和 R2 形成邻接关系，与 R3 只是邻居关系

Adjacent with neighbor 1.1.1.1 (Designated Router)

Adjacent with neighbor 2.2.2.2 (Backup Designated Router)

//上面两行表示与 DR 和 BDR 形成邻接关系

Suppress hello for 0 neighbor(s)

从上面的路由器 R1 和 R4 的输出得知，邻居关系和邻接关系是不能混为一谈的，邻居关系是指达到 2WAY 状态的两台路由器，而邻接关系是指达到 FULL 状态的两台路由器。

（3）**debug ip ospf adj**

该命令显示 OSPF 邻接关系创建或中断的过程。

R2#**debug ip ospf adj**

OSPF adjacency events debugging is on

R2#**clear ip ospf process**

Reset ALL OSPF processes? [no]: y

*Feb 10 10:37:33.447: OSPF: Interface GigabitEthernet0/0 going Down

*Feb 10 10:37:33.447: OSPF: 2.2.2.2 address 192.168.1.2 on GigabitEthernet0/0 is dead, state **DOWN**

*Feb 10 10:37:33.447: OSPF: Neighbor change Event on interface GigabitEthernet0/0

*Feb 10 10:37:33.447: OSPF: DR/BDR election on GigabitEthernet0/0

*Feb 10 10:37:33.447: OSPF: Elect BDR 4.4.4.4

*Feb 10 10:37:33.447: OSPF: Elect DR 1.1.1.1

*Feb 10 10:37:33.447: OSPF: Elect BDR 4.4.4.4

*Feb 10 10:37:33.447: OSPF: Elect DR 1.1.1.1

*Feb 10 10:37:33.447: **DR: 1.1.1.1 (Id) BDR: 4.4.4.4 (Id)**

*Feb 10 10:37:33.447: OSPF: Reset adjacency with 3.3.3.3 on GigabitEthernet0/0, state 2WAY

*Feb 10 10:37:33.447: OSPF: 1.1.1.1 address 192.168.1.1 on GigabitEthernet0/0 is dead, state DOWN

*Feb 10 10:37:33.447: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down: Interface down or detached

*Feb 10 10:37:33.447: OSPF: Neighbor change Event on interface GigabitEthernet0/0

*Feb 10 10:37:33.447: OSPF: DR/BDR election on GigabitEthernet0/0

*Feb 10 10:37:33.447: OSPF: Elect BDR 4.4.4.4

*Feb 10 10:37:33.447: OSPF: Elect DR 4.4.4.4

*Feb 10 10:37:33.447: DR: 4.4.4.4 (Id) BDR: 4.4.4.4 (Id)

*Feb 10 10:37:33.447: OSPF: Remember old DR 1.1.1.1 (id)

*Feb 10 10:37:33.447: OSPF: 3.3.3.3 address 192.168.1.3 on GigabitEthernet0/0 is dead, state DOWN

*Feb 10 10:37:33.447: %OSPF-5-ADJCHG: Process 1, Nbr 3.3.3.3 on GigabitEthernet0/0 from 2WAY to DOWN, Neighbor Down: Interface down or detached

*Feb 10 10:37:33.447: OSPF: Neighbor change Event on interface GigabitEthernet0/0

*Feb 10 10:37:33.447: OSPF: DR/BDR election on GigabitEthernet0/0

*Feb 10 10:37:33.447: OSPF: Elect BDR 4.4.4.4

*Feb 10 10:37:33.447: OSPF: Elect DR 4.4.4.4

*Feb 10 10:37:33.447: **DR: 4.4.4.4 (Id) BDR: 4.4.4.4 (Id)**

*Feb 10 10:37:33.447: OSPF: 4.4.4.4 address 192.168.1.4 on GigabitEthernet0/0 is dead, state

DOWN

*Feb 10 10:37:33.447: %OSPF-5-ADJCHG: Process 1, Nbr 4.4.4.4 on GigabitEthernet0/0 from FULL to DOWN, Neighbor Down: Interface down or detached

*Feb 10 10:37:33.447: OSPF: Neighbor change Event on interface GigabitEthernet0/0

*Feb 10 10:37:33.447: OSPF: DR/BDR election on GigabitEthernet0/0

*Feb 10 10:37:33.447: OSPF: Elect BDR 0.0.0.0

*Feb 10 10:37:33.447: OSPF: Elect DR 0.0.0.0

*Feb 10 10:37:33.447: DR: none BDR: none

*Feb 10 10:37:33.447: OSPF: Remember old DR 4.4.4.4 (id)

*Feb 10 10:37:33.447: OSPF: Interface Loopback0 going Down

*Feb 10 10:37:33.447: OSPF: 2.2.2.2 address 2.2.2.2 on Loopback0 is dead, state DOWN

*Feb 10 10:37:33.459: OSPF: Interface GigabitEthernet0/0 going Up

*Feb 10 10:37:33.459: OSPF: Interface Loopback0 going Up

*Feb 10 10:37:33.459: OSPF: **2 Way** Communication to 1.1.1.1 on GigabitEthernet0/0, state 2WAY

*Feb 10 10:37:33.459: OSPF: 2 Way Communication to 3.3.3.3 on GigabitEthernet0/0, state 2WAY

*Feb 10 10:37:33.459: OSPF: 2 Way Communication to 4.4.4.4 on GigabitEthernet0/0, state 2WAY

*Feb 10 10:37:33.459: OSPF: Backup seen Event before WAIT timer on GigabitEthernet0/0

*Feb 10 10:37:33.459: OSPF: DR/BDR election on GigabitEthernet0/0

*Feb 10 10:37:33.459: OSPF: Elect BDR 4.4.4.4

*Feb 10 10:37:33.459: OSPF: Elect DR 1.1.1.1

*Feb 10 10:37:33.459: **DR: 1.1.1.1 (Id) BDR: 4.4.4.4 (Id)**

*Feb 10 10:37:33.459: OSPF: Send DBD to 1.1.1.1 on GigabitEthernet0/0 seq 0xC87 opt 0x52 flag 0x7 len 32

*Feb 10 10:37:33.459: OSPF: Send DBD to 4.4.4.4 on GigabitEthernet0/0 seq 0x1B1C opt 0x52 flag 0x7 len 32

*Feb 10 10:37:33.463: OSPF: Rcv DBD from 1.1.1.1 on GigabitEthernet0/0 seq 0x1A0 opt 0x52 flag 0x7 len 32 mtu 1500 state **EXSTART**

*Feb 10 10:37:33.463: OSPF: First DBD and we are not SLAVE

*Feb 10 10:37:33.463: OSPF: Rcv DBD from 1.1.1.1 on GigabitEthernet0/0 seq 0xC87 opt 0x52 flag 0x2 len 112 mtu 1500 state EXSTART

*Feb 10 10:37:33.463: OSPF: NBR Negotiation Done. We are the MASTER

*Feb 10 10:37:33.463: OSPF: Send DBD to 1.1.1.1 on GigabitEthernet0/0 seq 0xC88 opt 0x52 flag 0x1 len 32

*Feb 10 10:37:33.463: OSPF: Rcv DBD from 4.4.4.4 on GigabitEthernet0/0 seq 0x13C0 opt 0x52 flag 0x7 len 32 mtu 1500 state EXSTART

*Feb 10 10:37:33.463: OSPF: NBR Negotiation Done. We are the SLAVE

*Feb 10 10:37:33.463: OSPF: Send DBD to 4.4.4.4 on GigabitEthernet0/0 seq 0x13C0 opt 0x52 flag 0x0 len 32

*Feb 10 10:37:33.463: OSPF: Rcv DBD from 1.1.1.1 on GigabitEthernet0/0 seq 0xC88 opt 0x52 flag 0x0 len 32 mtu 1500 state **EXCHANGE**

*Feb 10 10:37:33.463: OSPF: Exchange Done with 1.1.1.1 on GigabitEthernet0/0

*Feb 10 10:37:33.463: OSPF: Send LS REQ to 1.1.1.1 length 48 LSA count 4

*Feb 10 10:37:33.463: OSPF: Rcv DBD from 4.4.4.4 on GigabitEthernet0/0 seq 0x13C1 opt 0x52 flag 0x3 len 112 mtu 1500 state EXCHANGE

*Feb 10 10:37:33.463: OSPF: Send DBD to 4.4.4.4 on GigabitEthernet0/0 seq 0x13C1 opt 0x52 flag 0x0 len 32

*Feb 10 10:37:33.463: OSPF: Rcv LS UPD from 1.1.1.1 on GigabitEthernet0/0 length 212 LSA count 4

*Feb 10 10:37:33.467: OSPF: Synchronized with 1.1.1.1 on GigabitEthernet0/0, state FULL

*Feb 10 10:37:33.467: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on GigabitEthernet0/0 from **LOADING** to **FULL**, Loading Done

*Feb 10 10:37:33.467: OSPF: Rcv DBD from 4.4.4.4 on GigabitEthernet0/0 seq 0x13C2 opt 0x52 flag 0x1 len 32 mtu 1500 state EXCHANGE

*Feb 10 10:37:33.467: OSPF: Exchange Done with 4.4.4.4 on GigabitEthernet0/0

*Feb 10 10:37:33.467: OSPF: Synchronized with 4.4.4.4 on GigabitEthernet0/0, state **FULL**

*Feb 10 10:37:33.467: %OSPF-5-ADJCHG: Process 1, Nbr 4.4.4.4 on GigabitEthernet0/0 from LOADING to FULL, Loading Done

*Feb 10 10:37:33.467: OSPF: Send DBD to 4.4.4.4 on GigabitEthernet0/0 seq 0x13C2 opt 0x52 flag 0x0 len 32

*Feb 10 10:37:33.947: OSPF: Build router LSA for area 0, router ID 2.2.2.2, seq 0x80000001, process 1

*Feb 10 10:37:38.155: OSPF: Rcv LS UPD from 4.4.4.4 on GigabitEthernet0/0 length 76 LSA count 1

*Feb 10 10:37:38.443: OSPF: Rcv LS UPD from 1.1.1.1 on GigabitEthernet0/0 length 76 LSA count 1

*Feb 10 10:37:38.595: OSPF: Rcv LS UPD from 4.4.4.4 on GigabitEthernet0/0 length 76 LSA count 1

*Feb 10 10:37:38.635: OSPF: Rcv LS UPD from 1.1.1.1 on GigabitEthernet0/0 length 76 LSA count 1

*Feb 10 10:37:43.155: OSPF: Build router LSA for area 0, router ID 2.2.2.2, seq 0x80000005, process 1

*Feb 10 10:37:43.155: OSPF: Rcv LS UPD from 1.1.1.1 on GigabitEthernet0/0 length 76 LSA count 1

以上的输出表明：

① DR 重新选举的过程和结果，新的 DR 是 R1，BDR 是 R4；

② 在 OSPF 邻接关系建立的过程中，接口的状态的变化包括 DOWN、2 Way、EXSTART、EXCHANGE、Loading 和 FULL。

6.4 OSPF 认证

6.4.1 实验 3：基于区域的 OSPF 简单口令认证

1.实验目的

通过本实验可以掌握：

（1）OSPF 认证的类型和意义

（2）基于区域的 OSPF 简单口令认证的配置和调试

2.实验拓扑

本实验的拓扑结构如图 6-3 所示。

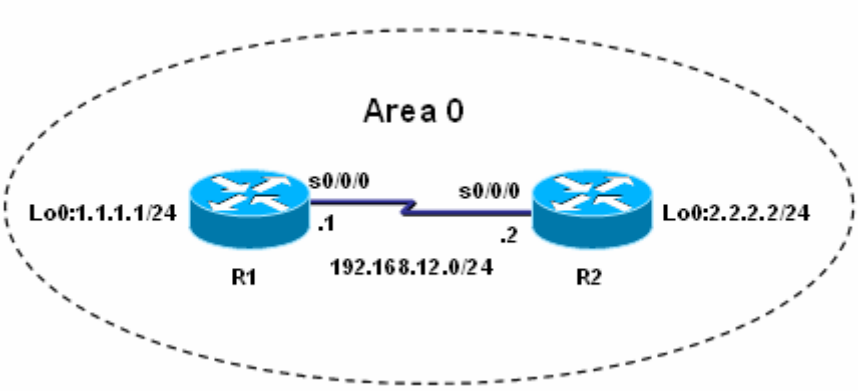

图 6-3 基于区域的 OSPF 简单口令认证

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 192.168.12.0 255.255.255.0 area 0
R1(config-router)#network 1.1.1.0 255.255.255.0 area 0
R1(config-router)#area 0 authentication //区域 0 启用简单口令认证
R1(config)#interface s0/0/0
R1(config-if)#ip ospf authentication-key cisco //配置认证密码
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 2.2.2.0 255.255.255.0 area 0
R2(config-router)#network 192.168.12.0 255.255.255.0 area 0
R2(config-router)#area 0 authentication
R2(config)#interface s0/0/0
R2(config-if)#ip ospf authentication-key cisco
```

4. 实验调试

（1）show ip ospf interface

```
R1#show ip ospf interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 192.168.12.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type POINT_TO_POINT, Cost: 781
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:02
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
```

```
Next 0x0(0)/0x0(0)
Last flood scan length is 0, maximum is 1
Last flood scan time is 0 msec, maximum is 0 msec
Neighbor Count is 0, Adjacent neighbor count is 0
Suppress hello for 0 neighbor(s)
Simple password authentication enabled
```

以上输出最后一行信息表明该接口启用了简单口令认证。

（2）show ip ospf

```
R1#show ip ospf
 Routing Process "ospf 1" with ID 1.1.1.1
 Supports only single TOS(TOS0) routes
 ......
    Area BACKBONE(0)
        Number of interfaces in this area is 2 (1 loopback)
        Area has simple password authentication
        SPF algorithm last executed 00:00:01.916 ago
        SPF algorithm executed 5 times
        Area ranges are
        Number of LSA 2. Checksum Sum 0x010117
        Number of opaque link LSA 0. Checksum Sum 0x000000
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

以上输出表明区域 0 采用简单口令认证。

（3）如果 R1 区域 0 没有启动认证，而 R2 区域 0 启动简单口令认证，则 R2 上出现下面的信息：

*Feb 10 11:03:03.071: OSPF: Rcv pkt from 192.168.12.1, Serial0/0/0 : Mismatch Authentication type. Input packet specified type 0, we use type 1

（4）如果 R1 和 R2 的区域 0 都启动简单口令认证，但是 R2 的接口下没有配置密码或密码错误，则 R2 上出现下面的信息：

*Feb 10 10:55:53.071: OSPF: Rcv pkt from 192.168.12.1, Serial0/0/0 : Mismatch Authentication Key - Clear Text

6.4.2 实验 4：基于区域的 OSPF MD5 认证

1.实验目的

通过本实验可以掌握：

（1）OSPF 认证的类型和意义

（2）基于区域的 OSPF MD5 认证的配置和调试

2.实验拓扑

本实验的拓扑结构如图 6-3 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 192.168.12.0 255.255.255.0 area 0
R1(config-router)#network 1.1.1.0 255.255.255.0 area 0
R1(config-router)#area 0 authentication message-digest //区域0启用MD5认证
R1(config)#interface s0/0/0
R1(config-if)#ip ospf message-digest-key 1 md5 cisco//配置认证key ID及密匙
```

（2）步骤2：配置路由器R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 2.2.2.0 255.255.255.0 area 0
R2(config-router)#network 192.168.12.0 255.255.255.0 area 0
R2(config-router)#area 0 authentication message-digest
R2(config)#interface s0/0/0
R2(config-if)#ip ospf message-digest-key 1 md5 cisco
```

4.实验调试

（1）show ip ospf interface

```
R1#show ip ospf interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 192.168.12.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type POINT_TO_POINT, Cost: 781
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:09
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
  Message digest authentication enabled
    Youngest key id is 1
```

输出最后两行信息表明该接口启用了MD5认证，而且密钥ID为1。

（2）show ip ospf

```
R1#show ip ospf
 Routing Process "ospf 1" with ID 1.1.1.1
 Supports only single TOS(TOS0) routes
......
    Area BACKBONE(0)
        Number of interfaces in this area is 2 (1 loopback)
```

```
Area has message digest authentication
SPF algorithm last executed 00:01:50.096 ago
SPF algorithm executed 5 times
Area ranges are
Number of LSA 2. Checksum Sum 0x010117
Number of opaque link LSA 0. Checksum Sum 0x000000
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0
```

以上输出表明区域 0 采用 MD5 认证。

（3）如果 R1 区域 0 启动 MD5 认证，而 R2 区域 0 启动简单口令认证，则 R2 上出现下面的信息：

*Feb 10 11:08:13.075: OSPF: Rcv pkt from 192.168.12.1, Serial0/0/0 : Mismatch Authentication type. Input packet specified type 2, we use type 1

（4）如果 R1 和 R2 的区域 0 都启动 MD5 认证，但是 R2 的接口下没有配置 key ID 和密码或密码错误，则 R2 上出现下面的信息：

*Feb 10 11:08:43.075: OSPF: Rcv pkt from 192.168.12.1, Serial0/0/0 : **Mismatch Authentication Key - No message digest key 1** on interface

6.4.3 实验 5：基于链路的 OSPF 简单口令认证

1.实验目的

通过本实验可以掌握：

（1）OSPF 认证的类型和意义

（2）基于链路的 OSPF 简单口令认证的配置和调试

2.实验拓扑

本实验的拓扑结构如图 6-3 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
R1(config-router)#network 192.168.12.0 0.0.0.255 area 0
R1(config)#interface s0/0/0
R1(config-if)#ip ospf authentication //链路启用简单口令认证
R1(config-if)#ip ospf authentication-key cisco //配置认证密码
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 2.2.2.0 0.0.0.255 area 0
R2(config-router)#network 192.168.12.0 0.0.0.255 area 0
R2(config)#interface s0/0/0
R2(config-if)#ip ospf authentication
R2(config-if)#ip ospf authentication-key cisco
```

4.实验调试

（1）show ip ospf interface

```
R1#show ip ospf interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 192.168.12.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type POINT_TO_POINT, Cost: 781
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:09
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 2.2.2.2
  Suppress hello for 0 neighbor(s)
  Simple password authentication enabled
```

以上输出最后一行信息表明该接口启用了简单口令认证。

（2）如果 R1 的 s0/0/0 接口启动简单口令认证，R2 的 s0/0/0 接口没有启动认证，则 R2 上出现下面的信息：

*Feb 10 11:19:33.074: OSPF: Rcv pkt from 192.168.12.1, Serial0/0/0 : Mismatch Authentication type. Input packet specified type 1, we use type 0

（3）如果 R1 和 R2 的 s0/0/0 都启动简单口令认证，但是 R2 的接口下没有配置认证密码或密码错误，则 R2 上出现下面的信息：

*Feb 10 11:22:33.074: OSPF: Rcv pkt from 192.168.12.1, Serial0/0/0 : Mismatch Authentication Key - Clear Text

6.4.4 实验 6：基于链路的 OSPF MD5 认证

1.实验目的

通过本实验可以掌握：

（1）OSPF 认证的类型和意义

（2）基于链路的 OSPF MD5 认证的配置和调试

2.实验拓扑

本实验的拓扑结构如图 6-3 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
R1(config-router)#network 192.168.12.0 0.0.0.255 area 0
```

R1(config)#interface s0/0/0
R1(config-if)#ip ospf authentication message-digest//接口 s0/0/0 启用 MD5 认证
R1(config-if)#ip ospf message-digest-key 1 md5 cisco //配置 key ID 及密匙
（2）步骤 2：配置路由器 R2
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 2.2.2.0 0.0.0.255 area 0
R2(config-router)#network 192.168.12.0 0.0.0.255 area 0
R2(config)#interface s0/0/0
R2(config-if)#ip ospf authentication message-digest
R2(config-if)#ip ospf message-digest-key 1 md5 cisco

4.实验调试

（1）show ip ospf interface

```
R1#show ip ospf interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 192.168.12.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type POINT_TO_POINT, Cost: 781
  Transmit Delay is 1 sec, State POINT_TO_POINT
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:00
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/1, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 2.2.2.2
  Suppress hello for 0 neighbor(s)
  Message digest authentication enabled
    Youngest key id is 1
```

输出最后两行信息表明该接口启用了 MD5 认证，而且密钥 ID 为 1。

（2）如果 R1 的 s0/0/0 启动 MD5 认证，而 R2s0/0/0 启动简单口令认证，则 R2 上出现下面的信息：

*Feb 10 11:08:13.075: OSPF: Rcv pkt from 192.168.12.1, Serial0/0/0 : Mismatch Authentication type. Input packet specified type 2, we use type 1

（3）如果 R1 和 R2 的 s0/0/0 都启动 MD5 认证，但是 R2 的接口下没有配置 key ID 和密码，则 R2 上出现下面的信息：

*Feb 10 11:31:13.078: OSPF: Rcv pkt from 192.168.12.1, Serial0/0/0 : Mismatch Authentication Key - No message digest key 1 on interface

【技术要点】

1. OSPF 链路认证优于区域认证；

2. OSPF 定义 3 种认证类型：0-表示不进行认证，是缺省的类型；1-表示采用简单口令认证；2-表示采用 MD5 认证。

6.5 实验 7: default-information originate

1. 实验目的

通过本实验可以掌握如何通过命令“default-information originate”向 OSPF 网络注入一条默认路由。

2. 实验拓扑

本实验的拓扑结构如图 6-4 所示。

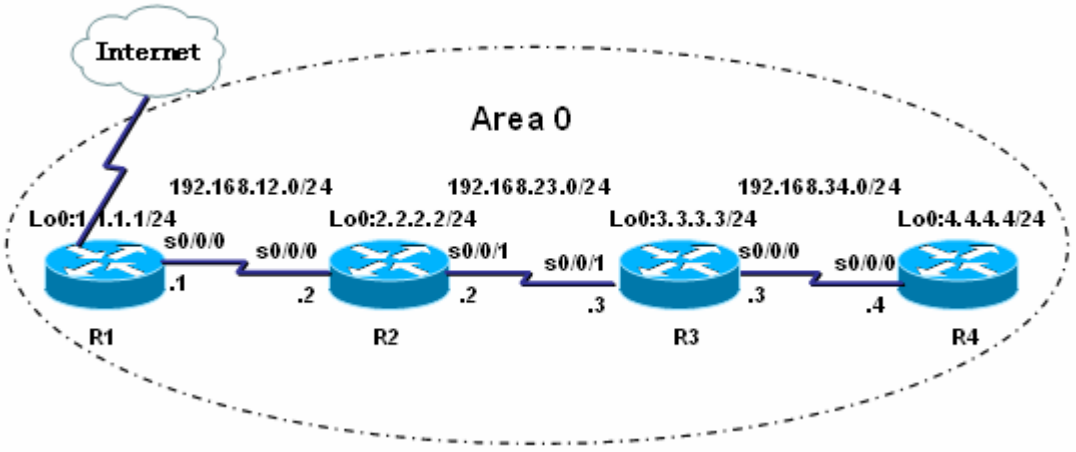

图 6-4 “default-information originate”向 OSPF 网络注入一条默认路由

3. 实验步骤

本实验用 R1 的环回接口 1 来模拟 Internet。

（1）步骤 1：配置路由器 R1

```
R1(config)#interface loopback 1
R1(config-if)#ip address 5.5.5.5 255.255.255.0
R1(config)#ip route 0.0.0.0 0.0.0.0 loopback 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
R1(config-router)#network 192.168.12.0 0.0.0.255 area 0
R1(config-router)#default-information originate
```

【技术要点】

“default-information originate”命令后面可以加可选的“always”参数，如果不使用该参数，路由器上必须存在一条默认路由，否则该命令不产生任何效果。如果使用该参数，无论路由器上是否存在默认路由，路由器都会向 OSPF 区域内注入一条默认路由。

（2）步骤 2：路由器 R2、R3 和 R4 的配置同 6.2 实验 1 的配置完全相同。

4. 实验调试

（1）show ip route

```
R4#show ip route ospf
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.34.3 to network 0.0.0.0

O    192.168.12.0/24 [110/2343] via 192.168.34.3, 00:01:26, Serial0/0/0
     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/2344] via 192.168.34.3, 00:01:26, Serial0/0/0
     2.0.0.0/32 is subnetted, 1 subnets
O       2.2.2.2 [110/1563] via 192.168.34.3, 00:01:26, Serial0/0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/782] via 192.168.34.3, 00:01:26, Serial0/0/0
O    192.168.23.0/24 [110/1562] via 192.168.34.3, 00:01:27, Serial0/0/0
O*E2 0.0.0.0/0 [110/1] via 192.168.34.3, 00:01:27, Serial0/0/0
R3#show ip route ospf
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.168.23.2 to network 0.0.0.0

O    192.168.12.0/24 [110/1562] via 192.168.23.2, 00:05:28, Serial0/0/1
     1.0.0.0/32 is subnetted, 1 subnets
O       1.1.1.1 [110/1563] via 192.168.23.2, 00:05:28, Serial0/0/1
     2.0.0.0/32 is subnetted, 1 subnets
O       2.2.2.2 [110/782] via 192.168.23.2, 00:05:28, Serial0/0/1
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/782] via 192.168.34.4, 00:05:28, Serial0/0/0
O*E2 0.0.0.0/0 [110/1] via 192.168.23.2, 00:05:30, Serial0/0/1
```

从上面R3和R4的路由表的输出，可以看到通过命令“default-information originate”确实可以向OSPF区域注入一条默认路由。

（2）show ip ospf database

```
R4#show ip ospf database
```

```
            OSPF Router with ID (4.4.4.4) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         746         0x80000010 0x000BD7 3
2.2.2.2         2.2.2.2         188         0x80000016 0x00CFB8 5
3.3.3.3         3.3.3.3         163         0x80000007 0x00282D 5
4.4.4.4         4.4.4.4         248         0x80000007 0x009402 3
                Type-5 AS External Link States

Link ID         ADV Router      Age         Seq#       Checksum Tag
0.0.0.0         1.1.1.1         863         0x80000001 0x001D91 1
```

通过查看 R4 的拓扑结构数据库可以看到，确实从外面注入了一条类型 5 的 LSA，相关的内容在 18 章介绍。

6.6 OSPF 命令汇总

表 6-2 列出了本章涉及到的主要的命令。

表 6-2 本章命令汇总

命令	作用
show ip route	查看路由表
show ip ospf neighbor	查看 OSPF 邻居的基本信息
show ip ospf database	查看 OSPF 拓扑结构数据库
show ip ospf interface	查看 OSPF 路由器接口的信息
show ip ospf	查看 OSPF 进程及其细节
debug ip ospf adj	显示 OSPF 邻接关系创建或中断的过程
debug ip ospf events	显示 OSPF 发生的事件
debug ip ospf packet	显示路由器收到的所有的 OSPF 数据包
router ospf	启动 OSPF 路由进程
router-id	配置路由器 ID
network	通告网络及网络所在的区域
ip ospf network	配置接口网络类型
ip ospf cost	配置接口 cost 值
ip ospf hello-interval	配置 hello 间隔
ip ospf dead-interval	配置 OSPF 邻居的死亡时间
ip ospf priority	配置接口优先级
auto-cost reference-bandwidth	配置参考带宽
clear ip ospf process	清除 OSPF 进程
area *area-id* authentication	启动区域简单口令认证
ip ospf authentication-key cisco	配置认证密码
area *area-id* authentication message-digest	启动区域 MD5 认证

ip ospf message-digest-key *key-id* md5 *key*	配置 key ID 及密匙
ip ospf authentication	启用链路简单口令认证
ip ospf authentication message-digest	启用链路 MD5 认证
default-information originate	向 OSPF 区域注入默认路由

第 7 章 HDLC 和 PPP

路由器经常用于构建广域网，广域网链路的封装和以太网上的封装有着非常大的差别。常见的广域网封装有 HDLC、PPP、Frame-relay 等，本章介绍 HDLC 和 PPP。相对而言，PPP 比起 HDLC 有较多的功能。

7.1 HDLC 和 PPP 简介

7.1.1 HDLC 介绍

HDLC 是点到点串行线路上（同步电路）的帧封装格式，其帧格式和以太网帧格式有很大的差别,HDLC 帧没有源 MAC 地址和目的 MAC 地址。Cisco 公司对 HDLC 进行了专有化,Cisco 的 HDLC 封装和标准的 HDLC 不兼容。如果链路的两端都是 Cisco 设备，使用 HDLC 封装没有问题，但如果 Cisco 设备与非 Cisco 设备进行连接，应使用 PPP 协议。HDLC 不能提供验证，缺少了对链路的安全保护。默认时，Cisco 路由器的串口是采用 Cisco HDLC 封装的。如果串口的封装不是 HDLC，要把封装改为 HDLC 使用命令“encapsulation hdlc”。

7.1.2 PPP 介绍

1. PPP 概述

和 HDLC 一样，PPP 也是串行线路上（同步电路或者异步电路）的一种帧封装格式，但是 PPP 可以提供对多种网络层协议的支持。PPP 支持认证、多链路捆绑、回拨、压缩等功能。PPP 经过 4 个过程在一个点到点的链路上建立通信连接：

- 链路的建立和配置协调：通信的发起方发送 LCP 帧来配置和检测数据链路
- 链路质量检测：在链路已经建立、协调之后进行，这一阶段是可选的
- 网络层协议配置协调：通信的发起方发送 NCP 帧以选择并配置网络层协议
- 关闭链路：通信链路将一直保持到 LCP 或 NCP 帧关闭链路或发生一些外部事件

2. PPP 认证：PAP 和 CHAP

（1）PAP——密码验证协议

PAP（Password Authentication Protocol）利用 2 次握手的简单方法进行认证。在 PPP 链路建立完毕后，源节点不停地在链路上反复发送用户名和密码，直到验证通过。PAP 的验证中，密码在链路上是以明文传输的，而且由于是源节点控制验证重试频率和次数，因此 PAP 不能防范再生攻击和重复的尝试攻击。

（2）CHAP——询问握手验证协议

CHAP(Challenge Handshake Authentication Protocol)利用 3 次握手周期地验证源端节点的身份。CHAP 验证过程在链路建立之后进行，而且在以后的任何时候都可以再次进行。这使得链路更为安全；CHAP 不允许连接发起方在没有收到询问消息的情况下进行验证尝试。CHAP 每次使用不同的询问消息，每个消息都是不可预测的唯一的值，CHAP 不直接传送密码，只传送一个不可预测的询问消息，以及该询问消息与密码经过 MD5 加密运算后的加密值。所以 CHAP 可以防止再生攻击，CHAP 的安全性比 PAP 要高。

7.2 实验 1:HDLC 和 PPP 封装

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 串行链路上的封装概念

（2） HDLC 封装

（3） PPP 封装

2. 实验拓扑

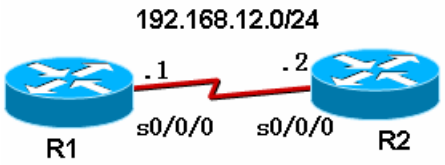

图 7-1 实验 1--实验 3 拓扑图

3. 实验步骤

（1） 步骤 1：在 R1 和 R2 路由器上配置 IP 地址、保证直连链路的连通性

```
R1(config)#int s0/0/0
R1(config-if)#ip address 192.168.12.1 255.255.255.0
R1(config-if)#no shutdown

R2(config)#int s0/0/0
R2(config-if)#clock rate 128000
R2(config-if)#ip address 192.168.12.2 255.255.255.0
R2(config-if)#no shutdown

R1#show interfaces s0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 192.168.12.1/24
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set          //该接口的默认封装为 HDLC 封装
（此处省略）
```

（2） 步骤 2：改变串行链路两端的接口封装为 PPP 封装

```
R1(config)#int s0/0/0
R1(config-if)#encapsulation ppp

R2(config)#int s0/0/0
R2(config-if)#encapsulation ppp

R1#show int s0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 192.168.12.1/24
```

MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation PPP, LCP Open　//该接口的封装为 PPP 封装
Open: IPCP, CDPCP, loopback not set　//网络层支持 IP 和 CDP 协议
（此处省略）

4. 实验调试

（1） 测试 R1 和 R2 之间串行链路的连通性

```
R1#ping 192.168.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/13/16 ms
```

如果链路的两端封装相同，则 ping 测试应该正常

（2） 链路两端封装不同协议

```
R1(config)#int s0/0/0
R1(config-if)#encapsulation ppp

R2(config)#int s0/0/0
R2(config-if)#encapsulation hdlc

R1#show int s0/0/0
Serial0/0/0 is up, line protocol is down
```

（此处省略）
//两端封装不匹配，导致链路故障

【提示】显示串行接口时，常见以下几种状态：

```
Serial0/0/0 is up, line protocol is up
```

//链路正常

```
Serial0/0/0 is administratively down, line protocol is down
```

//没有打开该接口，执行“no shutdown”可以打开接口

```
Serial0/0/0 is up, line protocol is down
```

//物理层正常，数据链路层有问题，通常是没有配置时钟、两端封装不匹配、PPP 认证错误

```
Serial0/0/0 is down, line protocol is down
```

//物理层故障，通常是连线问题

7.3 实验 2:PAP 认证

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） PAP 认证的配置方法

2. 实验拓扑

如图 7-1。

3. 实验步骤

在实验 1 基础上继续本实验。首先配置路由器 R1（远程路由器，被认证方）在路由器

R2（中心路由器，认证方）取得验证：

(1) 两端路由器上的串口采用 PPP 封装，用“encapsulation”命令：

R1(config)#int s0/0/0

R1(config-if)#encapsulation ppp

(2) 在远程路由器 R1 上，配置在中心路由器上登录的用户名和密码，使用“ppp pap sent-username 用户名 password 密码”命令：

R1(config-if)#ppp pap sent-username R1 password 123456

(3) 在中心路由器上的串口采用 PPP 封装，用“encapsulation”命令：

R2(config)#int s0/0/0

R2(config-if)#encapsulation ppp

(4) 在中心路由器上，配置 PAP 验证，使用“ppp authentication pap”命令：

R2(config-if)#ppp authentication pap

(5) 中心路由器上为远程路由器设置用户名和密码，使用“username 用户名 password 密码”命令：

R2(config)#username R1 password 123456

以上的步骤只是配置了 R1（远程路由器）在 R2（中心路由器）取得验证，即单向验证。然而在实际应用中通常是采用双向验证，即：R2 要验证 R1，而 R1 也要验证 R2。我们要采用类似的步骤进行配置 R1 对 R2 进行验证，这时 R1 为中心路由器，R2 为远程路由器了：

(6) 在中心路由器 R1 上，配置 PAP 验证，使用“ppp authentication pap”命令：

R1(config-if)#ppp authentication pap

(7) 在中心路由器 R1 上为远程路由器 R2 设置用户名和密码，使用“username 用户名 password 密码”命令：

R1(config)#username R2 password 654321

(8) 在远程路由器 R2 上，配置以什么用户和密码在远程路由器上登录，使用“ppp pap sent-username 用户名 password 密码”命令：

R2(config-if)#ppp pap sent-username R2 password 654321

【提示】在 ISDN 拨号上网时，却通常只是电信对用户进行验证（要根据用户名来收费），用户不能对电信进行验证，即验证是单向的。

4. 实验调试

使用“debug ppp authentication”命令可以查看 ppp 认证过程。

```
R1#debug ppp authentication
PPP authentication debugging is on
//以上打开 ppp 认证调试
R1(config)#int s0/0/0
R1(config-if)#shutdown
R1(config-if)#no shutdown
//由于 PAP 认证是在链路建立后进行一次，把接口关闭重新打开以便观察认证过程
*Feb 22 12:18:20.355: %LINK-3-UPDOWN: Interface Serial0/0/0, changed state to up
*Feb 22 12:18:20.355: Se0/0/0 PPP: Using default call direction
*Feb 22 12:18:20.355: Se0/0/0 PPP: Treating connection as a dedicated line
*Feb 22 12:18:20.355: Se0/0/0 PPP: Session handle[C0000006] Session id[15]
*Feb 22 12:18:20.355: Se0/0/0 PPP: Authorization required
*Feb 22 12:18:20.359: Se0/0/0 PAP: Using hostname from interface PAP
```

*Feb 22 12:18:20.359: Se0/0/0 PAP: Using password from interface PAP
*Feb 22 12:18:20.359: Se0/0/0 PAP: O AUTH-REQ id 13 len 14 from "R1"
*Feb 22 12:18:20.363: Se0/0/0 PAP: I AUTH-REQ id 2 len 14 from "R2"
*Feb 22 12:18:20.363: Se0/0/0 PAP: Authenticating peer R2
*Feb 22 12:18:20.363: Se0/0/0 PPP: Sent PAP LOGIN Request
*Feb 22 12:18:20.363: Se0/0/0 PPP: Received LOGIN Response PASS
*Feb 22 12:18:20.363: Se0/0/0 PPP: Sent LCP AUTHOR Request
*Feb 22 12:18:20.363: Se0/0/0 PPP: Sent IPCP AUTHOR Request
*Feb 22 12:18:20.363: Se0/0/0 LCP: Received AAA AUTHOR Response PASS
*Feb 22 12:18:20.363: Se0/0/0 IPCP: Received AAA AUTHOR Response PASS
*Feb 22 12:18:20.363: Se0/0/0 PAP: O AUTH-ACK id 2 len 5
*Feb 22 12:18:20.363: Se0/0/0 PAP: I AUTH-ACK id 13 len 5
*Feb 22 12:18:20.363: Se0/0/0 PPP: Sent CDPCP AUTHOR Request
*Feb 22 12:18:20.363: Se0/0/0 CDPCP: Received AAA AUTHOR Response PASS
*Feb 22 12:18:20.367: Se0/0/0 PPP: Sent IPCP AUTHOR Request
*Feb 22 12:18:21.363: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
//以上是认证成功的例子

*Feb 22 12:22:07.391: Se0/0/0 PPP: Authorization required
*Feb 22 12:22:09.411: Se0/0/0 PAP: Using hostname from interface PAP
*Feb 22 12:22:09.411: Se0/0/0 PAP: Using password from interface PAP
*Feb 22 12:22:09.411: Se0/0/0 PAP: O AUTH-REQ id 15 len 14 from "R1"
*Feb 22 12:22:09.411: Se0/0/0 PAP: I AUTH-REQ id 4 len 14 from "R2"
*Feb 22 12:22:09.411: Se0/0/0 PAP: Authenticating peer R2
*Feb 22 12:22:09.411: Se0/0/0 PPP: Sent PAP LOGIN Request
*Feb 22 12:22:09.415: Se0/0/0 PPP: Received LOGIN Response FAIL
*Feb 22 12:22:09.415: Se0/0/0 PAP: O AUTH-NAK id 4 len 26 msg is **"Authentication failed"**
//以上是认证失败的例子，例如密码错误等

7.4 实验 3:CHAP 认证

1. 实验目的

通过本实验，读者可以掌握如下技能：

(1) CHAP 认证的配置方法

2. 实验拓扑

如图 7-1。

3. 实验步骤

在实验 1 基础上继续本实验。

(1) 使用“**username 用户名 password 密码**” 命令为对方配置用户名和密码，需要注意的是两方的密码要相同：

R1(config)**#username R2 password hello**

R2(config)**#username R1 password hello**

(2) 路由器的两端串口采用 PPP 封装，并采用配置 CHAP 验证：

```
R1(config)#int s0/0/0
R1(config-if)#encapsulation ppp
R1(config-if)#ppp authentication chap

R2(config)#int s0/0/0
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication chap
```

上面是 CHAP 验证的最简单配置，也是实际应用中最常用的配置方式。配置时要求用户名为对方路由器名，而双方密码必须一致。原因是：由于 CHAP 默认使用本地路由器的名字做为建立 PPP 连接时的识别符。路由器在收到对方发送过来的询问消息后，将本地路由器的名字作为身份标识发送给对方；而在收到对方发过来的身份标识之后，默认使用本地验证方法，即在配置文件中寻找，看看有没有用户身份标识和密码；如果有，计算加密值，结果正确则验证通过；否则验证失败，连接无法建立。

【提示】在配置验证时也可以选择同时使用 PAP 和 CHAP，如：

```
R2(config-if)#ppp authentication chap pap 或
R2(config-if)#ppp authentication pap chap
```

如果同时使用两种验证方式，那么在链路协商阶段将先用第一种验证方式进行验证。如果对方建议使用第二种验证方式或者只是简单拒绝使用第一种方式，那么将采用第二种方式。

7.5 本章小结

本章介绍了串行链路上常用的两种封装方法:HDLC 和 PPP,前者只在 Cisco 设备间使用，后者可以用于不同厂商的设备间。PPP 和 HDLC 相比有较多的功能：支持多网络层协议、支持认证、支持多链路捆绑、支持回拨和压缩等。PPP 的认证有两种方式：PAP 和 CHAP，CHAP 比起 PAP 具有较好的安全性能。表 7-1 是本章出现的命令。

表 7-1 本章命令汇总

命令	作用
encapsulation hdlc	把接口的封装改为 hdlc
encapsulation ppp	把接口的封装改为 ppp
ppp pap sent-username R1 password 123456	pap 认证时，向对方发送用户名 R1 和密码 123456
ppp authentication pap	PPP 的认证方式为 pap
username R1 password 123456	为对方创建用户 R1，密码为 123456
debug ppp authentication	打开 ppp 的认证调试过程
ppp authentication chap	PPP 的认证方式为 chap

第 8 章 帧中继

帧中继线路是中小企业常用的广域网线路，其通信费用较低。由于帧中继技术的一些特殊性使得帧中继的配置较为复杂，特别是在帧中继上运行路由协议时更是如此。作为入门，对帧中继的理解应着重放在 DLCI、PVC、帧中继映射和子接口等概念上。本章通过几个实验详细介绍了帧中继的关键概念。

8.1 帧中继简介

8.1.1 什么是帧中继

帧中继（Frame Relay，FR）是面向连接的第二层传输协议，帧中继是典型的包交换技术。相比而言，同样带宽的帧中继通信费用比 DDN 专线要低，而且允许用户在帧中继交换网络比较空闲的时候以高于 ISP 所承诺的速率进行传输。

8.1.2 帧中继的合理性

用户经常需要租用线路把分散在各地的网络连接起来，如图 8-1（1），如果采用点到点的专用线路（例如 DDN），ISP 需要给每个地方的路由器拉 4 对物理线路，同时每个路由器需要有 4 个串口。而帧中继网络拓扑如 8-1（2）所示，每个路由器只通过一条线路连接到帧中继云上，线路的代价大大减低，每个路由器也只需要一个串行接口了。

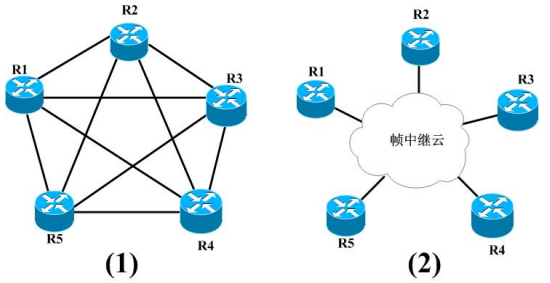

图 8-1　（1）用专线连接用户设备　（2）帧中继网络拓扑

8.1.3 DLCI

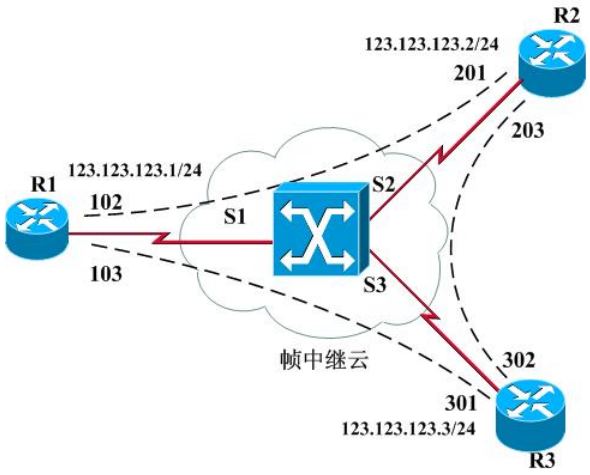

图 8-2 帧中继网络

DLCI（Data Link Circiut Identification，数据链路连接标识符）实际上就是帧中继网络中的第 2 层地址。如图 8-2，当路由器 R1 要把数据发向路由器 R2（IP 为 123.123.123.2）时，路由器 R1 可以用 DLCI=102 来对 IP 数据包进行第 2 层的封装。数据帧到了帧中继交换机，帧中继交换机根据帧中继交换表进行交换：从 S1 接口收到一个 DLCI 为 102 的帧时，交换机将把帧从 S2 接口发送出去，并且发送出去的帧的 DLCI 改为 201。这样路由器 R2 就会接收到 R1 发来的数据包。而当路由器 R2 要发送数据给 R1（IP 为 123.123.123.1）时，路由器 R2 可以用 DLCI=201 来对 IP 数据包进行第 2 层的封装，数据帧到了帧中继交换机，帧中继交换机同样根据帧中继交换表进行交换：从 S2 接口收到一个 DLCI 为 201 的帧时，交换机将把帧从 S1 接口发送出去，并且发送出去的帧的 DLCI 改为 102。这样路由器 R1 就会接收到 R2 发来的数据包。

通过以上分析可以知道DLCI实际上就是IP数据包在帧中继链路上进行封装时所需的第 2 层地址。图 8-2 各路由器中的第 3 层地址和第 2 层地址映射如下：

R1： 123.123.123.2→102
　　123.123.123.3→103
R2： 123.123.123.1→201
　　123.123.123.3→203
R3： 123.123.123.1→301
　　123.123.123.2→302

帧中继的一个非常重要特性是 NBMA（非广播多录访问）。在图 8-2 中，如果路由器在 DLCI 为 102 的 PVC 上发送一个广播，R2 路由器可以收到，然而 R3 是无法收到的。如果 R1 想发送的广播让 R2 和 R3 都收到，必须分别在 DLCI 为 102 和 103 的 PVC 上各发送一次，这就是非广播的含义。多路访问的意思是帧中继网络是多个设备接在同一网络介质上，以太网也是多路访问网络。

8.1.4 帧中继术语

(1) 永久虚电路（PVC）：虚电路是永久建立的链路，由 ISP 在其帧中继交换机静态配置交换表实现。不管电路两端的设备是否连接上，它总是为它保留相应的带宽。
(2) 数据链路连接标识符（DLCI）：一个在路由器和帧中继交换机之间标识 PVC 或者 SVC 的数值。
(3) 本地管理接口（LMI）：是路由器和帧中继交换机之间的一种信令标准，负责管理设备之间的连接及维护其连接状态。
(4) 承诺信息速率（CIR，Committed Information Rate）：也叫保证速率，是服务提供商承诺将要提供的有保证的速率，一般为一段时间内（承诺速率测量间隔 T）的平均值，其单位为 bps。
(5) 超量突发（EB，Excess Brust）：在承诺信息速率之外，帧中继交换机试图发送而未被准许的最大额外数据量，单位为 bit。超量突发依赖于服务提供商提供的服务状况，但它通常受到本地接入环路端口速率的限制。

8.1.5 LMI

LMI（Local Management Interface）提供了一个帧中继交换机和路由器之间的简单信令。在帧中继交换机和路由器之间必须采用相同的 LMI 类型，Cisco 路由器在较高版本(11.2 以后)的 IOS 中具有自动检测 LMI 类型的功能。配置接口 LMI 类型的命令为“**encapsulation frame-relay [*cisco* | *ietf*]**” 。路由器从帧中继交换机收到 LMI 信息后，可以得知 PVC 状态。三种 PVC 状态是：

- 激活状态（Active）：本地路由器与帧中继交换机的连接是启动且激活的。可以与帧中继交换机交换数据。
- 非激活状态（Inactive）：本地路由器与帧中继交换机的连接是启动且激活的，但 PVC 另一端的路由器未能与它的帧中继交换机通信。
- 删除状态（Deleted）：本地路由器没有从帧中继交换机上收到任何 LMI，可能线路或网络有问题，或者配置了不存在的 PVC。

8.1.6 帧中继映射

DLCI 是帧中继网络中的第 2 层地址。路由器要通过帧中继网络把 IP 数据包发到下一跳路由器时，它必须知道 IP 和 DLCI 的映射才能进行帧的封装。有两种方法可以获得该映射：一种是静态映射，由管理员手工输入；另一种是动态映射。默认时，路由器帧中继接口是开启动态映射的。

1．静态映射

管理员手工输入的映射就为静态映射，其命令为：

frame-relay map ip ***protocol address dlci*** **[broadcast]**

其中：

protocol：协议类型

address：网络地址

dlci：为所需要交换逆向 ARP 信息的本地接口的 DLCI 号

broadcast：参数表示允许在帧中继线路上传送路由广播或组播信息

例子：R1(config-if)**#frame map ip 123.123.123.2 102 broadcast**

2．动态映射

IARP（Inverse ARP，逆向 ARP）允许路由器自动建立帧中继映射，其工作原理如图 8-3 所示：

(1) R1 路由器从 DLCI=102 的 PVC 上发送 IARP 包,IARP 包中有 R1 的 IP 地址 12.12.12.1 ；

(2) 帧中继云对数据包进行交换，最终把 IARP 包通过 DLCI=201 的 PVC 发送给 R2；

(3) 由于 R2 是从 201 的 PVC 上接收到该 IARP 包，R2 就自动建立一个映射：

12.12.12.1——→201

(4) 同样 R2 也发送 IARP 数据包，R1 收到该 IARP 包，也会自动建立一个映射：

12.12.12.2——→102

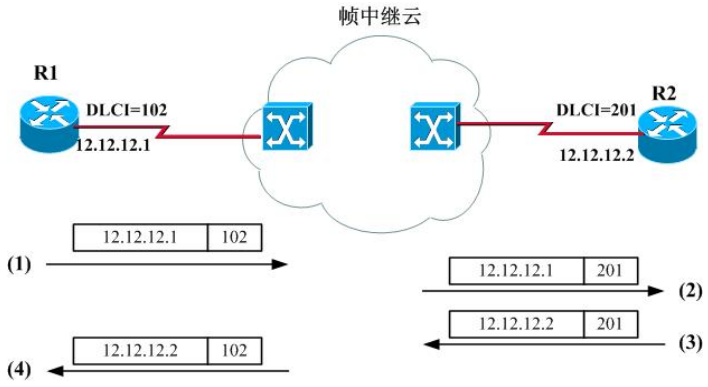

图 8-3 动态映射(IARP)工作示意图

8.1.7 子接口

子接口实际上是一个逻辑的接口，并不存在真正物理上的子接口。子接口有两种类型：

点到点、点到多点。采用点到点子接口时，每一个子接口用来连接一条 PVC，每条 PVC 的另一端连接到另一路由器的一个子接口或物理接口。这种子接口的连接与通过物理接口连接的点对点连接效果是一样的。每一对点对点的连接都是在不同的子网。

一个点到多点子接口被用来建立多条 PVC，这些 PVC 连接到远端路由器的多个子接口或物理接口。这时，所有加入连接的接口（不管是物理接口还是子接口）都应该在同一个子网上。点到多点子接口和一个没有配置子接口的物理主接口相同，路由更新要受到水平分割的限制。默认时多点子接口水平分割是开启的。

8.2 实验 1:把一台 Cisco 路由器配置为帧中继交换机

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解帧中继交换表的工作原理

（2） 理解 PVC 的概念

（3） 用路由器充当帧中继交换机的配置

2. 实验拓扑

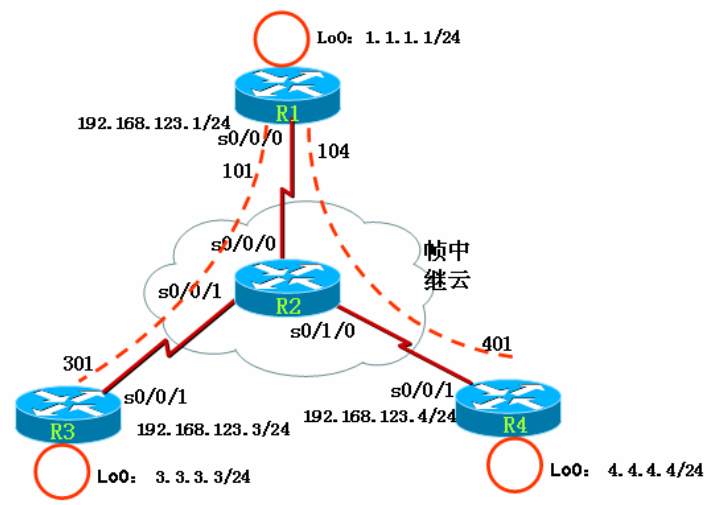

图 8-4 实验 1--实验 4 拓扑图

3. 实验步骤

我们这里只关心 R2 的配置。

(1) 步骤 1：开启帧中继交换功能

```
R2(config)#frame-relay switching  //注：把该路由器当成帧中继交换机
```

(2) 步骤 2：配置接口封装

```
R2(config)#int s0/0/0
R2(config-if)#no shutdown
R2(config-if)#clock rate 128000    //注：该接口为 DCE，要配置时钟
R2(config-if)#encapsulation frame-relay
```

//“encapsulation frame-relay [ietf]”命令用来配置接口封装成帧中继，如果不加 ietf 参数，帧类型为 cisco；如果加 ietf 参数，则帧类型为 ietf。

```
R2(config)#int s0/0/1
R2(config-if)#no shutdown
R2(config-if)#clock rate 128000
R2(config-if)#encapsulation frame-relay
R2(config)#int s0/1/0
R2(config-if)#no shutdown
R2(config-if)#clock rate 128000
R2(config-if)#encapsulation frame-relay
```

(3) 步骤 3：配置 LMI 类型

```
R2(config)#int s0/0/0
R2(config-if)#frame-relay lmi-type cisco
```

//命令“frame-relay lmi-type { ansi | cisco | q933a }用来配置 LMI 的类型，默认时是 cisco 。

```
R2(config-if)#frame-relay intf-type dce
```

//命令“frame-relay intf-type { dce | dte }”用来配置接口是帧中继的 DCE 还是 DTE，要注意的是：这里的帧中继接口 DCE 和 s0/0/0 接口是 DCE 还是 DTE 无关，也就是说即使 s0/0/0 是 DTE，也可以把它配置成帧中继的 DCE。

```
R2(config)#int s0/0/1
R2(config-if)#frame-relay lmi-type cisco
R2(config-if)#frame-relay intf-type dce
R2(config)#int s0/1/0
R2(config-if)#frame-relay lmi-type cisco
R2(config-if)#frame-relay intf-type dce
```

(4) 步骤 4：配置帧中继交换表

```
R2(config)#int s0/0/0
R2(config-if)#frame-relay route 103 interface s0/0/1 301
R2(config-if)#frame-relay route 104 interface s0/1/0 401
```

//命令“frame-relay route 103 interface s0/0/1 301”是配置帧中继交换表的，告诉路由器如果从该接口收到 DLCI=103 的帧，要从 s0/0/1 交换出去，并且 DLCI 改为 301。

```
R2(config)#int Serial0/0/1
R2(config-if)#frame-relay route 301 interface Serial0/0/0 103
R2(config)#int Serial0/1/0
R2(config-if)#frame-relay route 401 interface Serial0/0/0 104
```

4. 实验调试

可以使用“show frame-relay route”、“show frame pvc”、“show frame lmi”等命令检查帧中继交换机是否正常。

（1） “show frame-relay route”

```
R2#show frame-relay route
Input Intf      Input Dlci      Output Intf     Output Dlci     Status
```

```
Serial0/0/0      103             Serial0/0/1      301             inactive
Serial0/0/0      104             Serial0/1/0      401             inactive
Serial0/0/1      301             Serial0/0/0      103             inactive
Serial0/1/0      401             Serial0/0/0      104             inactive
```

（2） “show frame pvc”

```
R2#show frame-relay pvc
PVC Statistics for interface Serial0/0/0 (Frame Relay DCE)
              Active      Inactive      Deleted       Static
  Local          0             0             0            0
  Switched       0             2             0            0
  Unused         0             0             0            0
DLCI = 103, DLCI USAGE = SWITCHED, PVC STATUS = INACTIVE, INTERFACE = Serial0/0/0
//由于 PVC 还未被使用，所以状态为 inactive
  input pkts 0             output pkts 0            in bytes 0
  out bytes 0              dropped pkts 0           in pkts dropped 0
  out pkts dropped 0                 out bytes dropped 0
  in FECN pkts 0           in BECN pkts 0           out FECN pkts 0
  out BECN pkts 0          in DE pkts 0             out DE pkts 0
  out bcast pkts 0         out bcast bytes 0
  30 second input rate 0 bits/sec, 0 packets/sec
  30 second output rate 0 bits/sec, 0 packets/sec
  switched pkts 0
  Detailed packet drop counters:
  no out intf 0            out intf down 0          no out PVC 0
  in PVC down 0            out PVC down 0           pkt too big 0
  shaping Q full 0         pkt above DE 0           policing drop 0
  pvc create time 00:06:05, last time pvc status changed 00:06:05
```

（此处省略）

8.3 实验 2:帧中继基本配置、帧中继映射

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 帧中继的基本配置

（2） 帧中继的动态映射

（3） 帧中继的静态映射

2. 实验拓扑

如图 8-4。

3. 实验步骤

在实验 1 的基础上进行实验 2。图 8-4 中，我们已经模拟出了帧中继交换机，现配置 R1、R3、R4，使得它们能够互相通信，配置步骤如下：

(1)　帧中继接口基本配置

```
R1(config)#int s0/0/0
R1(config-if)#ip address 192.168.123.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#encapsulation frame-relay
```

//使用命令“encapsulation frame-relay [*ietf*]”。帧中继有两种封装类型：cisco 和 ietf（Internet Engineering Task Force)。对于 cisco 路由器，cisco 是它的默认值；对于非 cisco 路由器，须选用 ietf 类型。但国内帧中继线路一般为 ietf 类型的封装，我们这里由于上面的帧中继交换机中封装类型是 cisco，所以这里选择 cisco。

```
R1(config-if)#frame-relay lmi-type cisco
```

//如果采用的是 cisco 路由器且 IOS 是 11.2 及以后版本的，路由器可以自动适应 LMI 的类型，则本步骤可不做。国内帧中继线路一般采用 ansi 的 LMI 信令类型，我们这里采用的是 cisco。

```
R3(config)#int s0/0/1
R3(config-if)#ip address 192.168.123.3 255.255.255.0
R3(config-if)#no shutdown
R3(config-if)#encapsulation frame-relay

R4(config)#int s0/0/1
R4(config-if)#ip address 192.168.123.4 255.255.255.0
R4(config-if)#no shutdown
R4(config-if)#encapsulation frame-relay
```

(2)　测试连通性

从各个路由器 ping 其他路由器：

```
R1#ping 192.168.123.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/28 ms
R1#ping 192.168.123.4
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.123.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/28 ms

R1#show frame-relay map
Serial0/0/0 (up): ip 192.168.123.3 dlci 103(0x67,0x1870), dynamic,
              broadcast,, status defined, active
Serial0/0/0 (up): ip 192.168.123.4 dlci 104(0x68,0x1880), dynamic,
              broadcast,, status defined, active
```

//默认时，帧中继接口开启了动态映射，会自动建立帧中继映射，“dynamic”表明这是动态映射。

R1#show frame-relay pvc

PVC Statistics for interface Serial0/0/0 (Frame Relay DTE)

	Active	Inactive	Deleted	Static
Local	2	0	0	0
Switched	0	0	0	0
Unused	0	0	0	0

DLCI = 103, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/0/0

//可以看到DLCI=103的PVC的状态为active

input pkts 11 output pkts 11 in bytes 1074

out bytes 1074 dropped pkts 0 in pkts dropped 0

out pkts dropped 0 out bytes dropped 0

in FECN pkts 0 in BECN pkts 0 out FECN pkts 0

out BECN pkts 0 in DE pkts 0 out DE pkts 0

out bcast pkts 1 out bcast bytes 34

5 minute input rate 0 bits/sec, 0 packets/sec

5 minute output rate 0 bits/sec, 0 packets/sec

pvc create time 00:07:31, last time pvc status changed 00:06:01

（此处省略）

(3) 手工配置帧中继映射

默认情况下，路由器支持逆向ARP。若逆向ARP未打开，可以用下列命令设置：

R1(config-if)#frame-relay inverse-arp

我们也可以关闭IARP，使用静态映射，命令如下：

“frame-relay map ip *address dlci* [broadcast]”

这里的broadcast参数是允许该帧中继链路通过多播或广播包,如果帧中继链路上要运行路由协议，该参数非常重要。

R1(config)#int s0/0/0

R1(config-if)#no frame-relay inverse-arp //关闭自动映射

R1(config-if)#frame-relay map ip 192.168.123.3 103 broadcast

R1(config-if)#frame-relay map ip 192.168.123.4 104 broadcast

R3(config)#int s0/0/1

R3(config-if)#no frame-relay inverse-arp

R3(config-if)#frame-relay map ip 192.168.123.1 301 broadcast

R4(config)#int s0/0/1

R4(config-if)#no frame-relay inverse-arp

R4(config-if)#frame-relay map ip 192.168.123.1 401 broadcast

4. 实验调试

可以使用“show frame-relay map”、“show frame pvc”、“show frame lmi”等命令检查帧中继交换机是否正常。

R1#show frame-relay map

Serial0/0/0 (up): ip 192.168.123.3 dlci 103(0x67,0x1870), dynamic,

broadcast,, status defined, active

从命令输出中可以得到的信息有：

- 192.168.123.3 映射到 103
- Dynamic：表明是动态映射
- Broadcast：该 PVC 允许广播包的通过
- Active：该 PVC 是激活的

该命令是很重要的一条命令，如果在映射表中不存在映射，路由器将无法通信。可以使用名命令“clear frame-relay inarp”命令清除无效的帧中继映射表。

```
R1#show frame-relay pvc
PVC Statistics for interface Serial0/0 (Frame Relay DTE)
DLCI = 103, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/0
    input pkts 102024        output pkts 116191        in bytes 13974906
    out bytes 14707805       dropped pkts 0           in FECN pkts 287
    in BECN pkts 290         out FECN pkts 0          out BECN pkts 0
    in DE pkts 102024        out DE pkts 0
      pvc create time 1w1d, last time pvc status changed 1w1d
     ……
```

从命令输出中可以得到的信息有：

- DLCI = 103：表明该 PVC 的 DLCI 为 103
- PVC STATUS = ACTIVE：表明 PVC 的状态是激活的；若 PVC STATUS = INACTIVE——表明远端路由器没正确配置；若 PVC STATUS = DELETED——表明输入了错误的 DLCI，该 PVC 不存在。

```
R1#show frame-relay lmi
    LMI Statistics for interface Serial0/0 (Frame Relay DTE) LMI TYPE = CISCO
    Invalid Unnumbered info 0              Invalid Prot Disc 0
    Invalid dummy Call Ref 0               Invalid Msg Type 0
    Invalid Status Message 0               Invalid Lock Shift 0
    Invalid Information ID 0               Invalid Report IE Len 0
    Invalid Report Request 0               Invalid Keep IE Len 0
    Num Status Enq. Sent 74859             Num Status msgs Rcvd 74857
    Num Update Status Rcvd 0               Num Status Timeouts 2
```

从命令输出中可以得到的信息有：

- LMI TYPE = CISCO：表明帧中继 LMI 类型为 cisco；
- Frame Relay DTE：这是帧中继 DTE
- Num Status Enq. Sent 74859：表明路由器向帧中继交换机发送的 LMI 状态查询消息的数量；
- Num Status msgs Rcvd 74857：表明路由器从帧中继交换机收到的 LMI 状态信息的数量。

8.4 实验 3:帧中继上的 RIP

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 帧中继上路由协议运行的特殊性

（2） 水平分割

2. 实验拓扑

如图 8-4。

3. 实验步骤

在实验 2 的基础上继续本实验。

（1） 配置 RIP：

```
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config)#router rip
R1(config-router)#network 1.0.0.0
R1(config-router)#network 192.168.123.0

R3(config)#interface Loopback0
R3(config-if)#ip address 3.3.3.3 255.255.255.0
R3(config)#router rip
R3(config-router)#network 3.0.0.0
R3(config-router)#network 192.168.123.0

R4(config)#interface Loopback0
R4(config-if)#ip address 4.4.4.4 255.255.255.0
R4(config)#router rip
R4(config-router)#network 4.0.0.0
R4(config-router)#network 192.168.123.0
```

（2） 检查路由表、测试

在各个路由器上检查路由表，并测试从环回口之间的互相 ping。

```
R3#show ip route
（此处省略）
C    192.168.123.0/24 is directly connected, Serial0/0/1
R    1.0.0.0/8 [120/1] via 192.168.123.1, 00:00:26, Serial0/0/1
     3.0.0.0/24 is subnetted, 1 subnets
C       3.3.3.0 is directly connected, Loopback0
R    4.0.0.0/8 [120/2] via 192.168.123.1, 00:00:26, Serial0/0/1
//看到正常的路由表，注意 RIP 路由表中的下一跳

R3#ping 4.4.4.4 source loopback 0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
Packet sent with a source address of 3.3.3.3
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 52/54/56 ms
```

//以上表明从 R3 的 loopback0 接口能 ping 通 R4 的 loopback0 接口

R3#ping 4.4.4.4

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
....
```

//这里 ping 命令没指明源地址，则 ICMP 数据包的源 IP 为 192.168.123.3，目标为 4.4.4.4。R3 路由表查询路由表得知该数据包应该发送给 192.168.123.1，而 192.168.123.1 的帧中继映射 DLCI 为 301；数据包到达 R1，R1 路由表查询路由表得知该数据包应该发送给 192.168.123.4，而 192.168.123.4 的帧中继映射 DLCI 为 104。R4 收到数据包，进行响应，ICMP 数据包的源 IP 为 4.4.4.4，目标为 192.168.123.3；R4 有 192.168.123.0/24 的直连路由，然而却没有 192.168.123.3 的帧中继映射，因此无法进行封装。为了解决该问题，可以在 R4 中增加映射：

R4(config)#int s0/0/1

R4(config-if)#frame-relay map ip 192.168.123.3 401

这样从 R3 就可以直接 ping 通 R4 的 loopback0 接口了。

（3） 水平分割问题

R1#show ip int s0/0/0

```
Serial0/0/0 is up, line protocol is up
  Internet address is 192.168.123.1/24
（此处省略）
  Security level is default
  Split horizon is disabled      //接口封装了帧中继后，水平分割被自动关闭
  ICMP redirects are always sent
   （此处省略）
```

在 R1 上重新打开水平分割，在各路由器上检查路由表。

R1(config)#int Serial0/0/0

R1(config-if)#ip split-horizon

R1#clear ip route *   //清除路由表

R1#show ip route

```
C    192.168.123.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
R    3.0.0.0/8 [120/1] via 192.168.123.3, 00:00:01, Serial0/0/0
R    4.0.0.0/8 [120/1] via 192.168.123.4, 00:00:01, Serial0/0/0
```

//R1 可以获得 R3 和 R4 的环回口路由

R3#clear ip route *

R3#show ip route

```
C    192.168.123.0/24 is directly connected, Serial0/0/1
R    1.0.0.0/8 [120/1] via 192.168.123.1, 00:00:00, Serial0/0/1
     3.0.0.0/24 is subnetted, 1 subnets
C       3.3.3.0 is directly connected, Loopback0
```

//R3 只能获得 R1 的环回口路由，这是由于 R1 上的水平分割开启后，R1 从 R4 接收到 R4 公告的路由后，不从帧中继口发送出来，导致 R3 没有接收到 R4 上公告的路由

```
R3#clear ip route *
R4#show ip route
C    192.168.123.0/24 is directly connected, Serial0/0/1
R    1.0.0.0/8 [120/1] via 192.168.123.1, 00:00:01, Serial0/0/1
     4.0.0.0/24 is subnetted, 1 subnets
C       4.4.4.0 is directly connected, Loopback0
```

//R4 也只能获得 R1 的环回口路由

8.5 实验 4:帧中继点到多点子接口

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 点到多点子接口的配置

2. 实验拓扑

如图 8-4，R3 和 R4 使用帧中继主接口，在 R1 上创建点到多点子接口。

3. 实验步骤

在实验 1 的基础上继续本实验。

(1) 对主接口进行配置

```
R1(config)#interface serial0/0/0
R1(config-if)#no ip address     //注：主接口下不需要 IP 地址
R1(config-if)#encap frame-relay      //注：封装帧中继
R1(config-if)#no frame-relay inverse-arp   //注：通常需要关闭主接口下的 IARP
R1(config-if)#no shutdown
```

(2) 创建点到多点子接口

```
R1(config)#int s0/0/0.1 multipoint     //注：创建点到多点子接口
```

//这里命令“interface serial *slot-number/interface-number.subinterface-number* { multipoint | point-to-point }”用来创建子接口，其中：

- *slot-number/interface-number*：即本物理接口的号码
- *subinterface-number*：是子接口号，用户可以根据自己喜好来确定
- multipoint 和 point-to-point：属于必须选择的项，是子接口的类型，要么是点到多点，要么是点到点。

```
R1(config-subif)#ip address 192.168.123.1 255.255.255.0
R1(config-subif)#frame-relay map ip 192.168.123.3 103 broadcast
R1(config-subif)#frame-relay map ip 192.168.123.4 104 broadcast
```

//以上是配置帧中继映射

(3) R1 上配置路由协议：

```
R1(config)#router rip
```

```
R1(config-router)#network 1.0.0.0
R1(config-router)#network 192.168.123.0
```
(4)　R3、R4 完整的配置如下：
```
R3(config)#interface serial 0/0/1
R3(config-if)#ip address 192.168.123.3 255.255.255.0
R3(config-if)#encapsulation frame-relay
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#frame-relay map ip 192.168.123.1 301 broadcast
R3(config-if)＃no shutdown
R3(config)#router rip
R3(config-router)#network 3.0.0.0
R3(config-router)#network 192.168.123.0

R4(config)#interface serial 0/0/1
R4(config-if)#ip address 192.168.123.4 255.255.255.0
R4(config-if)#encapsulation frame-relay
R4(config-if)#no frame-relay inverse-arp
R4(config-if)#frame-relay map ip 192.168.123.1 401 broadcast
R4(config-if)＃no shutdown
R4(config)#router rip
R4(config-router)#network 4.0.0.0
R4(config-router)#network 192.168.123.0
```

【提示】可以使用“no interface s0/0/0.1”命令来删除子接口，然而需要重新启动路由器，该子接口才真正被删除。

4. 实验调试

在各个路由器上检查路由表，注意路由的下一跳。
```
R1#show ip route
（此处省略）
C    192.168.123.0/24 is directly connected, Serial0/0/0.1
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
R    3.0.0.0/8 [120/1] via 192.168.123.3, 00:00:19, Serial0/0/0.1
R    4.0.0.0/8 [120/1] via 192.168.123.4, 00:00:16, Serial0/0/0.1
```
//R1 可以获得 R3 和 R4 的环回口路由

```
R3#show ip route
（此处省略）
C    192.168.123.0/24 is directly connected, Serial0/0/1
R    1.0.0.0/8 [120/1] via 192.168.123.1, 00:00:01, Serial0/0/1
     3.0.0.0/24 is subnetted, 1 subnets
C       3.3.3.0 is directly connected, Loopback0
```
//R3 只能获得 R1 的环回口路由，这是由于默认时 R1 的点到多点子接口水平分割是开启的，

可以使用命令“ip split-horizon”在子接口下关闭水平分割。

```
R4#show ip route
（此处省略）
C     192.168.123.0/24 is directly connected, Serial0/0/1
R     1.0.0.0/8 [120/1] via 192.168.123.1, 00:00:01, Serial0/0/1
      4.0.0.0/24 is subnetted, 1 subnets
C        4.4.4.0 is directly connected, Loopback0
//R4 同样只能获得 R1 的环回口路由
```

8.6 实验 5:帧中继点到点子接口

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 点到点子接口的配置

2. 实验拓扑

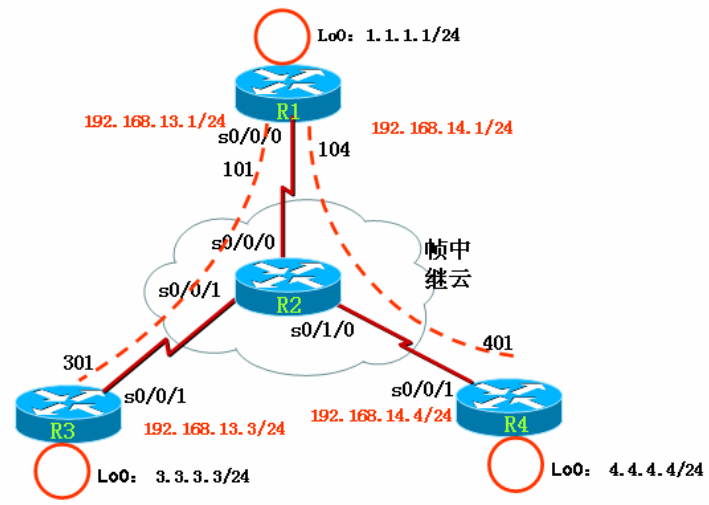

图 8-5 实验 5 拓扑图

3. 实验步骤

(1) 对主接口进行配置

```
R1(config)#interface serial0/0/0
R1(config-if)#no ip address
R1(config-if)#encap frame-relay
R1(config-if)#no frame-relay inverse-arp
R1(config-if)#no shutdown
```

(2) 创建两个点到点子接口

```
R1(config)#int s0/0/0.3 point-to-point     //注：创建点到点子接口
R1(config-subif)#ip address 192.168.13.1 255.255.255.0
R1(config-subif)#frame-relay interface-dlci 103
```

//在接口下不能使用“frame-relay map ip”命令来配置帧中继的映射，而是改用命令“frame-relay interface-dlci 103” 。

```
R1(config)#int s0/0/0.4 point-to-point
R1(config-subif)#ip address 192.168.14.1 255.255.255.0
R1(config-subif)#frame-relay interface-dlci 104
```

(3) R1 上配置路由协议：

```
R1(config)#router rip
R1(config-router)#network 1.0.0.0
R1(config-router)#network 192.168.13.0
R1(config-router)#network 192.168.14.0
```

(4) R3、R4 完整的配置如下：

```
R3(config)#interface serial 0/0/1
R3(config-if)#no ip address
R3(config-if)#encapsulation frame-relay
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#no shutdown
R3(config-if)#exit
R3(config)#interface serial 0/0/1.1 point-to-point
R3(config-subif)#ip address 192.168.13.3 255.255.255.0
R3(config-subif)#frame-relay interface-dlci 301
R3(config-subif)#exit
R3(config)#router rip
R3(config-router)#network 3.0.0.0
R3(config-router)#network 192.168.13.0

R4(config)#interface serial 0/0/1
R4(config-if)#no ip address
R4(config-if)#encapsulation frame-relay
R4(config-if)#no frame-relay inverse-arp
R4(config-if)#no shutdown
R4(config-if)#exit
R4(config)#interface serial 0/0/1.1 point-to-point
R4(config-subif)#ip address 192.168.14.4 255.255.255.0
R4(config-subif)#frame-relay interface-dlci 401
R4(config-subif)#exit
R4(config)#router rip
R4(config-router)#network 4.0.0.0
R4(config-router)#network 192.168.14.0
```

4. 实验调试

在各个路由器上检查路由表，注意路由的下一跳。

```
R3#show ip route
```

（此处省略）

```
R    1.0.0.0/8 [120/1] via 192.168.13.1, 00:00:14, Serial0/0/1.1
C    192.168.13.0/24 is directly connected, Serial0/0/1.1
R    192.168.14.0/24 [120/1] via 192.168.13.1, 00:00:14, Serial0/0/1.1
     3.0.0.0/24 is subnetted, 1 subnets
C       3.3.3.0 is directly connected, Loopback0
R    4.0.0.0/8 [120/2] via 192.168.13.1, 00:00:14, Serial0/0/1.1
```

8.7 本章小结

本章简要介绍了帧中继技术的使用场合，重点介绍理解帧中继的关键术语：DLCI、PVC和帧中继映射。DLCI 实际上就是数据链路层地址，路由器在帧中继链路上发送数据包，要获得该地址才能封装帧。可以手工或者使用 Inverse-arp 来把网络层地址和 DLCI 进行映射。子接口是逻辑接口，子接口的引入使得在帧中继链路上运行路由协议变得容易，也使得帧中继的配置更加灵活。表 8-1 是本章出现的命令。

表 8-1 本章命令汇总

命令	作用
frame-relay switching	把路由器当成帧中继交换机
encapsulation frame-relay	接口封装成帧中继
frame-relay lmi-type cisco	配置 LMI 的类型
frame-relay intf-type dce	配置接口是帧中继的 DCE 还是 DTE
frame-relay route	配置帧中继交换表
show frame-relay route	显示帧中继交换表
show frame pvc	显示帧中继 PVC 状态
show frame lmi	显示帧中继 LMI 信息
show frame-relay map	查看帧中继映射
no frame-relay inverse-arp	关闭帧中继自动映射
ip split-horizon	打开水平分割
int s0/0/0.1 multipoint	创建点到多点子接口
int s0/0/0.3 point-to-point	创建点到点子接口
frame-relay interface-dlci 104	在点到点子接口上配置 DLCI

第 9 章　ACL

随着大规模开放式网络的开发，网络面临的威胁也就越来越多。网络安全问题成为网络管理员最为头疼的问题。一方面，为了业务的发展，必须允许对网络资源的开发访问，另一方面，又必须确保数据和资源的尽可能安全。网络安全采用的技术很多，而通过访问控制列表（ACL）可以对数据流进行过滤，是实现基本的网络安全手段之一。本章只研究基于 IP 的 ACL。

9.1　ACL 概述

访问控制列表简称为 ACL，它使用包过滤技术，在路由器上读取第三层及第四层包头中的信息如源地址、目的地址、源端口、目的端口等，根据预先定义好的规则对包进行过滤，从而达到访问控制的目的。ACL 分很多种，不同场合应用不同种类的 ACL。

1. 标准 ACL

标准 ACL 最简单，是通过使用 IP 包中的源 IP 地址进行过滤，表号范围 1-99 或 1300-1999；

2. 扩展 ACL

扩展 ACL 比标准 ACL 具有更多的匹配项，功能更加强大和细化，可以针对包括协议类型、源地址、目的地址、源端口、目的端口、TCP 连接建立等进行过滤，表号范围 100-199 或 2000-2699；

3. 命名 ACL

以列表名称代替列表编号来定义 ACL，同样包括标准和扩展两种列表。

在访问控制列表的学习中，要特别注意以下两个术语。

1. 通配符掩码：一个 32 比特位的数字字符串，它规定了当一个 IP 地址与其他的 IP 地址进行比较时，该 IP 地址中哪些位应该被忽略。通配符掩码中的“1”表示忽略 IP 地址中对应的位，而“0”则表示该位必须匹配。两种特殊的通配符掩码是“255.255.255.255”和“0.0.0.0”，前者等价于关键字“any”，而后者等价于关键字“host”；

2. Inbound 和 outbound：当在接口上应用访问控制列表时，用户要指明访问控制列表是应用于流入数据还是流出数据。

总之，ACL 的应用非常广泛，它可以实现如下的功能：

1. 拒绝或允许流入（或流出）的数据流通过特定的接口；
2. 为 DDR 应用定义感兴趣的数据流；
3. 过滤路由更新的内容；
4. 控制对虚拟终端的访问；
5. 提供流量控制。

9.2 实验 1：标准 ACL

1.实验目的

通过本实验可以掌握：

（1）ACL 设计原则和工作过程

（2）定义标准 ACL

（3）应用 ACL

（4）标准 ACL 调试

2.拓扑结构

实验拓扑如图 9-1 所示。

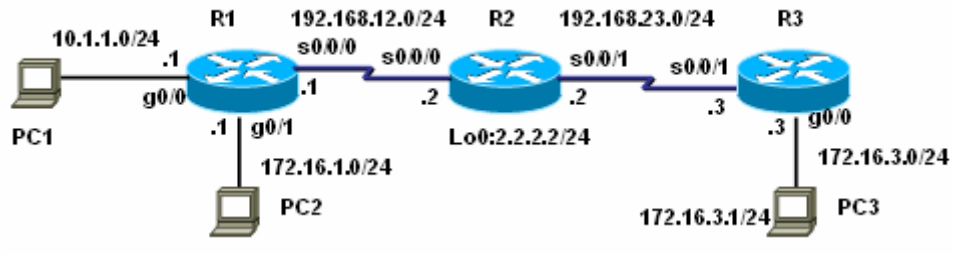

图 9-1 标准 ACL 配置

本实验拒绝 PC2 所在网段访问路由器 R2,同时只允许主机 PC3 访问路由器 R2 的 TELNET 服务。整个网络配置 EIGRP 保证 IP 的连通性。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router eigrp 1
R1(config-router)#network 10.1.1.0 0.0.0.255
R1(config-router)#network 172.16.1.0 0.0.0.255
R1(config-router)#network 192.168.12.0
R1(config-router)#no auto-summary
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router eigrp 1
R2(config-router)#network 2.2.2.0 0.0.0.255
R2(config-router)#network 192.168.12.0
R2(config-router)#network 192.168.23.0
R2(config-router)#no auto-summary
R2(config)#access-list 1 deny   172.16.1.0 0.0.0.255 //定义 ACL
R2(config)#access-list 1 permit any
R2(config)#interface Serial0/0/0
R2(config-if)#ip access-group 1 in    //在接口下应用 ACL
R2(config)#access-list 2 permit 172.16.3.1
R2(config-if)#line vty 0 4
R2(config-line)#access-class 2 in    //在 vty 下应用 ACL
R2(config-line)#password cisco
R2(config-line)#login
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router eigrp 1
R3(config-router)#network 172.16.3.0 0.0.0.255
R3(config-router)#network 192.168.23.0
R3(config-router)#no auto-summary
```

【技术要点】

（1）ACL 定义好，可以在很多地方应用，接口上应用只是其中之一，其它的常用应用包括在 route map 中的 match 应用（21 章介绍）和在 vty 下用“access-class”命令调用，来控制 telnet 的访问；

（2）访问控制列表表项的检查按自上而下的顺序进行，并且从第一个表项开始，所以必须考虑在访问控制列表中定义语句的次序；

（3）路由器不对自身产生的 IP 数据包进行过滤；

（4）访问控制列表最后一条是隐含的拒绝所有；

（5）每一个路由器接口的每一个方向，每一种协议只能创建一个 ACL；

（6）“access-class”命令只对标准 ACL 有效。

4.实验调试

在 PC1 网络所在的主机上 ping 2.2.2.2，应该通，在 PC2 网络所在的主机上 ping 2.2.2.2，应该不通，在主机 PC3 上 TELNET 2.2.2.2，应该成功。

（1）show ip access-lists

该命令用来查看所定义的 IP 访问控制列表。

```
R2#show ip access-lists
Standard IP access list 1
    10 deny   172.16.1.0, wildcard bits 0.0.0.255 (11 matches)
    20 permit any (405 matches)
Standard IP access list 2
    10 permit 172.16.3.1 (2 matches)
```

以上输出表明路由器 R2 上定义的标准访问控制列表为“1”和“2”，括号中的数字表示匹配条件的数据包的个数，可以用“clear access-list counters”将访问控制列表计数器清零。

（2）show ip interface

```
R2#show ip interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Internet address is 192.168.12.2/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Multicast reserved groups joined: 224.0.0.10
  Outgoing access list is not set
  Inbound  access list is 1
  ......
```

以上输出表明在接口 s0/0/0 的入方向应用了访问控制列表 1。

9.3 实验 2：扩展 ACL

1.实验目的

通过本实验可以掌握：

（1）定义扩展 ACL

（2）应用扩展 ACL

（3）扩展 ACL 调试

2.拓扑结构

实验拓扑如图 9-1 所示。

本实验要求只允许 PC2 所在网段的主机访问路由器 R2 的 WWW 和 TELNET 服务，并拒绝 PC3 所在网段 PING 路由器 R2。删除实验 1 中定义的 ACL，保留 EIGRP 的配置。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#access-list 100 permit tcp 172.16.1.0 0.0.0.255 host 2.2.2.2 eq www
R1(config)#access-list 100 permit tcp 172.16.1.0 0.0.0.255 host 192.168.12.2 eq www
R1(config)#access-list 100 permit tcp 172.16.1.0 0.0.0.255 host 192.168.23.2 eq www
R1(config)#access-list 100 permit tcp 172.16.1.0 0.0.0.255 host 2.2.2.2 eq telnet
R1(config)#access-list 100 permit tcp 172.16.1.0 0.0.0.255 host 192.168.12.2 eq telnet
R1(config)#access-list 100 permit tcp 172.16.1.0 0.0.0.255 host 192.168.23.2 eq telnet
R1(config)#interface g0/0
R1(config-if)#ip access-group 100 in
```

（2）步骤 2：配置路由器 R2

```
R2(config)#no access-list 1  //删除 ACL
R2(config)#no access-list 2
R2(config)#ip http server   //将路由器配置成 WEB 服务器
R2(config)#line vty 0 4
R2(config-line)#password cisco
R2(config-line)#login
```

（3）步骤 3：配置路由器 R3

```
R3(config)#access-list 101 deny icmp 172.16.3.0 0.0.0.255 host 2.2.2.2 log
R3(config)#access-list 101 deny icmp 172.16.3.0 0.0.0.255 host 192.168.12.2 log
R3(config)#access-list 101 deny icmp 172.16.3.0 0.0.0.255 host 192.168.23.2 log
R3(config)#access-list 101 permit ip any any
R3(config)#interface g0/0
R3(config-if)#ip access-group 101 in
```

【技术要点】

（1）参数“log”会生成相应的日志信息，用来记录经过 ACL 入口的数据包的情况；

（2）尽量考虑将扩展的访问控制列表放在靠近过滤源的位置上，这样创建的过滤器就不会反过来影响其它接口上的数据流。另外，尽量使标准的访问控制列表靠近目的，由于标准访问控制列表只使用源地址，如果将其靠近源会阻止数据包流向其他端口。

4.实验调试

（1)分别在 PC2 上访问路由器 R2 的 TELNET 和 WWW 服务,然后查看访问控制列表 100：

```
R1#show ip access-lists
Extended IP access list 100
```

```
    10 permit tcp 172.16.1.0 0.0.0.255 host 2.2.2.2 eq www (8 matches)
    20 permit tcp 172.16.1.0 0.0.0.255 host 192.168.12.2 eq www
    30 permit tcp 172.16.1.0 0.0.0.255 host 192.168.23.2 eq www
    40 permit tcp 172.16.1.0 0.0.0.255 host 2.2.2.2 eq telnet (20 matches)
    50 permit tcp 172.16.1.0 0.0.0.255 host 12.12.12.2 eq telnet (4 matches)
    60 permit tcp 172.16.1.0 0.0.0.255 host 23.23.23.2 eq telnet (4 matches)
```

（2）在 PC3 所在网段的主机 ping 路由器 R2，路由器 R3 会出现下面的日志信息：

```
*Feb 25 17:35:46.383: %SEC-6-IPACCESSLOGDP: list 101 denied icmp 172.16.3.1 -> 2.2.2.2
(0/0), 1 packet
*Feb 25 17:41:08.959: %SEC-6-IPACCESSLOGDP: list 101 denied icmp 172.16.3.1 -> 2.2.2.2
(0/0), 4 packets
*Feb 25 17:42:46.919: %SEC-6-IPACCESSLOGDP: list 101 denied icmp 172.16.3.1 ->
192.168.12.2 (0/0), 1 packet
*Feb 25 17:42:56.803: %SEC-6-IPACCESSLOGDP: list 101 denied icmp 172.16.3.1 ->
192.168.23.2 (0/0), 1 packet
```

以上输出说明在访问控制列表 101 在有匹配数据包的时候，系统作了日志。

（3）在路由器 R3 上查看访问控制列表 101：

```
R3#show access-lists
Extended IP access list 101
    10 deny icmp 172.16.3.0 0.0.0.255 host 2.2.2.2 log (5 matches)
    20 deny icmp 172.16.3.0 0.0.0.255 host 192.168.12.2 log (5 matches)
    30 deny icmp 172.16.3.0 0.0.0.255 host 192.168.23.2 log (5 matches)
    40 permit ip any any (6 matches)
```

9.4 实验 3：命名 ACL

命名 ACL 允许在标准 ACL 和扩展 ACL 中，使用字符串代替前面所使用的数字来表示 ACL。命名 ACL 还可以被用来从某一特定的 ACL 中删除个别的控制条目，这样可以让网络管理员方便地修改 ACL。

1.实验目的

通过本实验可以掌握：

（1）定义命名 ACL

（2）应用命名 ACL

2.拓扑结构

实验拓扑如图 9-1 所示。

3.实验步骤

本实验给出如何用命名 ACL 来实现 9.2 实验 1 中和 9.3 实验 2 中的要求。

（1）在路由器 R2 上配置命名的标准 ACL 实现 9.2 实验 1 的要求

```
R2(config)#ip access-list standard stand
R2(config-std-nacl)#deny 172.16.1.0 0.0.0.255
R2(config-std-nacl)#permit any
R2(config)#interface Serial0/0/0
R2(config-if)#ip access-group stand in
R2(config)#ip access-list standard class
```

```
R2(config-std-nacl)#permit 172.16.3.1
R2(config-if)#line vty 0 4
R2(config-line)#access-class class in
```

（2）在路由器 R2 上查看命名访问控制列表

```
R2#show access-lists
Standard IP access list class
    10 permit 172.16.3.1
Standard IP access list stand
    10 deny   172.16.1.0, wildcard bits 0.0.0.255
    20 permit any (42 matches)
```

（3）在路由器 R1 和 R3 上配置命名的扩展 ACL 实现 9.3 实验 2 的要求

```
R1(config)#ip access-list extended ext1
R1(config-ext-nacl)#permit tcp 172.16.1.0 0.0.0.255 host 2.2.2.2 eq www
R1(config-ext-nacl)#permit tcp 172.16.1.0 0.0.0.255 host 192.168.12.2 eq www
R1(config-ext-nacl)#permit tcp 172.16.1.0 0.0.0.255 host 192.168.23.2 eq www
R1(config-ext-nacl)#permit tcp 172.16.1.0 0.0.0.255 host 2.2.2.2 eq telnet
R1(config-ext-nacl)#permit tcp 172.16.1.0 0.0.0.255 host 192.168.12.2 eq telnet
R1(config-ext-nacl)#permit tcp 172.16.1.0 0.0.0.255 host 192.168.23.2 eq telnet
R1(config)#interface g0/0
R1(config-if)#ip access-group ext1 in
R3(config)#ip access-list extended ext3
R3(config-ext-nacl)#deny icmp 172.16.3.0 0.0.0.255 host 2.2.2.2 log
R3(config-ext-nacl)#deny icmp 172.16.3.0 0.0.0.255 host 192.168.12.2 log
R3(config-ext-nacl)#deny icmp 172.16.3.0 0.0.0.255 host 192.168.23.2 log
R3(config-ext-nacl)#permit ip any any
R3(config)#interface g0/0
R3(config-if)#ip access-group ext3 in
```

（4）在路由器 R1 和 R3 上查看命名访问控制列表

```
R1#show access-lists
Extended IP access list ext1
    10 permit tcp 172.16.1.0 0.0.0.255 host 2.2.2.2 eq www
    20 permit tcp 172.16.1.0 0.0.0.255 host 192.168.12.2 eq www
    30 permit tcp 172.16.1.0 0.0.0.255 host 192.168.23.2 eq www
    40 permit tcp 172.16.1.0 0.0.0.255 host 2.2.2.2 eq telnet
    50 permit tcp 172.16.1.0 0.0.0.255 host 192.168.12.2 eq telnet
    60 permit tcp 172.16.1.0 0.0.0.255 host 192.168.23.2 eq telnet
R3#show access-lists
Extended IP access list ext3
    10 deny icmp 172.16.3.0 0.0.0.255 host 2.2.2.2 log
    20 deny icmp 172.16.3.0 0.0.0.255 host 192.168.12.2 log
    30 deny icmp 172.16.3.0 0.0.0.255 host 192.168.23.2 log
```

40 permit ip any any

9.5 实验 4：基于时间 ACL

1.实验目的

通过本实验可以掌握：

（1）定义 time-range

（2）配置基于时间 ACL

（3）基于时间 ACL 调试

2.拓扑结构

实验拓扑如图 9-1 所示。

本实验要求只允许PC3主机在周一到周五的每天的8:00-18:00访问路由器R2的TELNET服务。

3.实验步骤

R3(config)#time-range time　　//定义时间范围

R3(config-time-range)#periodic weekdays 8:00 to 18:00

R3(config)#access-list 111 permit tcp host 172.16.3.1 host 2.2.2.2 eq telnet time-range time　//在访问控制列表中调用 time-range

R3(config)#access-list 111 permit tcp host 172.16.3.1 host 192.168.12.2 eq telnet time-range time

R3(config)#access-list 111 permit tcp host 172.16.3.1 host 192.168.23.2 eq telnet time-range time

R3(config)#interface g0/0

R3(config-if)#ip access-group 111 in

4.实验调试

（1）用“clock”命令将系统时间调整到周一至周五的 8:00-18:00 范围内，然后在 PC3 上 TELNET 路由器 R2，此时可以成功，然后查看访问控制列表 111：

R3#show access-lists

Extended IP access list 111

10 permit tcp host 172.16.3.1 host 2.2.2.2 eq telnet time-range time (active) (15 matches)

20 permit tcp host 172.16.3.1 host 192.168.12.2 eq telnet time-range time (active)

30 permit tcp host 172.16.3.1 host 192.168.23.2 eq telnet time-range time (active)

（2）用“clock”命令将系统时间调整到 8:00-18:00 范围之外，然后在 PC3 上 TELNET 路由器 R2，此时不可以成功，然后查看访问控制列表 111：

R3#show access-lists

Extended IP access list 111

10 permit tcp host 172.16.3.1 host 2.2.2.2 eq telnet time-range time (inactive) (45 matches)

20 permit tcp host 172.16.3.1 host 192.168.12.2 eq telnet time-range time (inactive)

30 permit tcp host 172.16.3.1 host 192.168.23.2 eq telnet time-range time (inactive)

（3）show time-range

该命令用来查看定义的时间范围。

R3#show time-range

time-range entry: time (active)

periodic weekdays 8:00 to 18:00

used in: IP ACL entry

used in: IP ACL entry

used in: IP ACL entry

以上输出表示在 3 条 ACL 中调用了该 time-range。

9.6 实验 5：动态 ACL

动态 ACL 是 Cisco IOS 的一种安全特性，它使用户能在防火墙中临时打开一个缺口，而不会破坏其它已配置了的安全限制。

1.实验目的

通过本实验可以掌握：

（1）动态 ACL 工作原理

（2）配置 VTY 本地登录

（3）配置动态 ACL

（4）动态 ACL 调试

2.拓扑结构

实验拓扑如图 9-1 所示。

本实验要求如果 PC3 所在网段想要访问路由器 R2 的 WWW 服务，必须先 TELNET 路由器 R2 成功后才能访问。

3.实验步骤

R2(config)#username ccie password cisco   //建立本地数据库

R2(config)#access-list 120 permit tcp 172.16.3.0 0.0.0.255 host 2.2.2.2 eq telnet //打开 TELNET 访问权限

R2(config)#access-list 120 permit tcp 172.16.3.0 0.0.0.255 host 12.12.12.2 eq telnet

R2(config)#access-list 120 permit tcp 172.16.3.0 0.0.0.255 host 23.23.23.2 eq telnet

R2(config)#access-list 120 permit eigrp any any //允许 EIGRP 协议

R2(config)#access-list 120 dynamic test timeout 120 permit ip 172.16.3.0 0.0.0.255 host 2.2.2.2

//“dynamic”定义动态 ACL，“timeout”定义动态 ACL 绝对的超时时间

R2(config)#access-list 120 dynamic test1 timeout 120 permit ip 172.16.3.0 0.0.0.255 host 23.23.23.2

R2(config)#access-list 120 dynamic test2 timeout 120 permit ip 172.16.3.0 0.0.0.255 host 12.12.12.2

R2(config)#interface s0/0/1

R2(config-if)#ip access-group 120 in

R2(config)#line vty 0 4

R2(config-line)#login local   //VTY 使用本地验证

R2(config-line)#autocommand access-enable host timeout 5

//在一个动态 ACL 中创建一个临时性的访问控制列表条目，“timeout”定义了空闲超时值，空闲超时值必须小于绝对超时值。

【技术要点】

如果用参数“host”，那么临时性条目将只为用户所用的单个 IP 地址创建，如果不使用，

那用户的整个网络都将被该临时性条目允许。

4.实验调试

（1）没有 TELNET 路由器 R2，在 PC3 上直接访问路由器 R2 的 WWW 服务，不成功，路由器 R2 的访问控制列表：

```
R2#show access-lists
Extended IP access list 120
    10 permit tcp 172.16.3.0 0.0.0.255 host 2.2.2.2 eq telnet (114 matches)
    20 permit tcp 172.16.3.0 0.0.0.255 host 12.12.12.2 eq telnet
    30 permit tcp 172.16.3.0 0.0.0.255 host 23.23.23.2 eq telnet
    40 permit eigrp any any (159 matches)
    50 Dynamic test permit ip 172.16.3.0 0.0.0.255 host 2.2.2.2
    60 Dynamic test1 permit ip 172.16.3.0 0.0.0.255 host 23.23.23.2
    70 Dynamic test2 permit ip 172.16.3.0 0.0.0.255 host 12.12.12.2
```

（2）TELNET 路由器 R2 成功之后，在 PC3 上访问路由器 R2 的 WWW 服务，成功，路由器 R2 的访问控制列表：

```
R2#show access-lists
Extended IP access list 120
    10 permit tcp 172.16.3.0 0.0.0.255 host 2.2.2.2 eq telnet (114 matches)
    20 permit tcp 172.16.3.0 0.0.0.255 host 12.12.12.2 eq telnet
    30 permit tcp 172.16.3.0 0.0.0.255 host 23.23.23.2 eq telnet
    40 permit eigrp any any (159 matches)
    50 Dynamic test permit ip 172.16.3.0 0.0.0.255 host 2.2.2.2
       permit ip host 172.16.3.1 host 2.2.2.2 (15 matches) (time left 288)
    60 Dynamic test1 permit ip 172.16.3.0 0.0.0.255 host 23.23.23.2
    70 Dynamic test2 permit ip 172.16.3.0 0.0.0.255 host 12.12.12.2
```

从(1)和(2)的输出结果可以看到，从主机 172.16.3.1 telnet 2.2.2.2,如果通过认证，该 telnet 会话就会被切断，IOS 软件将在动态访问控制列表中动态建立一临时条目“**permit ip host 172.16.3.1 host 2.2.2.2**”，此时在主机 172.16.3.1 上访问 2.2.2.2 的 Web 服务，成功。

9.7 实验 6：自反 ACL

1.实验目的

通过本实验可以掌握：

（1）自反 ACL 工作原理

（2）配置自反 ACL

（3）自反 ACL 调试

2.拓扑结构

实验拓扑如图 9-2 所示。

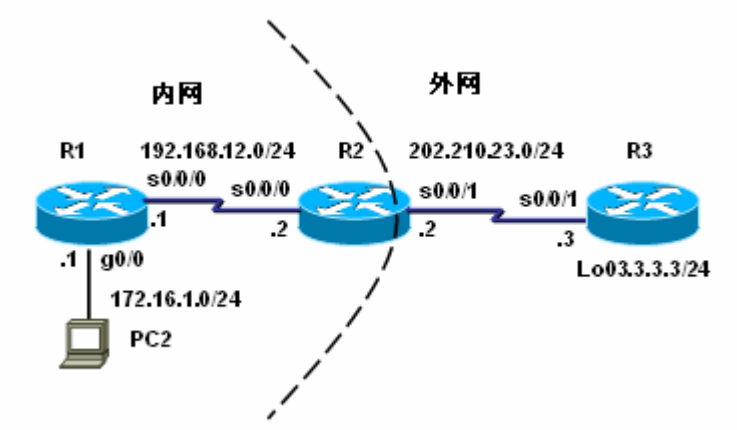

图 9-2 自反 ACL 配置

本实验要求内网可以主动访问外网，但是外网不能主动访问内网，从而有效保护内网。

3. 实验步骤

（1）步骤 1：分别在路由器 R1 和 R3 配置默认路由确保 IP 连通性

```
R1(config)#ip route 0.0.0.0 0.0.0.0 192.168.12.2
R3(config)#ip route 0.0.0.0 0.0.0.0 202.210.23.2
```

（2）步骤 2：在路由器 R2 上配置自反 ACL

```
R2(config)#ip access-list extended ACLOUT
R2(config-ext-nacl)#permit tcp any any reflect REF  //定义自反 ACL
R2(config-ext-nacl)#permit udp any any reflect REF
R2(config)#ip access-list extended ACLIN
R2(config-ext-nacl)#evaluate REF //评估反射
R2(config)#int s0/0/1
R2(config-if)#ip access-group ACLOUT out
R2(config-if)#ip access-group ACLIN in
```

【技术要点】

1. 自反 ACL 永远是 permit 的；
2. 自反 ACL 允许高层 Session 信息的 IP 包过滤；
3. 利用自反 ACL 可以只允许出去的流量，但是阻止从外部网络产生的向内部网络的流量，从而可以更好地保护内部网络；
4. 自反 ACL 是在有流量产生时（如出方向的流量）临时自动产生的，并且当 Session 结束条目就删除；
5. 自反 ACL 不是直接被应用到某个接口下的，而是嵌套在一个扩展命名访问列表下的。

4. 实验调试

（1）同时在路由器 R1 和 R3 都打开 TELNET 服务，在 R1(从内网到外网)TELNET 路由器 R3 成功，同时在路由器 R2 上查看访问控制列表：

```
R2#show access-lists
Extended IP access list ACLIN
    10 evaluate REF
Extended IP access list ACLOUT
```

```
        10 permit tcp any any reflect REF
        20 permit udp any any reflect REF
    Reflexive IP access list REF
         permit tcp host 202.210.23.3 eq telnet host 192.168.12.1 eq 11002 (48 matches) (time
left 268)
```

以上输出说明自反列表是在有内部到外部 TELNET 流量经过的时候，临时自动产生一条列表。

（2）在路由器 R1 打开 TELNET 服务，在 R3(从外网到内网)TELNET 路由器 R1 不能成功，同时在路由器 R2 上查看访问控制列表：

```
R2#show access-lists
Extended IP access list ACLIN
    10 evaluate REF
Extended IP access list ACLOUT
    10 permit tcp any any reflect REF
    20 permit udp any any reflect REF
Reflexive IP access list REF
```

以上输出说明自反列表是在有外部到内部 TELNET 流量经过的时候，不会临时自动产生一条列表，所以不能访问成功。

9.8 ACL 命令汇总

表 9-1 列出了本章涉及到的主要的命令。

表 9-1 本章命令汇总

命令	作用
show ip access-lists	查看所定义的 IP 访问控制列表
clear access-list counters	将访问控制列表计数器清零
access-list	定义 ACL
ip access-group	在接口下应用 ACL
access-class	在 vty 下应用 ACL
ip access-list	定义命名的 ACL
time-range time	定义时间范围
username *username* password *password*	建立本地数据库
autocommand	定义自动执行的命令

第 10 章　DHCP

IP 地址已是每台计算机必定配置的参数了，手工设置每一台计算机的 IP 地址成为管理员最不愿意做的一件事，于是自动配置 IP 地址的方法出现了，这就是 DHCP（Dynamic Host Configuration Protocol，动态主机配置协议）。DHCP 服务器能够从预先设置的 IP 地址池里自动给主机分配 IP 地址，它不仅能够保证 IP 地址不重复分配，也能及时回收 IP 地址以提高 IP 地址的利用率。

10.1　DHCP 概述

在动态 IP 地址的方案中，每台计算机并不设定固定的 IP 地址，而是在计算机开机时才被分配一个 IP 地址，这台计算机被称为 DHCP 客户端。而负责给 DHCP 客户端分配 IP 地址的计算机称为 DHCP 服务器。也就是说 DHCP 是采用客户/服务器(Client/Server)模式，有明确的客户端和服务器角色的划分。

DHCP 的工作过程如下：

1．DHCP 客户机启动时，客户机在当前的子网中广播 DHCPDISCOVER 报文向 DHCP 服务器申请一个 IP 地址。

2．DHCP 服务器收到 DHCPDISCOVER 报文后，它将从针对那台主机的地址区间中为它提供一个尚未被分配出去的 IP 地址，并把提供的 IP 地址暂时标记为不可用。服务器以 DHCPOFFER 报文送回给主机。如果网络里包含有不止一个的 DHCP 服务器，则客户机可能收到好几个 DHCPOFFER 报文，客户机通常只承认第一个 DHCPOFFER。

3．客户端收到 DHCPOFFER 后，向服务器发送一个含有有关 DHCP 服务器提供的 IP 地址的 DHCPREQUEST 报文。如果客户端没有收到 DHCPOFFER 报文并且还记得以前的网络配置，此时使用以前的网络配置（如果该配置仍然在有效期限内）。

4．DHCP 服务器向客户机发回一个含有原先被发出的 IP 地址及其分配方案的一个应答报文(DHCPACK)。

5．客户端接受到包含了配置参数的 DHCPACK 报文，利用 ARP 检查网络上是否有相同的 IP 地址。如果检查通过，则客户机接受这个 IP 地址及其参数，如果发现有问题，客户机向服务器发送 DHCPDECLINE 信息，并重新开始新的配置过程。服务器收到 DHCPDECLINE 信息，将该地址标为不可用。

6．DHCP 服务器只能将那个 IP 地址分配给 DHCP 客户一定时间，DHCP 客户必须在该次租用过期前对它进行更新。客户机在 50%租借时间过去以后，每隔一段时间就开始请求 DHCP 服务器更新当前租借，如果 DHCP 服务器应答则租用延期。如果 DHCP 服务器始终没有应答，在有效租借期的 87.5%，客户应该与任何一个其他的 DHCP 服务器通信，并请求更新它的配置信息。如果客户机不能和所有的 DHCP 服务器取得联系，租借时间到后，它必须放弃当前的 IP 地址并重新发送一个 DHCPDISCOVER 报文开始上述的 IP 地址获得过程。

7．客户端可以主动向服务器发出 DHCPRELEASE 报文，将当前的 IP 地址释放。

10.2 实验 1：DHCP 基本配置

1.实验目的

通过本实验可以掌握：

（1）DHCP 的工作原理和工作过程

（2）DHCP 服务器的基本配置和调试

（3）客户端配置

2. 拓扑结构

实验拓扑如图 10-1 所示。

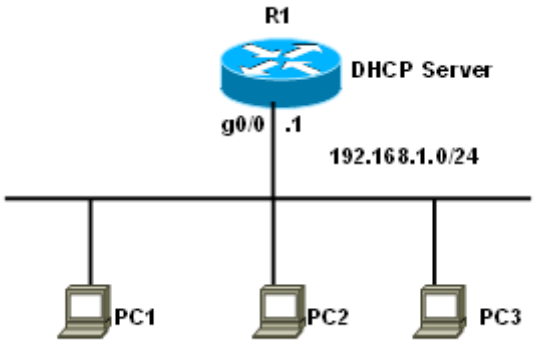

图 10-1 DHCP 基本配置

3. 实验步骤

（1）步骤 1：配置路由器 R1 提供 DHCP 服务

```
R1(config)#service dhcp                          //开启 DHCP 服务
R1(config)#no ip dhcp conflict logging           //关闭 DHCP 冲突日志
R1(config)#ip dhcp pool ccie                     //定义地址池
R1(dhcp-config)#network 192.168.1.0 /24          //DHCP 服务器要分配的网络和掩码
R1(dhcp-config)#domain-name cisco.com            //域名
R1(dhcp-config)#default-router 192.168.1.1
 //默认网关，这个地址要和相应网络所连接的路由器的以太口地址相同
R1(dhcp-config)#netbios-name-server 192.168.1.2     //WINS 服务器
R1(dhcp-config)#dns-server 192.168.1.4              //DNS 服务器
R1(dhcp-config)#option 150 ip 192.168.1.3           //TFTP 服务器
R1(dhcp-config)#lease infinite                      //定义租期
R1(config)#ip dhcp excluded-address 192.168.1.1 192.168.1.5 //排除的地址段
```

（2）步骤 2：设置 windows 客户端

首先在 Windows 下把 TCP/IP 地址设置为自动获得（如图 10-2 所示），如果 DHCP 服务器还提供 DNS、WINS 等，也把它们设置为自动获得。

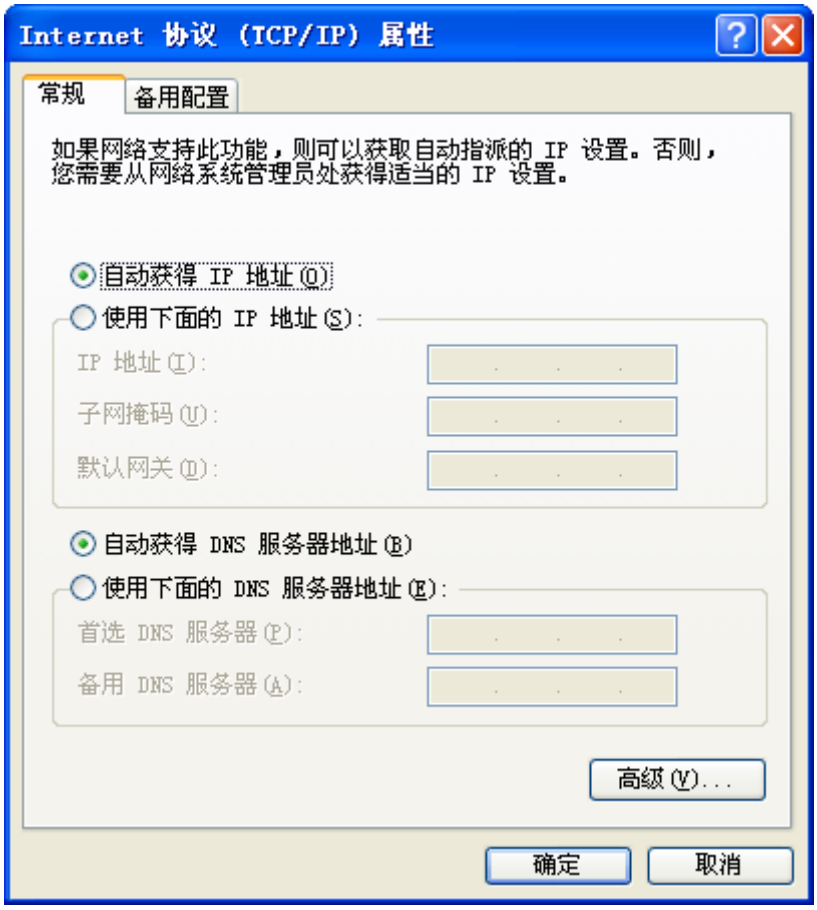

图 10-2 修改 TCP/IP 属性

4. 实验调试

（1）在客户端测试

在“命令提示符”下，执行 C:/>ipconfig/renew 可以更新 IP 地址。而执行 C:/>ipconfig/all 可以看到 IP 地址、WINS、DNS、域名是否正确。要释放地址用 C:/>ipconfig/release 命令。

```
C:\>ipconfig/renew
Windows IP Configuration
Ethernet adapter 本地连接:

        Connection-specific DNS Suffix  . : cisco.com
        IP Address. . . . . . . . . . . . : 192.168.1.7
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.1.1

C:\>ipconfig/all
Windows IP Configuration
Ethernet adapter 本地连接:

        Connection-specific DNS Suffix  . : cisco.com
        Description . . . . . . . . . . . : Realtek RTL8139/810x Family Fast Eth
```

```
ernet NIC
        Physical Address. . . . . . . . . : 00-60-67-00-DD-5B
        Dhcp Enabled. . . . . . . . . . . : Yes
        Autoconfiguration Enabled . . . . : Yes
        IP Address. . . . . . . . . . . . : 192.168.1.7
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.1.1
        DHCP Server . . . . . . . . . . . : 192.168.1.1
        DNS Servers . . . . . . . . . . . : 192.168.1.4
        Primary WINS Server . . . . . . . : 192.168.1.2
        Lease Obtained. . . . . . . . . . : 2007 年 2 月 22 日 13:01:01
        Lease Expires . . . . . . . . . . : 2038 年 1 月 19 日 11:14:07
```

（2）show ip dhcp pool

该命令用来查看 DHCP 地址池的信息。

```
R1#show ip dhcp pool
Pool ccie :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254             //地址池中共计 254 个地址
 Leased addresses               : 2               //已经分配出去 2 个地址
 Pending event                  : none
 1 subnet is currently in the pool :
 Current index        IP address range                    Leased addresses
 192.168.1.8          192.168.1.1      - 192.168.1.254     2
//下一个将要分配的地址、地址池的范围以及分配出去的地址的个数
```

（3）show ip dhcp binding

该命令用来查看 DHCP 的地址绑定情况。

```
R1#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address       Client-ID/          Lease expiration        Type
                 Hardware address/
                 User name
192.168.1.6      0063.6973.636f.2d   Infinite                Automatic
192.168.1.7      0100.6067.00dd.5b   Infinite                Automatic
```

以上输出表明 DHCP 服务器自动分配给客户端的 IP 地址以及所对应的客户端的硬件地址。

10.3 实验 2：DHCP 中继

1.实验目的

通过本实验可以掌握通过 DHCP 中继实现跨网络的 DHCP 服务。

2.拓扑结构

实验拓扑如图 10-3 所示。

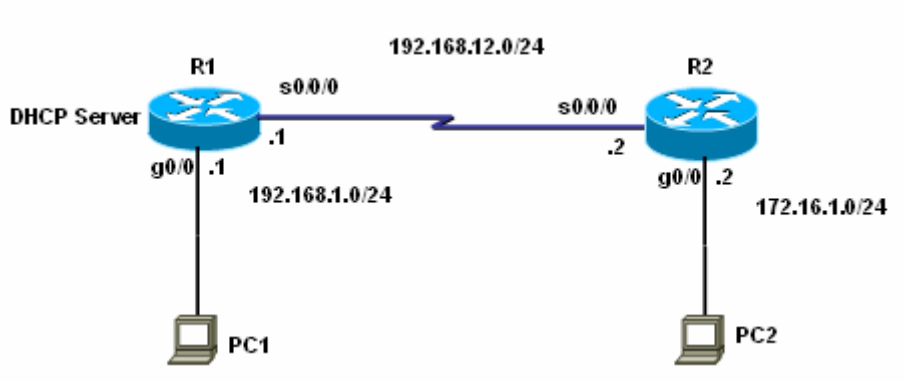

图 10-3 DHCP 中继配置

本实验中，R1 仍然担任 DHCP 服务器的角色，负责向 PC1 所在网络和 PC2 所在网络的主机动态分配 IP 地址，所以 R1 上需要定义两个地址池。整个网络运行 RIPv2 协议，确保网络 IP 连通性。

3. 实验步骤

（1）步骤 1：配置路由器 R1 提供 DHCP 服务

```
R1(config)#interface gigabitEthernet0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#no shutdown
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 192.168.1.0
R1(config-router)#network 192.168.12.0
R1(config)#service dhcp
R1(config)#no ip dhcp conflict logging
R1(config)#ip dhcp pool ccie     //定义第一个地址池
R1(dhcp-config)#network 192.168.1.0 /24
R1(dhcp-config)#default-router 192.168.1.1
R1(dhcp-config)#domain-name cisco.com
R1(dhcp-config)#netbios-name-server 192.168.1.2
R1(dhcp-config)#dns-server 192.168.1.4
R1(dhcp-config)#option 150 ip 192.168.1.3
R1(dhcp-config)#lease infinite
R1(config)#ip dhcp excluded-address 192.168.1.1 192.168.1.5
R1(config)#ip dhcp pool ccnp   //定义第二个地址池
R1(dhcp-config)#network 172.16.1.0 255.255.255.0
R1(dhcp-config)#domain-name szpt.net
R1(dhcp-config)#default-router 172.16.1.2
R1(dhcp-config)#netbios-name-server 192.168.1.2
R1(dhcp-config)#dns-server 192.168.1.4
R1(dhcp-config)#option 150 ip 192.168.1.3
R1(dhcp-config)#lease infinite
R1(config)#ip dhcp excluded-address 172.16.1.1 172.16.1.2
```

（2）步骤 2：配置路由器 R2

```
R2(config)#interface gigabitEthernet0/0
R2(config-if)#ip address 172.16.1.2 255.255.255.0
R2(config-if)#ip helper-address 192.168.12.1    //配置帮助地址
R2(config-if)#no shutdown
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.12.0
R2(config-router)#network 172.16.0.0
```

【技术要点】

路由器是不能转发“255.255.255.255”的广播，但是很多服务（如 DHCP、TFTP 等）的客户端请求都是以泛洪广播的方式发起的，我们不可能将每个网段都放置这样的服务器，因此使用 Cisco IOS 帮助地址特性是很好的选择。通过使用帮助地址，路由器可以被配置为接受对 UDP 服务的广播请求，然后将之以单点传送的方式发给某个具体的 IP 地址，或者以定向广播的形式向某个网段转发这些请求，这就是中继。

4.实验调试

（1）show ip dhcp binding

在 PC1 和 PC2 上自动获取 IP 地址后，在 R1 上执行：

```
R1#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address       Client-ID/          Lease expiration        Type
                 Hardware address/
                 User name
172.16.1.3       0100.6067.00dd.5b   Infinite                Automatic
192.168.1.6      0063.6973.636f.2d   Infinite                Automatic
192.168.1.7      0100.6067.00ef.31   Infinite                Automatic
```

以上输出表明两个地址池都为相应的网络上的主机分配了 IP 地址。

（2）show ip dhcp pool

```
R1#show ip dhcp pool
Pool ccie :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254
 Leased addresses               : 2
 Pending event                  : none
 1 subnet is currently in the pool :
 Current index        IP address range                    Leased addresses
 192.168.1.8          192.168.1.1      - 192.168.1.254     2

Pool ccnp :
 Utilization mark (high/low)    : 100 / 0
```

```
Subnet size (first/next)        : 0 / 0
Total addresses                 : 254
Leased addresses                : 1
Pending event                   : none
1 subnet is currently in the pool :
Current index        IP address range                    Leased addresses
172.16.1.4           172.16.1.1       - 172.16.1.254      1
```

（3）**debug ip dhcp server events**

在 PC2 上先执行“ipconfig/release”,再执行“ipconfig/renew”，显示如下：

```
R1#debug ip dhcp server events
R1#clear ip dhcp binding *
*Feb 22 05:50:24.475: DHCPD: Sending notification of DISCOVER:
*Feb 22 05:50:24.475: DHCPD: htype 1 chaddr 0060.6700.dd5b
*Feb 22 05:50:24.475: DHCPD: circuit id 00000000
*Feb 22 05:50:24.475: DHCPD: Seeing if there is an internally specified pool class:
*Feb 22 05:50:24.475: DHCPD: htype 1 chaddr 0060.6700.dd5b
*Feb 22 05:50:24.475: DHCPD: circuit id 00000000
*Feb 22 05:50:26.475: DHCPD: client requests 172.16.1.4.
*Feb 22 05:50:26.475: DHCPD: Adding binding to radix tree (172.16.1.4)
*Feb 22 05:50:26.475: DHCPD: Adding binding to hash tree
*Feb 22 05:50:26.475: DHCPD: assigned IP address 172.16.1.4 to client 0100.6067.00dd.5b.
*Feb 22 05:50:26.519: DHCPD: Sending notification of ASSIGNMENT:
*Feb 22 05:50:26.519:  DHCPD: address 172.16.1.4 mask 255.255.255.0
*Feb 22 05:50:26.519:   DHCPD: htype 1 chaddr 0060.6700.dd5b
*Feb 22 05:50:26.519:   DHCPD: lease time remaining (secs) = 4294967295
```

以上输出显示了 DHCP 动态分配 IP 地址的基本过程。

（4）**show ip interface**

```
R2#show ip interface gigabitEthernet0/0
GigabitEthernet0/0 is up, line protocol is up
  Internet address is 172.16.1.2/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is 192.168.12.1
  ……
```

以上输出看到 gigabitEthernet0/0 接口使用了帮助地址 192.168.12.1。

10.4 DHCP 命令汇总

表 10-1 列出了本章涉及到的主要的命令。

表 10-1 本章命令汇总

命令	作用
show ip dhcp pool	查看 DHCP 地址池的信息
show ip dhcp binding	查看 DHCP 的地址绑定情况

show ip dhcp database	查看 DHCP 数据库
show ip interface	查看接口信息
debug ip dhcp server events	动态查看 DHCP 服务器的事件
service dhcp	开启 DHCP 服务
no ip dhcp conflict logging	关闭 DHCP 冲突日志
ip dhcp pool	配置 DHCP 分配的地址池
network	DHCP 服务器要分配的网络和掩码
default-router	默认网关
domain-name	域名
netbios-name-server	WINS 服务器
dns-server	域名服务器
option 150 ip	FTP 服务器
lease	配置租期
ip dhcp excluded-address	排除地址段
ip helper-address	配置 DHCP 中继的地址

第 11 章　NAT

Internet 技术的飞速发展，使越来越多的用户加入到互联网，因此 IP 地址短缺已成为一个十分突出的问题。NAT(Network Address Translation，网络地址翻译)是解决 IP 地址短缺的重要手段。

11.1　NAT 概述

NAT 是一个 IETF 标准，允许一个机构以一个地址出现在 Internet 上。NAT 技术使得一个私有网络可以通过 Internet 注册 IP 连接到外部世界，位于 Inside 网络和 Outside 网络中的 NAT 路由器在发送数据包之前，负责把内部 IP 地址翻译成外部合法 IP 地址。NAT 将每个局域网节点的 IP 地址转换成一个合法 IP 地址，反之亦然。它也可以应用到防火墙技术里，把个别 IP 地址隐藏起来不被外界发现，对内部网络设备起到保护的作用，同时，它还帮助网络可以超越地址的限制，合理地安排网络中的公有 Internet 地址和私有 IP 地址的使用。

NAT 有三种类型：静态 NAT、动态 NAT 和端口地址转换（PAT）。

1．静态 NAT

静态 NAT 中，内部网络中的每个主机都被永久映射成外部网络中的某个合法的地址。静态地址转换将内部本地地址与内部合法地址进行一对一的转换，且需要指定和哪个合法地址进行转换。如果内部网络有 E-mail 服务器或 FTP 服务器等可以为外部用户提供的服务，这些服务器的 IP 地址必须采用静态地址转换，以便外部用户可以使用这些服务。

2．动态 NAT

动态 NAT 首先要定义合法地址池，然后采用动态分配的方法映射到内部网络。动态 NAT 是动态一对一的映射。

3．PAT

PAT 则是把内部地址映射到外部网络的 IP 地址的不同端口上，从而可以实现多对一的映射。PAT 对于节省 IP 地址是最为有效的。

11.2 实验 1：静态 NAT 配置

1.实验目的

通过本实验可以掌握

（1）静态 NAT 的特征

（2）静态 NAT 基本配置和调试

2.拓扑结构

实验拓扑如图 11-1 所示。

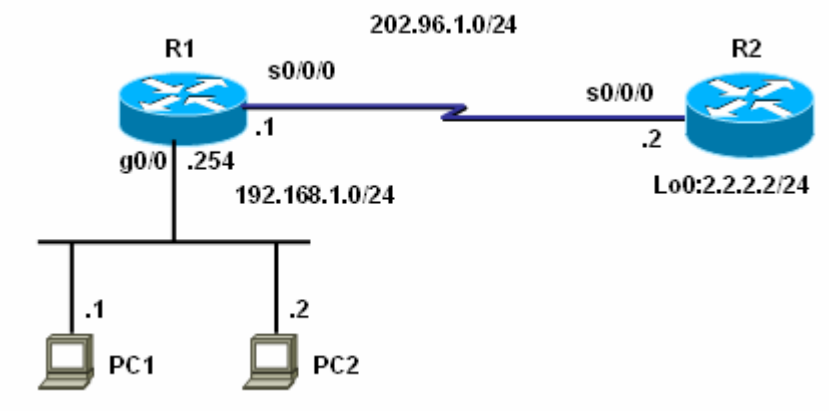

图 11-1 静态 NAT 配置

3.实验步骤

（1）步骤 1：配置路由器 R1 提供 NAT 服务

```
R1(config)#ip nat inside source static 192.168.1.1 202.96.1.3
//配置静态 NAT 映射
R1(config)#ip nat inside source static 192.168.1.2 202.96.1.4
R1(config)#interface g0/0
R1(config-if)#ip nat inside
//配置 NAT 内部接口
R1(config)#interface s0/0/0
R1(config-if)#ip nat outside
//配置 NAT 外部接口
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 202.96.1.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 202.96.1.0
R2(config-router)#network 2.0.0.0
```

4.实验调试

（1）debug ip nat

该命令可以查看地址翻译的过程。

在 PC1 和 PC2 上 Ping 2.2.2.2（路由器 R2 的环回接口），此时应该是通的，路由器 R1 的输出信息如下：

```
R1#debug ip nat
*Mar  4 02:02:12.779: NAT*: s=192.168.1.1->202.96.1.3, d=2.2.2.2 [20240]
*Mar  4 02:02:12.791: NAT*: s=2.2.2.2, d=202.96.1.3->192.168.1.1 [14435]
......
*Mar  4 02:02:25.563: NAT*: s=192.168.1.2->202.96.1.4, d=2.2.2.2 [25]
*Mar  4 02:02:25.579: NAT*: s=2.2.2.2, d=202.96.1.4->192.168.1.2 [25]
......
```

以上输出表明了 NAT 的转换过程。首先把私有地址“192.168.1.1”和“192.168.1.2”分别转换成公网地址“202.96.1.3”和“202.96.1.4”访问地址“2.2.2.2”，然后回来的时候把公网地址“202.96.1.3”和 “202.96.1.4”分别转换成私有地址“192.168.1.1”和“192.168.1.2”。

（2）show ip nat translations

该命令用来查看 NAT 表。静态映射时，NAT 表一直存在。

```
R1#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
--- 202.96.1.3         192.168.1.1        ---                ---
```

--- 202.96.1.4 192.168.1.2 --- ---

以上输出表明了内部全局地址和内部局部地址的对应关系。

【术语】

① 内部局部（inside local）地址：在内部网络使用的地址，往往是 RFC1918 地址；

② 内部全局（inside global）地址：用来代替一个或多个本地 IP 地址的、对外的、向 NIC 注册过的地址；

③ 外部局部（outside local）地址：一个外部主机相对于内部网络所用的 IP 地址。不一定是合法的地址；

④ 外部全局（outside global）地址：外部网络主机的合法 IP 地址。

11.3 实验 2：动态 NAT

1.实验目的

通过本实验可以掌握:

（1）动态 NAT 的特征

（2）动态 NAT 配置和调试

2.拓扑结构

实验拓扑如图 11-1 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1 提供 NAT 服务

```
R1(config)#ip nat pool NAT 202.96.1.3 202.96.1.100 netmask 255.255.255.0
//配置动态 NAT 转换的地址池
R1(config)#ip nat inside source list 1 pool NAT
//配置动态 NAT 映射
R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
//允许动态 NAT 转换的内部地址范围
R1(config)#interface g0/0
R1(config-if)#ip nat inside
R1(config-if)#interface s0/0/0
R1(config-if)#ip nat outside
```

4.实验调试

在 PC1 上访问 2.2.2.2（路由器 R2 的环回接口）的 WWW 服务，在 PC2 上分别 telnet 和 ping 2.2.2.2（路由器 R2 的环回接口），调试结果如下：

（1）debug ip nat

```
R1#debug ip nat
IP NAT debugging is on
R1#clear ip nat translation * //清除动态 NAT 表
*Mar  4 01:34:23.075: NAT*: s=192.168.1.1->202.96.1.4, d=2.2.2.2 [19833]
*Mar  4 01:34:23.087: NAT*: s=2.2.2.2, d=202.96.1.4->192.168.1.1 [62333]
......
*Mar  4 01:28:49.867: NAT*: s=192.168.1.2->202.96.1.3, d=2.2.2.2 [62864]
*Mar  4 01:28:49.875: NAT*: s=2.2.2.2, d=202.96.1.3->192.168.1.2 [54062]
......
```

【提示】

如果动态 NAT 地址池中没有足够的地址作动态映射，则会出现类似下面的信息，提示 NAT 转换失败，并丢弃数据包。

*Feb 22 09:02:59.075: NAT: translation failed (A), dropping packet s=192.168.1.2 d=2.2.2.2

（2）show ip nat translations

```
R1#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
tcp 202.96.1.4:1721    192.168.1.1:1721   2.2.2.2:80         2.2.2.2:80
--- 202.96.1.4         192.168.1.1        ---                ---
icmp 202.96.1.3:3      192.168.1.2:3      2.2.2.2:3          2.2.2.2:3
tcp 202.96.1.3:14347   192.168.1.2:14347  2.2.2.2:23         2.2.2.2:23
--- 202.96.1.3         192.168.1.2        ---                ---
```

以上信息表明当 PC1 和 PC2 第一次访问“2.2.2.2”地址的时候,NAT 路由器 R1 为主机 PC1 和 PC2 动态分配两个全局地址“202.96.1.4”和“202.96.1.3”，在 NAT 表表中生成两条动态映射的记录，同时会在 NAT 表中生成和应用向对应的协议和端口号的记录（过期时间为 60 秒）。在动态映射没有过期（过期时间为 86400 秒）之前，再有应用从相同主机发起时，NAT 路由器直接查 NAT 表,然后为应用分配相应的端口号。

（3）show ip nat statistics

该命令用来查看 NAT 转换的统计信息。

```
R1#show ip nat statistics
Total active translations: 5 (0 static, 5 dynamic; 3 extended)
//有 5 个转换是动态转化，
Outside interfaces:
  Serial0/0/0
//NAT 外部接口
Inside interfaces:
  GigabitEthernet0/0
//NAT 内部接口
Hits: 54  Misses: 6
CEF Translated packets: 60, CEF Punted packets: 5
Expired translations: 12   //NAT 表中过期的转换
Dynamic mappings:          //动态映射
-- Inside Source
[Id: 1] access-list 1 pool NAT refcount 2
 pool NAT: netmask 255.255.255.0  //地址池名字和掩码
        start 202.96.1.3 end 202.96.1.100 //地址池范围
        type generic, total addresses 98, allocated 2 (2%), misses 0
//共 98 个地址，分出去 2 个
Queued Packets: 0
```

11.4 实验 3： PAT 配置

1.实验目的

通过本实验可以掌握:

（1）PAT 的特征

（2）overload 的使用

（3）PAT 配置和调试

2.拓扑结构

实验拓扑如图 11-1 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1 提供 NAT 服务

```
R1(config)#ip nat pool NAT 202.96.1.3 202.96.1.100 netmask 255.255.255.0
R1(config)#ip nat inside source list 1 pool NAT overload //配置 PAT
R1(config)#access-list 1 permit 192.168.1.0 0.0.0.255
R1(config)#interface g0/0
R1(config-if)#ip nat inside
R1(config-if)#interface s0/0
R1(config-if)#ip nat outside
```

4.实验调试

在 PC1 上访问 2.2.2.2（路由器 R2 的环回接口）的 WWW 服务，在 PC2 上分别 telnet 和 ping 2.2.2.2（路由器 R2 的环回接口），调试结果如下：

（1）debug ip nat

```
*Mar  4 01:53:47.983: NAT*: s=192.168.1.1->202.96.1.3, d=2.2.2.2 [20056]
*Mar  4 01:53:47.995: NAT*: s=2.2.2.2, d=202.96.1.3->192.168.1.1 [46201]
......
*Mar  4 01:54:03.015: NAT*: s=192.168.1.2->202.96.1.3, d=2.2.2.2 [20787]
*Mar  4 01:54:03.219: NAT*: s=2.2.2.2, d=202.96.1.3->192.168.1.2 [12049]
......
```

（2）show ip nat translations

```
R1#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
tcp 202.96.1.3:1732    192.168.1.1:1732   2.2.2.2:80         2.2.2.2:80
icmp 202.96.1.3:4      192.168.1.2:4      2.2.2.2:4          2.2.2.2:4
tcp 202.96.1.3:12320   192.168.1.2:12320  2.2.2.2:23         2.2.2.2:23
```

以上输出表明进行 PAT 转换使用的是同一个 IP 地址的不同端口号。

（3）show ip nat statistics

```
Total active translations: 3 (0 static, 3 dynamic; 3 extended)
Outside interfaces:
  Serial0/0/0
Inside interfaces:
  GigabitEthernet0/0
Hits: 762  Misses: 22
CEF Translated packets: 760, CEF Punted packets: 47
Expired translations: 19
Dynamic mappings:
-- Inside Source
[Id: 2] access-list 1 pool NAT refcount 3
 pool NAT: netmask 255.255.255.0
        start 202.96.1.3 end 202.96.1.100
```

```
        type generic, total addresses 98, allocated 1 (1%), misses 0
Queued Packets: 0
```

【提示】

动态 NAT 的过期时间是 86400 秒，PAT 的过期时间是 60 秒，通过命令“**show ip nat translations verbose**”可以查看。也可以通过下面的命令来修改超时时间：

R1(config)**#ip nat translation timeout** *timeout*

参数 timeout 的范围是 0-2147483。

如果主机的数量不是很多，可以直接使用 outside 接口地址配置 PAT，不必定义地址池，命令如下：

R1(config)**#ip nat inside source list 1 interface s0/0/0 overload**

11.5 NAT 命令汇总

表 11-1 列出了本章涉及到的主要的命令。

表 11-1 本章命令汇总

命令	**作用**
clear ip nat translation *	清除动态 NAT 表
show ip nat translation	查看 NAT 表
show ip nat statistics	查看 NAT 转换的统计信息
debug ip nat	动态查看 NAT 转换过程
ip nat inside source static	配置静态 NAT
ip nat inside	配置 NAT 内部接口
ip nat outside	配置 NAT 外部接口
ip nat pool	配置动态 NAT 地址池
ip nat inside source list *access-list-number* pool *name*	配置动态 NAT
ip nat inside source list *access-list-number* pool *name* overload	配置 PAT

第 12 章 交换机基本配置

交换机是局域网中最重要的设备，交换机是基于 MAC 来进行工作的。和路由器类似，交换机也有 IOS，IOS 的基本使用方法是一样的。本章将简单介绍交换机的一些基本配置，以及交换机独特的密码恢复、IOS 恢复步骤。关于 VLAN、Trunk 等将在后面章节介绍。

12.1 交换机简介

交换机是第二层的设备，可以隔离冲突域。交换机是基于收到的数据帧中的源 MAC 地址和目的 MAC 地址来进行工作。交换机的作用主要有这么两个：一个是维护 CAM（Context Address Memory）表，该表是 MAC 地址和交换机端口的映射表；另一个是根据 CAM 来进行数据帧的转发。交换机对帧的处理有三种：交换机收到帧后，查询 CAM 表，如果能查询到目的计算机所在的端口，并且目的计算机所在的端口不是交换机接收帧的源端口，交换机将把帧从这一端口转发出去（Forward）；如果该计算机所在的端口和交换机接收帧的源端口是同一端口，交换机将过滤掉该帧（Filter）；如果交换机不能查询到目的计算机所在的端口，交换机将把帧从源端口以外的其他所有端口上发送出去，这称为泛洪（Flood），当交换机接收到的是帧是广播帧或者多播帧，交换机也会泛洪帧。

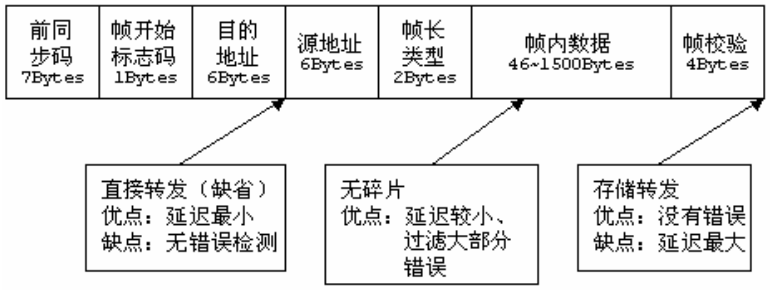

图 12-1　三种交换方式的比较

以太网交换机转发数据帧有三种交换方式，如图12-1：

（1） 存储转发（Store-and-Forward）

存储转发方式是先存储后转发的方式。它把从端口输入的数据帧先全部接收并存储起来；然后进行CRC（循环冗余码校验）检查，把错误帧丢弃；最后才取出数据帧目的地址，查找地址表后进行过滤和转发。存储转发方式延迟大；但是它可以对进入交换机的数据包进行高级别的错误检测。这种方式可以支持不同速度的端口间的转发。

（2） 直接转发（Cut-Through）

交换机在输入端口检测到一个数据帧时，检查该帧的帧头，只要获取了帧的目的地址，就开始转发帧。它的优点是：开始转发前不需要读取整个完整的帧，延迟非常小。它的缺点是：不能提供错误检测能力。

（3） 无碎片（Fragment-Free）

这是改进后的直接转发，是介于前两者之间的一种解决方法。无碎片方法在读取数据帧的长前64个字节后，就开始转发该帧。这种方式虽然也不提供数据校验，但是能够避免大多数的错误。它的数据处理速度比直接转发方式慢，但比存储转发方式快许多。

CISCO 交换机和路由器一样，本质上也是一台特殊的计算机，也有 CPU、RAM 等部件。也采用 IOS，所以交换机的很多基本配置（例如密码、主机名等）和路由器是类似的。

12.2 实验 1:交换机基本配置

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 熟悉交换机的基本配置

2. 实验拓扑

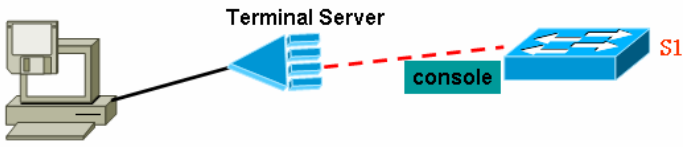

图 12-2 实验 1 拓扑图

3. 实验步骤

(1) 步骤 1：配置主机名

```
Switch>enable
Switch#conf terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#hostname S1
```

(2) 步骤 2：配置密码

```
S1(config)#enable secret cisco
S1(config)#line vty 0 15
S1(config-line)#password cisco
S1(config-line)#login
```

(3) 步骤 3：接口基本配置

默认时交换机的以太网接口是开启的。对于交换机的以太网口可以配置其双工模式、速率等。

```
S1(config)#interface f0/1
switch(config-if)#duplex { full | half | auto }
```

//duplex 用来配置接口的双工模式，full——全双工、half——半双工、auto——自动检测双工的模式

```
switch(config-if)#speed { 10 | 100 | 1000 | auto }
```

//speed 命令用来配置交换机的接口速度，10——10M、100——100M、1000——1000M、auto——自动检测接口速度。

(4) 配置管理地址

交换机也允许被 telnet，这时需要在交换机上配置一个 IP 地址,这个地址是在 VLAN 接口上配置的。如下：

```
S1(config)#int vlan 1
S1(config-if)#ip address 172.16.0.1 255.255.0.0
S1(config-if)#no shutdown
S1(config)#ip default-gateway 172.16.0.254
```

//以上在 VLAN 1 接口上配置了管理地址，接在 VLAN 1 上的计算机可以直接进行 telnet 该地址。为了其他网段的计算机也可以 telnet 交换机，我们在交换机上配置了缺省网关。

(5)　保存配置

```
S1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

12.3 实验 2:交换机端口安全

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1）　理解交换机的 MAC 表

（2）　理解交换机的端口安全

（3）　配置交换机的端口安全特性

2. 实验拓扑

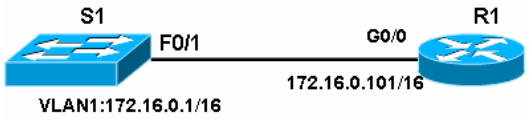

图 12-3　实验 2 拓扑图

3. 实验步骤

交换机端口安全特性，可以让我们配置交换机端口，使得非法的 MAC 地址的设备接入时，交换机自动关闭接口或者拒绝非法设备接入，也可以限制某个端口上最大的 MAC 地址数。我们这里限制 f0/1 接口只允许 R1 接入。

(1)　步骤 1：检查 R1 的 g0/0 接口的 MAC 地址

```
R1(config)#int g0/0
R1(config-if)#no shutdown
R1(config-if)#ip address 172.16.0.101 255.255.0.0

R1#show int g0/0
GigabitEthernet0/0 is up, line protocol is up
  Hardware is MV96340 Ethernet, address is 0019.5535.b828 (bia 0019.5535.b828)
  //这里可以看到 g0/0 接口的 MAC 地址，记下它
  Internet address is 172.16.0.101/16
  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,
（此处省略）
```

(2)　步骤 2：配置交换机端口安全

```
S1(config)#int f0/1
S1(config-if)#shutdown
S1(config-if)#switch mode access
//以上命令把端口改为访问模式，即用来接入计算机，在下一章详细介绍该命令的含义。
S1(config-if)#switch port-securitiy
//以上命令是打开交换机的端口安全功能。
```

S1(config-if)#switch port-securitiy maximum 1

//以上命令只允许该端口下的MAC条目最大数量为1，即只允许一个设备接入

S1(config-if)#switch port-securitiy violation { protect | shutdown | restrict }

- protect:当新的计算机接入时，如果该接口的MAC条目超过最大数量，则这个新的计算机将无法接入，而原有的计算机不受影响
- shutdown:当新的计算机接入时，如果该接口的MAC条目超过最大数量，则该接口将会被关闭,则这个新的计算机和原有的计算机都无法接入,需要管理员使用“no shutdown”命令重新打开。
- restrict:当新的计算机接入时，如果该接口的MAC条目超过最大数量，则这个新的计算机可以接入，然而交换机将向发送警告信息。

S1(config-if)#switchport port-security mac-address 0019.5535.b828

//允许R1路由器从f0/1接口接入

S1(config-if)#no shutdown

S1(config)#int vlan1

S1(config-if)#no shutdown

S1(config-if)#ip address 172.16.0.1 255.255.0.0

//以上配置交换机的管理地址

(3)　步骤3：检查MAC地址表

S1#show mac-address-table

```
           Mac Address Table
-------------------------------------------
（此处省略）
Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
 All    0100.0ccc.cccc    STATIC      CPU
   1    0018.ba11.eb91    DYNAMIC     Fa0/15
   1    0019.5535.b828    STATIC      Fa0/1
Total Mac Addresses for this criterion: 24
```

//R1的MAC已经被登记在f0/1接口，并且表明是静态加入的

(4)　步骤4：模拟非法接入

这时从R1 ping交换机的管理地址，可以ping通,如下：

R1#ping 172.16.0.1

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

在R1上修改g0/0的MAC地址为另一个地址，模拟是另外一台设备接入。如下：

R1(config)#int g0/0

R1(config-if)#mac-address 12.12.12

几秒钟后，则在S1上，出现：

```
01:09:39: %PM-4-ERR_DISABLE: psecure-violation error detected on Fa0/1, putting Fa0/1 in
err-disable state
```

01:09:39: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred, caused by MAC address 0012.0012.0012 on port FastEthernet0/1.

01:09:40: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

//以上提示 f0/1 接口被关闭

S1#show int f0/1

FastEthernet0/1 is down, **line protocol is down (err-disabled)**

  Hardware is Fast Ethernet, address is 0018.ba11.f503 (bia 0018.ba11.f503)

  MTU 1500 bytes, BW 100000 Kbit, DLY 100 usec,

     reliability 255/255, txload 1/255, rxload 1/255

//以上表明 f0/1 接口因为错误而被关闭。非法设备移除后，在 f0/1 接口下，执行“**shutdown**”和“**no shutdown**”命令可以重新打开该接口。

4. 实验调试

S1#show port-security

Secure Port	MaxSecureAddr (Count)	CurrentAddr (Count)	SecurityViolation (Count)	Security Action
Fa0/1	1	1	0	Shutdown

Total Addresses in System (excluding one mac per port)   : 0

Max Addresses limit in System (excluding one mac per port) : 6272

//以上可以查看端口安全的设置情况

12.4 实验 3:交换机的密码恢复

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 交换机的密码恢复步骤

2. 实验拓扑

如图 12-2。

3. 实验步骤

CISCO 交换机的密码恢复步骤和路由器的密码恢复方法差别较大，并且不同型号的交换机恢复方法也有所差异，以下是 Catalyst 3560（Catalyst 2950 也类似）交换机的密码恢复步骤。

（1） 拔掉交换机电源，按住交换机前面板的 Mode 键不放，接上电源，你会看到如下提示：

Base ethernet MAC Address: 00:18:ba:11:f5:00

Xmodem file system is available.

The password-recovery mechanism is enabled.

The system has been interrupted prior to initializing the

flash filesystem. The following commands will initialize
the flash filesystem, and finish loading the operating
system software:

flash_init
load_helper
boot

（2） 输入 **flash_init** 命令

```
Initializing Flash...
flashfs[0]: 3 files, 1 directories
flashfs[0]: 0 orphaned files, 0 orphaned directories
flashfs[0]: Total bytes: 32514048
flashfs[0]: Bytes used: 6076928
flashfs[0]: Bytes available: 26437120
flashfs[0]: flashfs fsck took 12 seconds.
...done Initializing Flash.
Boot Sector Filesystem (bs) installed, fsid: 3
Setting console baud rate to 9600...
```

（3） 输入 **load helper** 命令

（4） 输入 **dir flash:**

```
Directory of flash:/
2    -rwx  6073600   <date>              c3560-ipbasek9-mz.122-25.SEB4.bin
3    -rwx  1455      <date>              config.text
5    -rwx  24        <date>              private-config.text
26437120 bytes available (6076928 bytes used)
```

//config.text 就是交换机的启动配置文件，和路由器的 startup-config 类似

（5） 输入 **rename flash:config.text flash:config.old** 命令

//以上是把启动配置文件改名，这样交换机启动时就读不到 config.text 了，从而没有了密码。

（6） 输入 **boot** 命令引导系统，这时就不要再按住 mode 键了。

（7） 当出现如下提示时，输入 N：

Continue with the configuration dialog? [yes/no] : n

（8） 用 **enable** 命令进入 enable 状态，并将文件 config.old 改回 config.text，命令如下：

rename flash:config.old flash:config.text

（9） 将原配置装入内存，命令如下：

Switch# **copy flash:config.text system:running-config**

（10） 修改各个密码：

```
S1#conf t
S1(config)#enable secret cisco
S1(config)#exit
```

（11） 将配置写入 nvram

```
S1#copy running-config start-config
```

12.5 实验 4：交换机的 IOS 恢复

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 交换机的 IOS 恢复

2. 实验拓扑

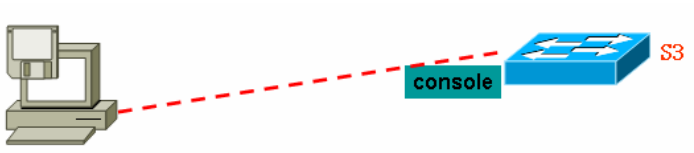

图 12-4 实验 4 拓扑图

3. 实验步骤

交换机如果已经正常开机，则 IOS 可以从 TFTP 服务器上恢复，具体步骤请参见路由器的 IOS 恢复步骤。然而如果交换机无法正常开机，IOS 的恢复要使用 XModem 方式，该方式是通过 concole 口从计算机下载 IOS，速度为 9600bps，因此速度很慢。步骤如下：

(1) 把计算机的串口和交换机的 console 口连接好，用超级终端软件连接上交换机

(2) 交换机开机后，执行以下命令：

switch: **flash_init**

switch: **load_helper**

(3) 输入拷贝指令：

switch:**copy xmodem: flash:c2950-i6q412-mz.121-22.EA5a.bin**

该命令的含义是通过 xmodem 方式拷贝文件，保存在 FLASH，文件名为 c2950-i6q412-mz.121-22.EA5a.bin。出现如下提示：

Begin the Xmodem or Xmodem-1K transfer now...

CCCC

在超级终端窗口中，选择【传送】→【传送文件】菜单，打开图 12-5 窗口，选择 IOS 文件，协议为“Xmodem”。点击“发送”按钮开始发送文件。由于速度很慢，请耐心等待，通信速率为 9600bps。

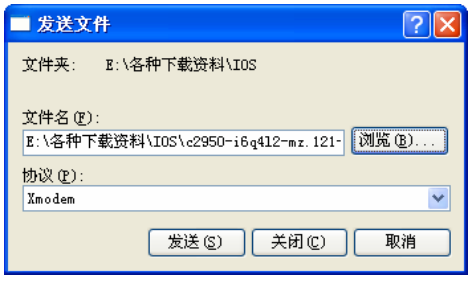

图 12-5 选择 IOS 文件

(4) 传送完毕后执行以下命令：

switch:**boot**

启动系统。

12.6 本章小结

本章简要介绍了交换机的基本配置，交换机的许多配置和路由器很类似。然而交换机的密码恢复和 IOS 恢复方法却和路由器有较大差别。为了减少非法设备的接入，可以在交换机上配置端口安全特性。表 12-1 是本章出现的命令。

表 12-1 本章命令汇总

命令	作用
duplex { full | half | auto }	配置以太口的双工属性
speed { 10 | 100 | 1000 | auto }	配置以太口的速率
ip default-gateway 172.16.0.254	配置缺省网关
switch mode access	把端口改为访问模式
switch port-securitiy	打开交换机的端口安全功能
switch port-securitiy maximum 1	允许该端口下的 MAC 条目最大数量为 1
switch port-securitiy violation { protect | shutdown | restrict }	配置交换机端口安全
switchport port-security mac-address 0019.5535.b828	允许 MAC 为 0019.5535.b828 的设备接入本接口
show mac-address-table	显示 MAC 地址表
mac-address 12.12.12	改变接口的 MAC 地址
rename flash:config.text flash:config.old	把 flash 中的文件改名
copy xmodem: flash:c2950-i6q4l2-mz.121-22.EA5a.bin	通过 Xmodem 模式把文件拷贝到 flash 中
boot	重启交换机

第 13 章 VLAN、TRUNK 和 VTP

Cisco 交换机不仅仅具有 2 层交换功能，它还具有 VLAN 等功能。VLAN 技术可以使我们很容易地控制广播域的大小。有了 VLAN，交换机之间的级联链路就需要 Trunk 技术来保证该链路可以同时传输多个 VLAN 的数据。同时为了方便管理各交换机上的 VLAN 信息，VTP 也被引入了。交换机之间的级联链路带宽如果不够，我们可以把多条链路捆绑起来形成逻辑链路。本章将一一介绍以上各种技术的具体配置。

13.1 VLAN、TRUNK 和 VTP 简介

13.1 VLAN

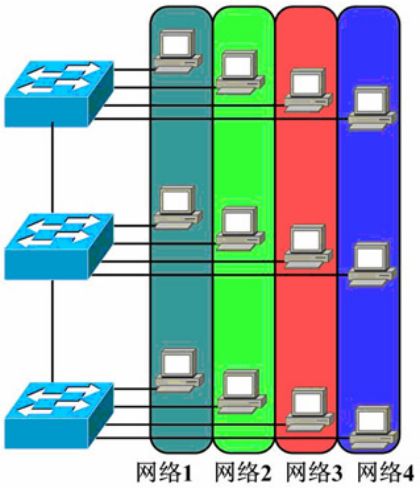

图 13-1 VLAN

如图 13-1，虚拟局域网 VLAN(Virtual LAN)是交换机端口的逻辑组合。VLAN 工作在 OSI 的第 2 层，一个 VLAN 就是一个广播域，VLAN 之间的通信是通过第 3 层的路由器来完成的。VLAN 有以下优点：

(1) 控制网络的广播问题：每一个VLAN是一个广播域，一个VLAN上的广播不会扩散到另一VLAN；

(2) 简化网络管理：当VLAN中的用户位置移动时，网络管理员只需设置几条命令即可；

(3) 提高网络的安全性：VLAN 能控制广播；VLAN 之间不能直接通信。

定义交换机的端口在什么 VLAN 上的常用方法有：

(1) 基于端口的 VLAN ：管理员把交换机某一端口指定为某一 VLAN 的成员；

(2) 基于 MAC 地址的 VLAN ：交换机根据节点的 MAC 地址，决定将其放置于哪个 VLAN 中。

13.2 Trunk

当一个 VLAN 跨过不同的交换机时，在同一 VLAN 上但是却是在不同的交换机上的计算机进行通信时需要使用 Trunk。Trunk 技术使得在一条物理线路上可以传送多个 VLAN 的信息，交换机从属于某一 VLAN（例如 VLAN3）的端口接收到数据，在 Trunk 链路上进行传输前，会

加上一个标记，表明该数据是 VLAN3 的；到了对方交换机，交换机会把该标记去掉，只发送到属于 VLAN3 的端口上。

有两种常见的帧标记技术：ISL 和 802.1Q。ISL 技术在原有的帧上重新加了一个帧头，并重新生成了帧较验序列（FCS），ISL 是思科特有的技术，因此不能在 Cisco 交换机和非 Cisco 交换机之间使用。而 802.1Q 技术在原有帧的源 MAC 地址字段后插入标记字段，同时用新的 FCS 字段替代了原有的 FCS 字段，该技术是国际标准，得到所有厂家的支持。

Cisco 交换机之间的链路是否形成 Trunk 是可以自动协商，这个协议称为 DTP（Dynamic Trunk Protocol），DTP 还可以协商 Trunk 链路的封装类型。表 13-1 是链路两端是否会形成 Trunk 的总结。

表 13-1 DTP 总结

	negotiate	desirable	auto	nonegotiate
negotiate	√	√	√	√
desirable	√	√	√	×
auto	√	√	×	×
nonegotiate	√	×	×	√

13.3 VTP

VTP（VLAN Trunk Protocol）提供了一种用于在交换机上管理 VLAN 的方法，该协议使得我们可以一个或者几个中央点（Server）上创建、修改、删除 VLAN， VLAN 信息通过 Trunk 链路自动扩散到其他交换机，任何参与 VTP 的交换机就可以接受这些修改，所有交换机保持相同的 VLAN 信息。

VTP 被组织成管理域（VTP Domain）,相同域中的交换机能共享 VLAN 信息。根据交换机在 VTP 域中的作用不同，VTP 可以分为三种模式：

（1） 服务器模式(Server)：在 VTP 服务器上能创建、修改、删除 VLAN，同时这些信息会通告给域中的其他交换机。默认情况下，交换机是服务器模式。每个 VTP 域必须至少有一台服务器，域中的 VTP 服务器可以有多台。

（2） 客户机模式(Client)：VTP 客户机上不允许创建、修改、删除 VLAN，但它会监听来自其他交换机的 VTP 通告并更改自己的 VLAN 信息。接收到的 VTP 信息也会在 Trunk 链路上向其他交换机转发，因此这种交换机还能充当 VTP 中继。

（3） 透明模式(Transparent)：这种模式的交换机不参与 VTP。可以在这种模式的交换机上创建、修改、删除 VLAN，但是这些 VLAN 信息并不会通告给其他交换机，它也不接受其他交换机的 VTP 通告而更新自己的 VLAN 信息。然而需要注意的是，它会通过 Trunk 链路转发接收到的 VTP 通告从而充当了 VTP 中继的角色，因此完全可以把该交换机看成是透明的。

VTP 通告是以组播帧的方式发送的，VTP 通告中有一个字段称为修订号（Revision），初始值为 0。只要在 VTP Server 上创建、修改、删除 VLAN，通告的 Revision 就增加 1，通告中还包含了 VLAN 的变化信息。需要注意的是：高 Revision 的通告会覆盖低 Revision 的通告，而不管谁是 Server 还是 Client。交换机只接受比本地保存的 Resivison 号更高的通告；如果交换机收到 Resivison 号更低的通告，会用自己的 VLAN 信息反向覆盖。

13.4 EtherChannel

EtherChannel（以太通道）是由Cisco公司开发的，应用于交换机之间的多链路捆绑技术。它的基本原理是：将两个设备间多条快速以太或千兆以太物理链路捆绑在一起组成一条逻辑链路，从而达到带宽倍增的目的。除了增加带宽外，EtherChannel还可以在多条链路上

均衡分配流量，起到负载分担的作用；在一条或多条链路故障时，只要还有链路正常，流量将转移到其他的链路上，整个过程在几毫秒内完成，从而起到冗余的作用，增强了网络的稳定性和安全性。EtherChannel中，负载在各个链路上的分布可以根据源IP地址、目的IP地址、源MAC地址、目的MAC地址、源IP地址和目的IP地址组合、源MAC地址和目的MAC地址组合等来进行分布。

两台交换机之间是否形成EtherChannel也可以用协议自动协商。目前有两个协商协议：PAGP和LACP，前者是CISCO专有的协议，而LACP是公共的标准。表13-2是PAGP协商的规律总结，表13-3是LACP协商的规律总结。

表13-2 PAGP协商的规律总结

	ON	Desirable	auto
ON	√	×	×
desirable	×	√	√
auto	×	√	×

表13-3 LACP协商的规律总结

	ON	active	passive
ON	√	×	×
active	×	√	√
passive	×	√	×

13.2 实验 1：划分 VLAN

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 熟悉 VLAN 的创建

（2） 把交换机接口划分到特定 VLAN

2. 实验拓扑

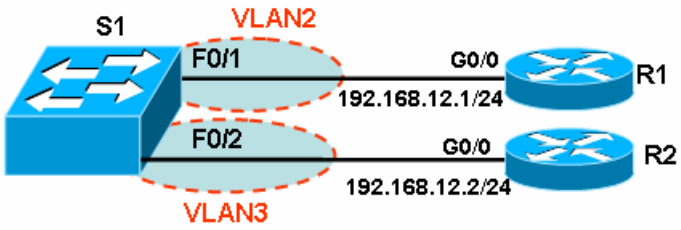

图 13-2 实验 1 拓扑图

3. 实验步骤

要配置 VLAN，首先要先创建 VLAN，然后才把交换机的端口划分到特定的端口上：

(1) 步骤 1：在划分 VLAN 前，配置 R1 和 R2 路由器的 g0/0 接口，从 R1 ping 192.168.12.2。默认时，交换机的全部接口都在 VLAN1 上，R1 和 R2 应该能够通信；

(2) 步骤 2：在 S1 上创建 VLAN

```
S1#vlan database
//进入到 VLAN 配置模式
```

S1(vlan)#vlan 2 name VLAN2
VLAN 2 added:
Name: VLAN2
//以上创建 vlan,2 就是 vlan 的编号，VLAN 号的范围为 1～1001，VLAN2 是该 VLAN 的名字：

S1(vlan)#vlan 3 name VLAN3
VLAN 3 added:
Name: VLAN3
S1(vlan)#exit
APPLY completed.
Exiting....
//退出VLAN模式，创建的VLAN立即生效：

【提示】交换机中的 VLAN 信息存放在单独的文件中 flash:vlan.dat，因此如果要完全清除交换机的配置，除了使用“erase starting-config”命令外，还要使用“delete flash:vlan.dat”命令把 VLAN 数据删除。
【提示】新的 IOS 版本中，可以在全局配置模式中创建 VLAN，如下：
S1(config)#vlan 2
S1(config-vlan)#name VLAN2
S1(config-vlan)#exit
S1(config)#vlan 3
S1(config-vlan)#name VLAN3
（3） 步骤 3：把端口划分在 VLAN 中
S1(config)#interface f0/1
S1(config-if)#switch mode access
//以上把交换机端口的模式改为 access 模式，说明该端口是用于连接计算机的，而不是用于 trunk
S1(config-if)#switch access vlan 2
//然后把该端口 f0/1 划分到 VLAN 2 中

S1(config)#interface f0/2
S1(config-if)#switch mode access
S1(config-if)#switch access vlan 3

【提示】默认时,所有交换机接口都在 VLAN 1 上，VLAN 1 是不能删除的。如果有多个接口需要划分到同一 VLAN 下，也可以采用如下方式以节约时间,注意破折号前后的空格：
S1(config)#interface range f0/2 -3
S1(config-if)#switch mode access
S1(config-if)#switch access vlan 2
【提示】如果要删除VLAN，使用 “no vlan 2”命令即可。删除某一VLAN后，要记得把该VLAN上的端口重新划分到别的VLAN上，否则将导致端口的“消失”。

4. 实验调试

（1） 查看 VLAN

使用“show vlan”或者“show vlan brief”命令可以查看 VLAN 的信息，以及每个 VLAN 上有什么端口。要注意这里只能看到的是本交换机上哪个端口在 VLAN 上，而不能看到其他交换机的端口在什么 VLAN 上。如下：

```
SW1#show vlan
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/16, Fa0/17
                                                Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                                Fa0/22, Fa0/23, Fa0/24, Gi0/1
                                                Gi0/2
2    VLAN2                            active
3    VLAN3                            active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
（此处省略）
```

//在交换上，VLAN1是默认VLAN，不能删除，也不能改名。此外还有1002、1003等VLAN的存在。

（2） VLAN 间的通信

由于 f0/1 和 f0/2 属于不同的 VLAN，从 R1 ping 192.168.12.2 应该不能成功了。

13.2 实验 2:trunk 配置

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 配置交换机接口的 trunk

（2） 理解 DTP 的协商规律

2. 实验拓扑

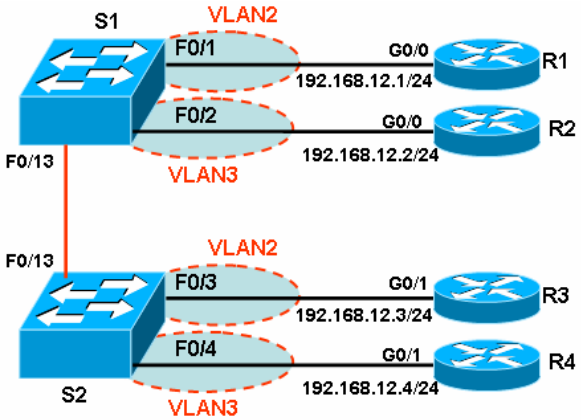

图 13-3 实验 2 拓扑图

3. 实验步骤

在实验 1 的基础上继续本实验。

（1） 根据实验 1 的步骤在 S2 上创建 VLAN，并把接口划分在图 13-3 所示的 VLAN 中

（2） 配置 trunk：

S1(config)#int f0/13

S1(config-if)#switchport trunk encanpsulation dot1q

//以上是配置 trunk 链路的封装类型，同一链路的两端封装要相同。有的交换机，例如 2950 只能封装 dot1q，因此无需执行该命令。

S1(config-if)#switch mode trunk

//以上是把接口配置为 trunk

S2(config)#int f0/13

S2(config-if)#switchport trunk encanpsulation dot1q

S2(config-if)#switch mode trunk

（3） 检查 trunk 链路的状态，测试跨交换机、同一 VLAN 主机间的通信

使用“show interface f0/13 trunk”可以查看交换机端口的 trunk 状态，如下：

Port	Mode	Encapsulation	Status	Native vlan
Fa0/13	on	802.1q	trunking	1

//f0/13 接口已经为 trunk 链路了，封装为 802.1q

Port	Vlans allowed on trunk
Fa0/13	1-4094

Port	Vlans allowed and active in management domain
Fa0/13	1-3

Port	Vlans in spanning tree forwarding state and not pruned
Fa0/13	2-3

需要在链路的两端都确认 trunk 的形成。测试 R1 和 R3、R2 和 R4 之间的通信。由于 R1 和 R3 在同一 VLAN，所以 R1 应该能 ping 通 R3。

（4） 配置 Native VLAN：

S1(config)#int f0/13

S1(config-if)#switchport trunk native vlan 2

//以上是在 Trunk 链路上配置 Native VLAN，我们把它改为 VLAN 2 了，默认是 VLAN 1。

S2(config)#int f0/13

S2(config-if)#switchport trunk native vlan 2

S1#show interface f0/13 trunk

Port	Mode	Encapsulation	Status	Native vlan
Fa0/13	on	802.1q	trunking	2

//可以查看 trunk 链路的 Native VLAN 改为 2 了。

【技术要点】之前介绍说在 Trunk 链路上，数据帧会根据 ISL 或者 802.1Q 被重新封装，然而如果是 Native VLAN 的数据，是不会被重新封装就在 Trunk 链路上传输。很显然链路两端的 Native VLAN 是要一样的。如果不一样，交换机会提示出错。

（5） DTP 配置：

【技术要点】

和 DTP 配置有关的有以下命令，这些命令不能任意组合：

“switchport trunk encapsulation { negotiate | isl | dot1q }”：配置 Trunk 链路上的封装类型，可以是双方协商确定，也可以是指定的 isl 或者 dot1q

“switchport nonegotiate” ：Trunk 链路上不发送协商包，默认是发送的

“switch mode { trunk | dynamic desirable | dynamic auto }”：

- trunk:这个设置将端口置为永久 trunk 模式，封装类型由“switchport trunk encapsulation”命令决定
- dynamic desirable：端口主动变为 trunk，如果另一端为 negotiate、dynamic desirable 、dynamic auto 将成功协商
- dynamic auto：被动协商，如果另一端为 negotiate、dynamic desirable 将成功协商。

如果想把接口配置为negotiate，使用：

```
S1(config-if)#switchport trunk encapsulation { isl | dot1q }
S1(config-if)#switchport mode trunk
S1(config-if)#no switchport negotiate
```

如果想把接口配置为nonegotiate，使用：

```
S1(config-if)#switchport trunk encapsulation { isl | dot1q }
S1(config-if)#switchport mode trunk
S1(config-if)#switchport nonegotiate
```

如果想把接口配置为desirable，使用：

```
S1(config-if)#switchport mode dynamic desirable
S1(config-if)#switchport trunk encapsulation { negotiate | isl | dot1q }
```

如果想把接口配置为auto，使用：

```
S1(config-if)#switchport mode dynamic auto
S1(config-if)#switchport trunk encapsulation  { negotiate | isl | dot1q }
```

我们这里，进行如下配置：

```
S1(config-if)#switchport mode dynamic desirable
S1(config-if)#switchport trunk encapsulation negotiate
S2(config-if)#switchport mode dynamic auto
S2(config-if)#switchport trunk encapsulation negotiate
```

```
S1#show interfaces f0/13 trunk
Port        Mode         Encapsulation  Status        Native vlan
Fa0/13      desirable    n-isl          trunking      1
```

//可以看到 trunk 已经形成，封装为 n-isl，这里的“n”表示封装类型也是自动协商的。需要在两端都进行检查，确认两端都形成 Trunk 才行。

```
Port      Vlans allowed on trunk
Fa0/13      1-4094

Port        Vlans allowed and active in management domain
Fa0/13      1-3

Port        Vlans in spanning tree forwarding state and not pruned
```

Fa0/13　　2-3

【提示】由于交换机有缺省配置，进行以上配置后，使用“show running”可能看不到我们配置的命令。默认时 catalyst 2950 和 3550 的配置是 desirable 模式；而 catalyst 3560 是 auto 模式，所以两台 3560 交换机之间不会自动形成 Trunk，3560 交换机和 2950 交换机之间却可以形成 Trunk。

13.4 实验 3:VTP 配置

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 VTP 的三种模式

（2） 熟悉 VTP 的配置

2. 实验拓扑

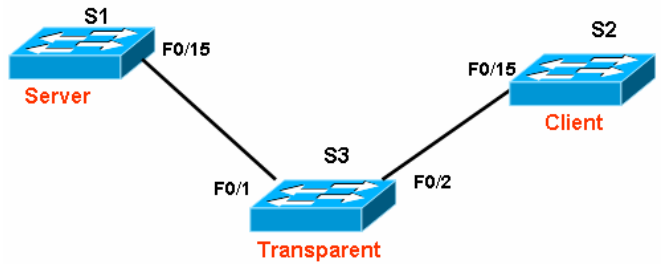

图 13-4　实验 3 拓扑图

3. 实验步骤

（1） 把三台交换机的配置清除干净，重启交换机

```
S1#delete flash:vlan.dat
S1#erase startup-config
S1#reload
```

（2） 检查 S1 和 S3 之间、S3 和 S2 之间链路 trunk 是否自动形成，如果没有请参照实验 2 步骤配置 trunk

（3） 配置 S1 为 VTP server

```
S1(config)#vtp mode server
Device mode already VTP SERVER.
//以上配置 S1 为 VTP server，实际上这是默认值
S1(config)#vtp domain VTP-TEST
Changing VTP domain name from NULL to VTP-TEST
//以上配置 VTP 域名
S1(config)#vtp password cisco
Setting device VLAN database password to cisco
//以上配置 VTP 的密码，目的是为了安全，防止不明身份的交换机加入到域中
```

（4） 配置 S3 为 VTP transparent

```
S3#vlan database
S3(vlan)#vtp transparent
Setting device to VTP TRANSPARENT mode.
S3(vlan)#vtp domain VTP-TEST
```

Domain name already set to VTP-TEST .
S3(vlan)#vtp password cisco
Setting device VLAN database password to cisco.
【提示】有的 IOS 版本只支持在 vlan database 下配置 vlan。
（5） 配置 S2 为 VTP client
S2(config)#vtp mode client
Setting device to VTP CLIENT mode.
S2(config)#vtp domain VTP-TEST
Domain name already set to VTP-TEST.
S2(config)#vtp password cisco

4. 实验调试

（1） 在 S1 上创建 VLAN，检查 S2、S3 上的 VLAN 信息
S1(config)#vlan 2
S1(config)#vlan 3
S2#show vlan

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8
2	VLAN0002	active	
3	VLAN0003	active	
1002	fddi-default	act/unsup	

//可以看到 S2 已经学习到了在 S1 上创建的 VLAN 了。

S3#show vlan

VLAN	Name	Status	Ports
1	default	active	Fa0/3, Fa0/4, Fa0/5, Fa0/6 Fa0/7, Fa0/8, Fa0/9, Fa0/10 Fa0/11, Fa0/12
1002	fddi-default	active	
1003	token-ring-default	active	
1004	fddinet-default	active	
1005	trnet-default	active	

//可以看到 S2 上有了 VLAN2 和 VLAN3，而 S3 上并没有，因为 S3 是透明模式。
（2） 查看 VTP 信息
S1#show vtp status
VTP Version : 2 //该 VTP 支持版本 2
Configuration Revision : 2 //修订号为 2，该数值非常重要
Maximum VLANs supported locally : 1005
Number of existing VLANs : 7 //VLAN 数量
VTP Operating Mode : Server //VTP 模式
VTP Domain Name : VTP-TEST //VTP 域名
VTP Pruning Mode : Disabled //VTP 修剪没有启用

VTP V2 Mode   : Disabled //VTP 版本 2 没有启用，现在是版本 1
VTP Traps Generation   : Disabled
MD5 digest   : 0xD4 0x30 0xE7 0xB7 0xDC 0xDF 0x1B 0xD8
Configuration last modified by 0.0.0.0 at 3-1-93 00:22:16
Local updater ID is 0.0.0.0 (no valid interface found)

（3） 观察 VTP 的 revision 数值

在 S1 上，修改、创建或者删除 VLAN，在 S2、S3 上观察 revision 数值是否增加 1。

（4） 配置修剪、版本 2

```
S1(config)#vtp pruning
S1(config)#vtp version 2
S1#show vtp status
VTP Version                    : 2
Configuration Revision         : 4
Maximum VLANs supported locally : 1005
Number of existing VLANs       : 7
VTP Operating Mode             : Server
VTP Domain Name                : VTP-TEST
VTP Pruning Mode               : Enabled   //VTP 修剪启用了
VTP V2 Mode                    : Enabled   //VTP 版本为 2 了
VTP Traps Generation           : Disabled
MD5 digest                     : 0xA6 0x56 0x25 0xDE 0xE2 0x39 0x6A 0x10
Configuration last modified by 0.0.0.0 at 3-1-93 00:32:28
Local updater ID is 0.0.0.0 (no valid interface found)
```

【提示】VTP 修剪和 VTP 版本只需要在一个 VTP server 进行即可，其他 server 或者 client 会自动跟着更改。VTP 修剪是为了防止不必要的流量从 trunk 链路上通过，通常需要启用。

13.5 实验 4：EtherChannel 配置

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） Etherchannel 的工作原理

（2） Etherchannel 的配置

2. 实验拓扑

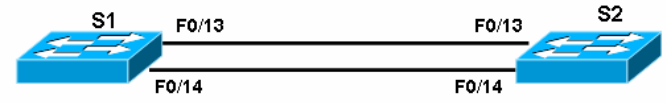

图 13-5 实验 4 拓扑图

3. 实验步骤

构成EtherChannel的端口必须具有相同的特性，如双工模式、速度、Trunking的状态等。配置EtherChannel有手动配置和自动配置（PAGP或者LAGP）两种方法，自动配置就是让EtherChannel协商协议自动协商EtherChannel的建立。

(1) 手动配置EtherChannel

```
S1 (config)#interface port-channel 1
```

//以上是创建以太通道，要指定一个唯一的通道组号，组号的范围是1～6的正整数。要取消EtherChannel时用“no interface port-channel 1”

```
S1(config)#interface f0/13
S1(config-if)#channel-group 1 mode on
S1(config)#interface f0/14
S1(config-if)#channel-group 1 mode on
```

//以上将物理接口指定到已创建的通道中。

```
S1(config)#int port-channel 1
S1(config-if)#switchport mode trunk
S1(config-if)#speed 100
S1(config-if)#duplex full
```

//以上配置通道中的物理接口的属性

```
S2(config)#interface port-channel 1
S2(config)#interface f0/13
S2(config-if)#channel-group 1 mode on
S2(config)#interface f0/14
S2(config-if)#channel-group 1 mode on
S2(config)#int port-channel 1
S2(config-if)#switchport mode trunk
S2(config-if)#speed 100
S2(config-if)#duplex full
```

```
S1(config)# port-channel load-balance dst-mac
S2(config)# port-channel load-balance dst-mac
```

//以上是配置EtherChannel的负载平衡方式，命令格式为“port-channel load-balance *method* ”， 负载平衡的方式有：dst-ip、dst-mac、src-dst-ip、src-dst-mac等。

（3） 查看etherchannel信息

```
S1#show etherchannel summary
Flags:  D - down        P - in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port

Number of channel-groups in use: 1
Number of aggregators:           1

Group  Port-channel  Protocol    Ports
```

```
------+------------+-----------+-----------------------------------------------
```

1 Po1(SU) - Fa0/13(P) Fa0/14(P)

//可以看到 EtherChannel 已经形成，“SU”表示 EtherChannel 正常，如果显示为“SD”，把 EtherChannel 接口关掉重新开启。

（4） 配置 PAGP 或者 LAGP

【技术要点】

要想把接口配置为 PAGP 的 desirable 模式使用命令：**“channel-group 1 mode desirable”;**
要想把接口配置为 PAGP 的 auto 模式使用命令：**“channel-group 1 mode auto”;**
要想把接口配置为 LACP 的 active 模式使用命令：**“channel-group 1 mode active”;**
要想把接口配置为 LACP 的 passive 模式使用命令：**“channel-group 1 mode passive”**。

我们这里进行如下配置：

```
S1(config)#interface range f0/13 -14
S1(config-if)#channel-group 1 mode desirable
S2(config)#interface range f0/13 -14
S2(config-if)#channel-group 1 mode desirable

S1#show etherchannel summary
Flags:  D - down        P - in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port

Number of channel-groups in use: 1
Number of aggregators:           1
Group  Port-channel  Protocol    Ports
------+------------+-----------+-----------------------------------------------
1      Po1(SU)         PAgP      Fa0/13(P)   Fa0/14(P)
```

//可以看到 etherchannel 协商成功。注意应在链路的两端都进行检查，确认两端都形成以太通道才行。

13.6 本章小结

本章首先介绍了交换机上的 VLAN 创建以及如何把接口划分在指定的 VLAN 中。在交换机之间级联的链路应该配置为 Trunk，Trunk 是否形成可以通过 DTP 协商，Trunk 有两种封装方式。VTP 可以让我们集中化管理 VLAN 的信息，VTP 有三种模式，不同模式可以完成不同的功能。EtherChannel 技术可以把多条链路捆绑起来形成大带宽的逻辑链路，EtherChannel 是否形成也可以用协议自动协商。表 13-4 是本章出现的命令。

表 13-4 本章命令汇总

命令	作用
vlan database	进入到 vlan database 配置模式
vlan 2 name VLAN2	创建 vlan 2
switch access vlan 2	把端口划分到 VLAN 2 中
interface range f0/2 - 3	批量配置接口的属性
show vlan	查看 VLAN 的信息
switchport trunk encanpsulation	配置 trunk 链路的封装类型
switch mode trunk	把接口配置为 trunk
show interface f0/13 trunk	查看交换机端口的 trunk 状态
switchport nonegotiate	Trunk 链路上不发送 trunk 协商包
vtp mode server	配置交换机为 VTP server
vtp domain VTP-TEST	配置 VTP 域名
vtp password cisco	配置 VTP 的密码
vtp mode client	配置交换机为 VTP client
vtp transparent	配置交换机为 VTP transparent
show vtp status	显示 vtp 的状态
vtp pruning	启用 VTP 修剪
vtp version 2	VTP 版本为 2
interface port-channel 1	创建以太通道
channel-group 1 mode on	把接口加入到以太网通道中，并指明以太通道模式
port-channel load-balance dst-mac	配置 etherChannel 的负载平衡方式
show etherchannel summary	查看 etherchannel 的简要信息

第 14 章 STP

为了减少网络的故障时间，我们经常会采用冗余拓扑。STP 可以让具有冗余结构的网络在故障时自动调整网络的数据转发路径。STP 重新收敛时间较长，通常需要 30—50 秒，为了减少这个时间，引入了一些补充技术，例如 uplinkfast、backbonefast 等。RSTP 则在协议上对 STP 作了根本的改进形成新的协议，从而减少收敛时间。STP 还有许多改进，例如 PVST、MST 协议，以及安全措施，本章将介绍这些常用的配置。

14.1 STP 简介

14.1.1 基本 STP

为了增加局域网的冗余性，我们常常会在网络中引入冗余链路，然而这样却会引起交换环路。交换环路会带来三个问题：广播风暴、同一帧的多个拷贝、交换机 CAM 表不稳定。STP(STP，Spanning Tree Protocol)可以解决这些问题，STP 基本思路是阻断一些交换机接口，构建一棵没有环路的转发树。STP 利用 BPDU(Bridge Protocol Data Unit)和其他交换机进行通信，从而确定哪个交换机该阻断哪个接口。在 BPDU 中有几个关键的字段，例如：根桥 ID、路径代价、端口 ID 等。

为了在网络中形成一个没有环路的拓扑，网络中的交换机要进行以下三个步骤：（1）选举根桥、（2）选取根口、（3）选取指定口。这些步骤中，哪个交换机能获胜将取决于以下因素（按顺序进行）：

（1） 最低的根桥 ID；
（2） 最低的根路径代价；
（3） 最低发送者桥 ID；
（4） 最低发送者端口 ID。

每个交换机都具有一个唯一的桥 ID，这个 ID 由两部分组成：网桥优先级+MAC 地址。网桥优先级是一个 2 个字节的数，交换机的默认优先级为 32768；MAC 地址就是交换机的 MAC 地址。具有最低桥 ID 的交换机就是根桥。根桥上的接口都是指定口，会转发数据包。

选举了根桥后，其他的交换机就成为非根桥了。每台非根桥要选举一条到根桥的根路径。STP 使用路径 Cost 来决定到达根桥的最佳路径（Cost 是累加的，带宽大的链路 Cost 低），最低 Cost 值的路径就是根路径，该接口就是根口；如果 Cost 职一样，就根据选举顺序选举根口。根口是转发数据包的。

交换机的其他接口还要决定是指定口还是阻断口，交换机之间将进一步根据上面的四个因素来竞争。指定口是转发数据帧的。剩下的其它的接口将被阻断，不转发数据包。这样网络就构建出一棵没有环路的转发树。

当网络的拓扑发生变化时，网络会从一个状态向另一个状态过渡，重新打开或阻断某些接口。交换机的端口要经过几种状态： 禁用（Disable）、阻塞（Blocking）、监听状态(Listening)、学习状态（Learning）、最后是转发状态(Forwarding)。

14.1.2 PVST

当网络上有多个 VLAN 时，PVST(Per Vlan STP)会为每个 VLAN 构建一棵 STP 树。这样的好处是可以独立地为每个 VLAN 控制哪些接口要转发数据，从而实现负载平衡。缺点是如果 VLAN 数量很多，会给交换机带来沉重的负担。Cisco 交换机默认的模式就是 PVST。

14.1.3 portfast、uplinkfast、backbonefast

STP 的收敛时间通常需要 30—50 秒。为了减少收敛时间，有一些改善措施。Portfast 特性使得以太网接口一旦有设备接入，就立即进入转发状态，如果接口上连接的只是计算机或者其他不运行 STP 的设备，这是非常合适的。

Uplinkfast 则经常用在接入层交换机上，当它连接到主干交换机上的主链路上故障时，能立即切换到备份链路上，而不需要经过 30 秒或者 50 秒。Uplinkfast 只需要在接入层交换机上配置即可。

Backbonefast 则主要用在主干交换机之间，当主干交换机之间的链路上故障时，可以比原有的 50 秒少 20 秒就切换到备份链路上。Backbonefast 需要在全部交换机上配置。

14.1.4 RSTP

RSTP 实际上是把减少 STP 收敛时间的一些措施融合在 STP 协议中形成新的协议。RSTP 中，接口的角色有：根接口、指定接口、备份接口(Backup Interface)、替代接口(Alternate Interface)。接口的状态有：丢弃（Discarding）、学习状态（Learning）、转发状态(Forwarding)。接口还分为边界接口（Edge Port）、点到点接口（Point-to-Point Port）、共享接口（Share Port）。

14.1.5 MST

在 PVST 中，交换机为每个 VLAN 都构建一棵 STP 树，不仅会带来 CPU 的很大负载，也会占用大量的带宽。MST 则是把多个 VLAN 映射到一个 STP 实例上，从而减少了 STP 实例。MST 可以和 STP、PVST 配合使用。对于运行 STP、PVST 的交换机来说，一个 MST 域看起来就像一台交换机。

14.1.6 STP 防护

STP 协议并没有什么措施对交换机的身份进行认证。在稳定的网络中如果接入非法的交换机将可能给网络中的 STP 树带来灾难性的破坏。有一些简单的措施来保护网络，虽然这些措施显得软弱无力。Root Guard 特性将使得交换机的接口拒绝接收比原有根桥优先级更高的 BPDU。而 BPDU Guard 主要是和 portfast 特性配合使用，portfast 使得接口一有计算机接入就立即进入转发状态,然而万一这个接口接入的是交换机很可能造成环路。BPDU Guard 可以使得 portfast 接口一旦接收到 BPDU，就关闭该接口。

14.2 实验 1：STP、PVST

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 STP 的工作原理

（2） 掌握 STP 树的控制

（3） 利用 PVST 进行负载平衡

2. 实验拓扑

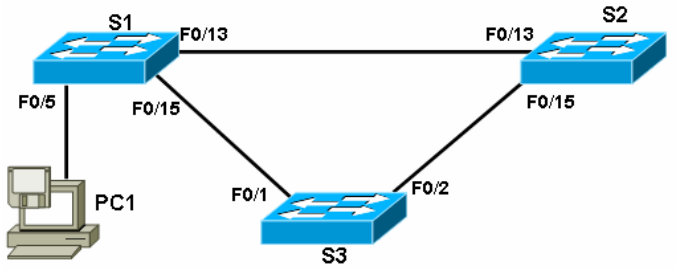

图 14-1 实验 1、实验 2、实验 4 拓扑图

图 14-1 中，S1 和 S2 模拟为核心层的交换机，而 S3 为接入的交换机。S1 和 S2 实际上是三层交换机，我们这里并不利用其三层功能，所以它们也采用二层交换机的图标。

3. 实验步骤

我们要在网络中配置 2 个 VLAN，不同 VLAN 的 STP 具有不同的根桥，实现负载平衡。

（1） 步骤 1：利用 VTP 在交换机上创建 VLAN2，在 S1 和 S2 之间的链路配置 Trunk

```
S1(config)#vtp domain VTP-TEST
Changing VTP domain name from NULL to VTP-TEST
S1(config)#vlan 2
```

//在 S1 上配置 VTP 的域名，并创建 VLAN 2。由于默认时 S2 和 S3 的 VTP 域名为空，它们将自动学习到 S1 的 VTP 域名，同时 S2、S3 也将自动学习到 VLAN 2，请确认是否成功。

```
S1(config)#int f0/14
S1(config-if)#shutdown
```

//关闭该接口，以免影响我们的实验

```
S1(config)#int f0/13
S1(config-if)#switchport trunk encapsulation dot1q
S1(config-if)#switchport mode trunk
```

//S1 的 f0/13 改为 negotiate 后，由于默认时 S2 的 f0/13 为 auto 模式，S1 和 S2 将自动协商成功 Trunk。而默认时 S3 的以太网接口就是 desirable 模式，所以 S3 和 S1、S2 的链路也自动协商成功 Trunk。请确认三条链路的 Trunk 是否成功。

（2） 步骤 2：检查初始的 STP 树

```
S1#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
```

//以上表明运行的 STP 协议是 IEEE 的 802.1D

```
  Root ID     Priority    32768
              Address     0009.b7a4.b181
              Cost        19
              Port        17 (FastEthernet0/15)
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

//以上显示 VLAN 1 的 STP 树的根桥信息，通过根桥的 MAC 地址可以确定 S3 是根桥。这是因为 S3 是较早的交换机，具有较低的 MAC 地址。由于 S3 是一台低端的交换机，成为根桥显然是不合理的。

```
  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
```

```
             Address     0018.ba11.f500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300
```

//以上显示该交换机的桥 ID

```
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/13           Altn BLK 19        128.15   P2p
Fa0/15           Root FWD 19        128.17   P2p
```

//以上显示该交换机各个接口的状态，f0/13 为阻断状态，f0/15 为根口

```
VLAN0002
  Spanning tree enabled protocol ieee
  Root ID    Priority    32768
             Address     0009.b7a4.b182
             Cost        19
             Port        17 (FastEthernet0/15)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32770  (priority 32768 sys-id-ext 2)
             Address     0018.ba11.f500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/13           Altn BLK 19        128.15   P2p
Fa0/15           Root FWD 19        128.17   P2p
```

//以上是 VLAN 2 的 STP 树情况，VLAN 2 的 STP 树和 VLAN 1 的类似。默认时，Cisco 交换机会为每个 VLAN 都生成一个单独的 STP 树，称为 PVST(Per VLAN Spanning Tree)。

【技术要点】需要仔细分析为什么 STP 会是目前这种情况。三个交换机的默认优先级都是 32768，而 S3 的 MAC 较低，所以成为了根桥，则 S3 上的 f0/1 和 f0/2 是指定口，处于 Forword 状态。S1 有两个接口可以到达 S3，一个接口是 f0/13，到达 S3 的 Cost 为 19＋19＝38，另一个接口是 f0/15，到达 S1 的 Cost 为 19，因此 f0/15 是根口，处于 Forword 状态。同样 S2 上，f0/15 也是根口，处于 Forword 状态。在 S1 和 S2 之间的链路上，要选举出一个指定口。根据选举的要素，根桥的 ID 是一样的，不能决出胜负；到达根桥的 Cost 值也是一样的，都为 19，不能决出胜负；但是发送者桥 ID 不一样，S1 的 MAC 地址高，S2 的 MAC 地址低，S2 获胜，所以 S2 的 f0/13 是指定口，处于 Forward 状态，S1 的 f0/13 就处于 Block 状态了。

（3） 步骤 3：控制 S1 为 VLAN1 的根桥，S2 为 VLAN2 的根桥

```
S1(config)#spanning-tree vlan 1 priority 4096
S2(config)#spanning-tree vlan 2 priority 4096
```

//对于 VLAN 1 来说，S1 的优先级为 4096，而 S2 和 S3 保持默认值 32768，这样 S1 就成为了 VLAN 1 的根桥。同样我们控制 S2 成为了 VLAN 2 的根桥。优先级通常要是 4096 的倍数。

S1#show spanning-tree

VLAN0001

  Spanning tree enabled protocol ieee

  Root ID   Priority   4097

             Address   0018.ba11.f500

             This bridge is the root

             Hello Time   2 sec Max Age 20 sec Forward Delay 15 sec

//S1 成为了 VLAN 1 的根桥了

  Bridge ID Priority   4097   (priority 4096 sys-id-ext 1)

             Address   0018.ba11.f500

             Hello Time   2 sec Max Age 20 sec Forward Delay 15 sec

             Aging Time 15

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/13	**Desg**	**FWD**	**19**	**128.15**	**P2p**
Fa0/15	**Desg**	**FWD**	**19**	**128.17**	**P2p**

//对于 VLAN 1 来说，f0/13 和 f0/15 是指定口，都处于转发状态了

VLAN0002

  Spanning tree enabled protocol ieee

  Root ID   Priority   4098

             Address   0018.ba11.eb80

             Cost   19

             Port   15 (FastEthernet0/13)

             Hello Time   2 sec Max Age 20 sec Forward Delay 15 sec

//S2 成为了 VLAN 2 的根桥了

  Bridge ID Priority   32770 (priority 32768 sys-id-ext 2)

             Address   0018.ba11.f500

             Hello Time   2 sec Max Age 20 sec Forward Delay 15 sec

             Aging Time 15

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/13	**Root**	**FWD**	**19**	**128.15**	**P2p**
Fa0/15	**Altn**	**BLK**	**19**	**128.17**	**P2p**

//对于 VLAN 2 来说，f0/13 是根口，处于转发状态，而 f0/15 却是阻断状态

S3#show spanning-tree brief

VLAN1

  Spanning tree enabled protocol ieee

  Root ID   Priority   4097

             Address   0018.ba11.f500

             Cost   19

             Port   1 (FastEthernet0/1)

```
            Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768
             Address     0009.b7a4.b181
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                   Designated
Name                Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet0/1     128.1    128    19 FWD     0  4097 0018.ba11.f500 128.17
FastEthernet0/2     128.2    128    19 FWD    19 32768 0009.b7a4.b181 128.2
```

//在 S3 上，对于 VLAN1，S3 的 f0/1 和 f0/2 都处于转发状态。

```
VLAN2
  Spanning tree enabled protocol ieee
  Root ID    Priority    4098
             Address     0018.ba11.eb80
             Cost        19
             Port        2 (FastEthernet0/2)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32768
             Address     0009.b7a4.b182
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                   Designated
Name                Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet0/1     128.1    128    19 FWD    19 32768 0009.b7a4.b182 128.1
FastEthernet0/2     128.2    128    19 FWD     0  4098 0018.ba11.eb80 128.17
```

//S3 上，对于 VLAN2，S3 的 f0/1 和 f0/2 也都处于转发状态。

（4） 步骤 4：控制指定口

在步骤 3 中可以看到对于 VLAN 1，S1 成为了根桥，S1 的 f0/13 和 f0/15 处于转发状态；S2 的 f0/13 是根口，也处于转发状态；S3 的 f0/1 是根口，也处于转发状态；然而 S2 和 S3 之间的链路上，却是低端交换机 S3 的 f0/2 在转发数据，原因在于 S2 和 S3 在竞争指定口时，由于 S3 的 MAC 较低而获胜了，这是不合理的。VLAN 2 的情况类似。

我们要控制指定口，这可以通过改变优先级实现，如下：

```
S2(config)#spanning-tree vlan 1 priority 8192
S1(config)#spanning-tree vlan 2 priority 8192
```

//对于 VLAN 1 来说，S2 的优先级为 8192,比 S1 的 4096 低，不至于成为根桥，但是比 S3 的 32768 低，所以在竞争指定口时会获胜。VLAN 2 的情况类似。

```
S3#show spanning-tree brief
```

VLAN1

（此处省略）

Interface Name	Port ID	Prio	Cost	Sts	Designated Cost	Bridge ID	Port ID
FastEthernet0/1	128.1	128	19	FWD	0	4097 0018.ba11.f500	128.17
FastEthernet0/2	128.2	128	19	BLK	19	8193 0018.ba11.eb80	128.17

//S3 上，对于 VLAN1，S3 的 f0/1 处于转发状态，而 f0/2 处于阻断状态。

VLAN2

（此处省略）

Interface Name	Port ID	Prio	Cost	Sts	Designated Cost	Bridge ID	Port ID
FastEthernet0/1	128.1	128	19	BLK	19	8194 0018.ba11.f500	128.17
FastEthernet0/2	128.2	128	19	FWD	0	4098 0018.ba11.eb80	128.17

// S3 上，对于 VLAN 2，S3 的 f0/1 处于阻断状态，而 f0/2 处于转发状态，这样起到了负载平衡的作用。

14.3 实验 2：portfast、uplinkfast、backbonefast

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 portfast 的工作场合和配置

（2） 理解 uplinkfast 的工作场合和配置

（3） 理解 backbonefast 的工作场合和配置

2. 实验拓扑

如图 14-1。

3. 实验步骤

在实验 1 的基础上继续本实验，我们将只关心 VLAN 1 的 STP 树。

（1） 步骤 1：配置 portfast

图 14-1 中，S1 的 f0/5 是用于接入计算机。当计算机接入时，f0/5 接口立即进入 Listening 状态，随后经过 Learning，最后才成为 Forwarding，这期间需要 30 秒的时间。这对于有些场合是不可忍受的，可以配置 portfast 特性，使得计算机一接入，接口立即进入 Forwarding。

```
S1(config)#int f0/5
S1(config-if)#spanning-tree portfast
%Warning: portfast should only be enabled on ports connected to a single
 host. Connecting hubs, concentrators, switches, bridges, etc... to this
 interface  when portfast is enabled, can cause temporary bridging loops.
 Use with CAUTION
```

%Portfast has been configured on FastEthernet0/5 but will only
have effect when the interface is in a non-trunking mode.
//交换机会警告该接口只能用于接入计算机或者路由器，不要接入其他的交换机

（2） 步骤 2：配置 uplinkfast

先确认实验 1 的 STP 树已经正确。在图 14-1 中的 S1 上，关闭 f0/15 接口，在 S3 上反复执行“**show spanning-tree vlan 1 brief**”观察 f0/2 接口的状态变化：

FastEthernet0/2 128.2 128 3019 LIS 19 8193 0018.ba11.eb80 128.17

大约 15 秒后变为：

FastEthernet0/2 128.2 128 3019 LRN 19 8193 0018.ba11.eb80 128.17

大约 15 秒后变为：

FastEthernet0/2 128.2 128 3019 FWD 19 8193 0018.ba11.eb80 128.17

合计大约 15＋15＝30 秒，f0/2 变为转发状态。

S3(config)#**spanning-tree uplinkfast**
S1(config)#**int f0/15**
S1(config-if)#**no shutdown**
S1(config-if)#**shutdown** //等 STP 重新稳定后，才执行该语句

在 S3 上重复执行“**show spanning-tree vlan 1 brief**”，可以看到 f0/2 很快就进入了 Forwarding 状态。

【技术要点】没有配置 uplinkfast 时，交换机 S3 如果能直接检测到 f0/1 接口上的链路故障，f0/2 会立即进入 Listen 状态，这样 30 秒就能进入 Forward 状态。然而如果 S1 和 S3 之间存在一个 Hub，S1 上的 f0/15 接口故障了，S3 将无法直接检测到故障，S3 只能等待 10 个周期没有收到 S1 的 BPDU（每个周期 2 秒），20 秒中后，S3 的 f0/2 才进入 Listen 状态，这样总共 50 秒才就能进入 Forward 状态。所以 STP 重新收敛的时间通常需要 30—50 秒。

（3） 步骤 3：配置 backbonefast

打开 S1 上 f0/15 接口，确认 STP 树已经正确。在图 14-1 中的 S1 上，关闭 f0/13 接口，在 S3 上反复执行“**show spanning-tree vlan 1 brief**”观察 f0/2 接口的状态变化：

FastEthernet0/2 128.2 128 3019 BLK 19 8193 0018.ba11.eb80 128.17

大约 20 秒后变为：

FastEthernet0/2 128.2 128 3019 LIS 19 8193 0018.ba11.eb80 128.17

大约 15 秒后变为：

FastEthernet0/2 128.2 128 3019 LRN 19 8193 0018.ba11.eb80 128.17

大约 15 秒后变为：

FastEthernet0/2 128.2 128 3019 FWD 19 8193 0018.ba11.eb80 128.17

合计大约 20＋15＋15＝50 秒，f0/2 变为转发状态。

S1(config)#**spanning-tree backbonefast**
S2(config)#**spanning-tree backbonefast**
S3(config)#**spanning-tree backbonefast**
S1(config)#**int f0/13**
S1(config-if)#**no shutdown**
S1(config-if)#**shutdown** //等 STP 重新稳定后，才执行该语句

在 S3 上重复执行“**show spanning-tree vlan 1 brief**”，可以看到 f0/2 很快就进入了 Listening 状态，合计大约 15＋15＝30 秒后，f0/2 就变为转发状态，比之前的 50 秒少了 20 秒。

【提示】uplinkfast 命令只需要在 S3 配置即可，而 backbonefast 命令需要在 S1、S2、S3 三台交换机上都配置。

14.4 实验 3:RSTP

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 熟悉 RSTP 的配置

2. 实验拓扑

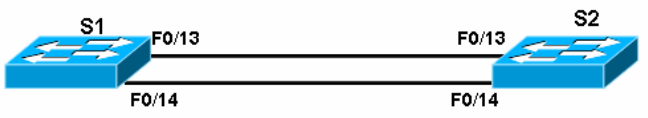

图 14-2 实验 3 拓扑图

3. 实验步骤

（1） 步骤 1：请把两台交换机的配置清除干净，重启交换机

```
S1#delete flash:vlan.dat
S1#erase startup-config
S1#reload
S2#delete flash:vlan.dat
S2#erase startup-config
S2#reload
```

（2） 步骤 2：配置 S1 和 S2 之间的 Trunk

```
S1(config)#int f0/13
S1(config-if)#switchport trunk encapsulation dot1q
S1(config-if)#switchport mode trunk
S1(config)#int f0/14
S1(config-if)#switchport trunk encapsulation dot1q
S1(config-if)#switchport mode trunk
```

（3） 步骤 3：配置 S1 成为根桥

```
S1(config)#spanning-tree vlan 1 priority 4096
```

在 S1 和 S2 上用“**show spanning-tree**”命令检查 STP 的情况，S2 的 f0/14 应该处于阻断状态。

【技术要点】S1 是根桥，S2 要选取到达 S1 的根路径，有两条路径，Cost 都为 19。这时由于 S2 在 f0/13 接口上收到的 BPDU 中，发送者（S1）端口号为 13；在 f0/14 接口上收到的 BPDU 中，发送者端口号为 14。所以 f0/13 被选举为根口了，f0/14 则只能被阻断了。

（4） 步骤 4：在 S2 上关闭 f0/13 接口，观察 STP 树的重新生成

在 S2 上关闭 f0/13 接口，重复执行“show spanning-tree”，可以看到 f0/14 经过 30 秒后才进入了 Forwarding 状态。

（5） 步骤 5：配置 RSTP

```
S1(config)#spanning-tree mode rapid-pvst
S2(config)#spanning-tree mode rapid-pvst
```

（6） 步骤 6：在 S2 上关闭 f0/13 接口，观察 STP 树的重新生成

在 S2 上重新打开 f0/13 接口，确认 STP 稳定后，在 S2 上关闭 f0/13 接口，重复执行“show spanning-tree”，可以看到 f0/14 立即进入了 Forwarding 状态。说明 RSTP 的收敛比普通 STP 有了很大的改善。

（7） 步骤 7：配置链路类型

```
S1(config)#int range f0/13 -14
S1(config-if-range)#duplex full
S1(config-if-range)#spanning-tree link-type point-to-point
S2(config)#int range f0/13 -14
S2(config-if-range)#duplex full
S2(config-if-range)#spanning-tree link-type point-to-point
```

//S1 和 S2 之间的链路是 Trunk 链路，自动协商为全双工，RSTP 会自动把它们的链路类型标识为点到点。我们这里强制配置了一遍。

【技术要点】RSTP 中接口分为边界接口（Edge Port）、点到点接口（Point-to-Point Port）、共享接口（Share Port）。如果接口上配置了 spanning portfast，接口就为边界接口；如果接口是半双工，接口就为共享接口；如果接口是全双工，接口就为点到点接口。在接口上指明类型有利于 RSTP 的运行。

14.5 实验 4：MST

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 MST 的工作原理

（2） 掌握 MST 的配置

2. 实验拓扑

如图 14-1。

3. 实验步骤

我们要在网络中创建 4 个 VLAN，VLAN 1 和 VLAN 2 使用 MST 实例 1，VLAN 3 和 VLAN 4 使用 MST 实例 2。

（1） 步骤 1：利用 VTP 在交换机上创建 VLAN，在 S1 和 S2 之间的链路配置 Trunk

```
S1(config)#vtp domain VTP-TEST
Changing VTP domain name from NULL to VTP-TEST
S1(config)#vlan 2
S1(config)#vlan 3
S1(config)#vlan 4
```

S1(config)#int f0/14
S1(config-if)#shutdown
//关闭该接口，以免影响我们的实验
S1(config)#int f0/13
S1(config-if)#switchport trunk encapsulation dot1q
S1(config-if)#switchport mode trunk
S2(config)#int f0/13
S2(config-if)#switchport trunk encapsulation dot1q
S2(config-if)#switchport mode trunk
（2） 步骤 2：配置 MST
只有 S1 和 S2 才能支持 MST。
S1(config)#spanning-tree mode mst
//以上把生成树的模式改为 MST，默认时是 PVST。
S1(config)#spanning-tree mst configuration
//以上是进入 MST 的配置模式下
S1(config-mst)#name TEST-MST
//以上命名 MST 的名字
S1(config-mst)#revision 1
//以上配置 MST 的 revision 号，只有名字和 revision 号相同的交换机才是在同一个 MST 区域
S1(config-mst)#instance 1 vlan 1-2
//以上是把 VLAN 1 和 VLAN 2 的生成树映射到实例 1
S1(config-mst)#instance 2 vlan 3-4
//以上是把 VLAN 3 和 VLAN 4 的生成树映射到实例 2，我们这里一共有三个 MST 实例，实例 0 是系统要使用的
S1(config-mst)#exit
//要退出，配置才能生效
S1(config)#spanning-tree mst 1 priority 8192
S1(config)#spanning-tree mst 2 priority 12288
//以上配置 S1 为 MST 实例 1 的根桥

S2(config)#spanning-tree mode mst
S2(config)#spanning-tree mst configuration
S2(config-mst)#name TEST-MST
S2(config-mst)#revision 1
S2(config-mst)#instance 1 vlan 1-2
S2(config-mst)#instance 2 vlan 3-4
S2(config-mst)#exit
S2(config)#spanning-tree mst 1 priority 12288
S2(config)#spanning-tree mst 2 priority 8192
//以上配置 S2 为 MST 实例 2 的根桥
（3） 步骤 3：检查生成树
S1#show spanning-tree
MST00

```
  Spanning tree enabled protocol mstp
```
//以上表明运行的是MST协议
```
  Root ID    Priority    32768
             Address     0009.b7a4.b181
             Cost        200000
             Port        15 (FastEthernet0/13)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    32768  (priority 32768 sys-id-ext 0)
             Address     0018.ba11.f500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/13           Root FWD 200000    128.15   P2p
Fa0/15           Altn BLK 200000    128.17   P2p Bound(PVST)
```
//以上的MST00是系统要使用的实例，BPDU是通过它来发送的

```
MST01
  Spanning tree enabled protocol mstp
  Root ID    Priority    8193
             Address     0018.ba11.f500
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    8193   (priority 8192 sys-id-ext 1)
             Address     0018.ba11.f500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/13           Desg FWD 200000    128.15   P2p
Fa0/15           Boun BLK 200000    128.17   P2p Bound(PVST)

MST02
  Spanning tree enabled protocol mstp
  Root ID    Priority    8194
             Address     0018.ba11.eb80
             Cost        200000
             Port        15 (FastEthernet0/13)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    12290  (priority 12288 sys-id-ext 2)
             Address     0018.ba11.f500
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/13           Root FWD 200000    128.15   P2p
Fa0/15           Boun BLK 200000    128.17   P2p Bound(PVST)
```

//以上显示的是 S1 上的 MST 实例情况。

S3#show spanning-tree brie

```
VLAN1
  Spanning tree enabled protocol ieee
  Root ID    Priority    32768
             Address     0009.b7a4.b181
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    32768
             Address     0009.b7a4.b181
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 15
Interface                                    Designated
Name                Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet0/1     128.1    128    19 FWD     0 32768 0009.b7a4.b181 128.1
FastEthernet0/2     128.2    128    19 FWD     0 32768 0009.b7a4.b181 128.2
```

（此处省略）

//以上表明 S3 成为了所有 VLAN 的根桥，f0/1 和 f0/2 都处于转发状态，这不是我们想要的。

（4） 步骤 4：控制 S1 成为根桥

S1(config)#spanning-tree mst 0 priority 4096

//注意这里应该配置 MST 0 的优先级，只有 MST 0 才发送 BPDU。

S3#show spanning-tree brief

```
VLAN1
  Spanning tree enabled protocol ieee
  Root ID    Priority    4096
             Address     0018.ba11.f500
             Cost        19
             Port        1 (FastEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

//以上表明 S1 是 VLAN 1 的根桥了

```
  Bridge ID  Priority    32768
             Address     0009.b7a4.b181
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface                                    Designated
Name                Port ID Prio Cost  Sts Cost  Bridge ID            Port ID
------------------- ------- ---- ----- --- ----- -------------------- -------
FastEthernet0/1     128.1    128    19 FWD     0  4096 0018.ba11.f500 128.17
FastEthernet0/2     128.2    128    19 BLK     0 32768 0018.ba11.eb80 128.17
```

（此处省略）

//对于 S3 上所有的 VLAN 来说，f0/2 都是阻断的，无法取得负载平衡。

（5） 步骤 5：控制负载平衡

S3(config)#int f0/2

S3(config-if)#spanning-tree vlan 3 cost 10

S3(config-if)#spanning-tree vlan 4 cost 10

//以上改变 VLAN 3 和 VLAN 4 在 f0/2 接口上的 Cost 值。这样对于 VLAN 3 和 VLAN 4，S3 的 f0/2 接口就处于转发状态了。

14.6 实验 5：STP 保护

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） ROOT GUARD 的使用

（2） BPDU GUARD 的使用

2. 实验拓扑

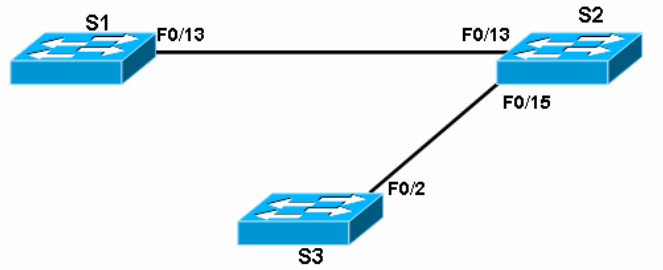

图 14-3 实验 6 拓扑图

3. 实验步骤

（1） 步骤 1：关闭不需要的接口，配置 S1 和 S2 之间的 Trunk，

S1(config)#int f0/14

S1(config-if)#shutdown

S1(config)#int f0/15

S1(config-if)#shutdown

S1(config)#int f0/13

S1(config-if)#switchport trunk encapsulation dot1q

S1(config-if)#switchport mode trunk

（2） 步骤 2：配置 S1 成为根桥

S1(config)#spanning-tree vlan 1 priority 8192

（3） 步骤 3：在 S2 的 f0/15 上配置 guard root

S2(config)#int f0/15

S2(config-if)#spanning-tree guard root

（4） 步骤 4：把 S3 改为根桥，观察 S2 的动作

S3(config)#spanning-tree vlan 1 priority 4096

S2#show spanning-tree inconsistentports

Name Interface Inconsistency

```
------------------- ---------------------- ------------------
VLAN0001            FastEthernet0/15       Root Inconsistent
Number of inconsistent ports (segments) in the system : 1
```

//S2 将从 f0/15 收到 S3 发送的更优的 BPDU，然而由于该接口上配置 Root guard，S2 的接口进入阻断状态。

```
S2#show spanning-tree
VLAN0001
（此处省略）
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Fa0/13           Root FWD 19        128.15   P2p
Fa0/15           Desg BKN*19        128.17   P2p *ROOT_Inc
```

（5） 步骤 5：配置 BPDU Guard

```
S2(config)#int f0/15
S2(config-if)#shutdown
```

//关闭接口

```
S2(config-if)#no spanning-tree guard root
```

//去掉之前的配置

```
S2(config-if)#spanning-tree portfast
S2(config-if)#spanning-tree bpduguard enable
```

//以上配置 BPDU Guard

```
S2(config)#int f0/15
S2(config-if)#no shutdown
0:28:49: %SPANTREE-2-BLOCK_BPDUGUARD: Received BPDU on port FastEthernet0/15 with BPDU Guard enabled. Disabling port.
00:28:49: %PM-4-ERR_DISABLE: bpduguard error detected on Fa0/15, putting Fa0/15 in err-disable state
00:28:50: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/15, changed state to down
```

//交换机从 f0/15 接口收到 S3 的 BPDU，f0/15 被 disable 了

```
S2#show interfaces f0/15
FastEthernet0/15 is down, line protocol is down (err-disabled)
```

//可以看到 f0/15 接口关闭了。要重新开启，请先移除 BPDU 源，在接口下执行“shutdown”、“no shutdown”命令。

14.7 本章小结

本章首先介绍了 STP 的作用和基本工作原理，交换机通过 STP 协议有选择性地阻断了某些接口，从而构建无环路的转发路径，STP 需要选取根桥、根口、指定口。802.1D 的 STP 需要较长时间才收敛，通常为 30—50 秒。本章还介绍减少 STP 收敛的措施：uplinkfast、backbonefast 和 RSTP 协议。默认时 CISCO 交换机为每个 VLAN 构建一棵树，这样方便控制 STP 树，但导致 STP 树数量太多。MST 则可以为多个 VLAN 共同构建一棵树。本章最后介绍了

保护 STP 树的两个简单措施：Root Guard 和 BPDU Guard。表 14-1 是本章出现的命令。

表 14-1 本章命令汇总

命令	作用
show spanning-tree	查看 STP 树信息
spanning-tree vlan 1 priority 4096	配置 VLAN1 的桥优先级
spanning-tree portfast	配置接口为 portfast，当有设备接入时立即进入转发状态
spanning-tree uplinkfast	配置 uplinkfast 特性
spanning-tree backbonefast	配置 backbonefast 特性
spanning-tree mode rapid-pvst	把 STP 的运行模式设为 RSTP+PVST
spanning-tree link-type point-to-point	把接口的链路类型改为点对点
spanning-tree mode mst	把生成树的模式改为 MST
spanning-tree mst configuration	进入 MST 的配置模式
name TEST-MST	命名 MST 的名字
revision 1	配置 MST 的 revision 号
instance 1 vlan 1-2	把 VLAN 1 和 VLAN 2 的生成树映射到实例 1
spanning-tree guard root	在接口上配置 root guard 特性
spanning-tree bpduguard enable	在接口上配置 bpduguard 特性

第 15 章 VLAN 间路由

在交换机上划分 VLAN 后，VLAN 间的计算机就无法通信了。VLAN 间的通信需要借助第三层设备，我们可以使用路由器来实现这个功能，如果使用路由器通常会采用单臂路由模式。实践上，VLAN 间的路由大多是通过三层交换机实现的，三层交换机可以看成是路由器加交换机，然而因为采用了特殊的技术，其数据处理能力比路由器要大得多。本章将分别介绍两种方法的具体配置。

15.1 VLAN 间路由简介

15.1 单臂路由

处于不同 VLAN 的计算机即使它们是在同一交换机上，它们之间的通信也必须使用路由器。可以在每个 VLAN 上都有一个以太网口和路由器连接。采用这种方法，如果要实现 N 个 VLAN 间的通信，则路由器需要 N 个以太网接口，同时也会占用了 N 个交换上的以太网接口。单臂路由提供另外一种解决方案。路由器只需要一个以太网接口和交换机连接，交换机的这个接口设置为 Trunk 接口。在路由器上创建多个子接口和不同的 VLAN 连接，子接口是路由器物理接口上的逻辑接口。工作原理如图 15-1，当交换机收到 VLAN1 的计算机发送的数据帧后，从它的 Trunk 接口发送数据给路由器，由于该链路是 Trunk 链路，帧中带有 VLAN1 的标签，帧到了路由器后，如果数据要转发到 VLAN2 上，路由器将把数据帧的 VLAN1 标签去掉，重新用 VLAN2 的标签进行封装，通过 Trunk 链路发送到交换机上的 Trunk 接口；交换机收到该帧，去掉 VLAN2 标签，发送给 VLAN2 上的计算机，从而实现了 VLAN 间的通信。

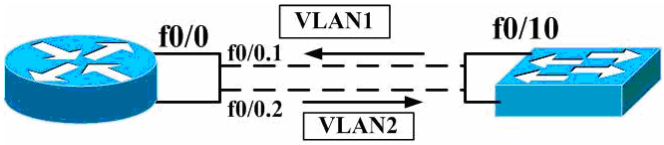

图 15-1　路由器的子接口工作原理

15.2 三层交换

单臂路由实现 VLAN 间的路由时转发速率较慢，实际上在局域网内部多采用三层交换。三层交换机通常采用硬件来实现，其路由数据包的速率是普通路由器的几十倍。

从使用者的角度可以把三层交换机看成是二层交换机和路由器的组合，如图 15-2，这个虚拟的路由器和每个 VLAN 都有一个接口进行连接，不过这个接口是 VLAN1 或 VLAN2 接口。Cisco 早些年采用的基于 NetFlow 的三层交换技术；现在 Cisco 主要采用 CEF 技术。CEF 技术中，交换机利用路由表形成转发信息库（FIB），FIB 和路由表是同步的，关键的是它的查询是硬件化，查询速度快得多。除了 FIB，还有邻接表(Adjacency Table)，该表和 ARP 表有些类似，主要放置了第二层的封装信息。FIB 和邻接表都是在数据转发之前就已经建立准备好了，这样一有数据要转发，交换机就能直接利用它们进行数据转发和封装，不需要查询路由表和发送 ARP 请求，所以 VLAN 间的路由速率大大提高。

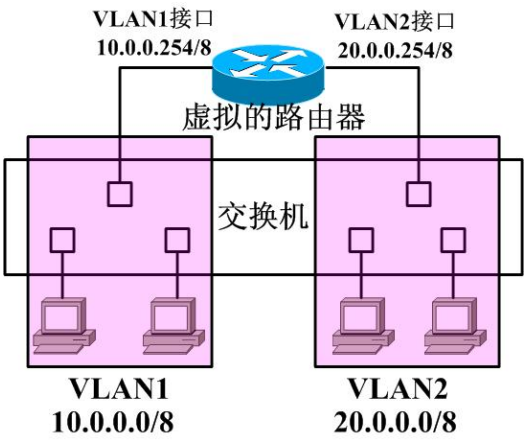

图 15-2 三层交换机原理示意图

15.2 实验 1：单臂路由实现 VLAN 间路由

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 路由器以太网接口上的子接口

（2） 单臂路由实现 VLAN 间路由的配置

2. 实验拓扑

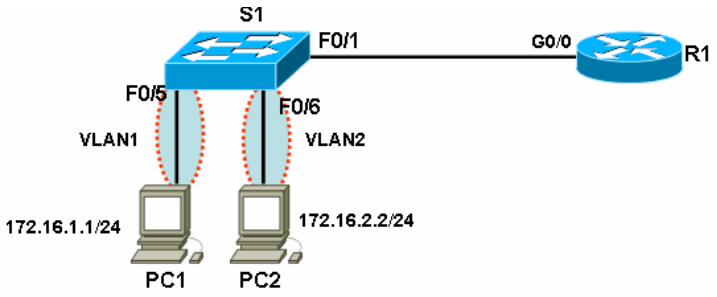

图 15-3 实验 1 拓扑图

3. 实验步骤

我们要用 R1 来实现分别处于 VLAN1 和 VLAN2 的 PC1 和 PC2 间的通信。

（1） 步骤 1：在 S1 上划分 VLAN

```
S1(config)#vlan 2
S1(config-vlan)#exit
S1(config)#int f0/5
S1(config-if)#switchport mode access
S1(config-if)#switchport access vlan 1
S1(config-if)#int f0/6
S1(config-if)#switchport mode access
S1(config-if)#switchport access vlan 2
```

（2） 步骤 2：要先把交换机上的以太网接口配置成 Trunk 接口

```
S1(config)#int f0/1
S1(config-if)#switch trunk encap dot1q
S1(config-if)#switch mode trunk
```

（3） 在路由器的物理以太网接口下创建子接口，并定义封装类型

```
R1(config)#int g0/0
R1(config-if)#no shutdown
R1(config)#int g0/0.1
R1(config-subif)#encapture dot1q 1 native
```

//以上是定义该子接口承载哪个 VLAN 流量，由于交换机上的 native vlan 是 VLAN 1，所以我们这里也要指明该 VLAN 就是 native vlan。实际上默认时 native vlan 就是 vlan 1。

```
R1 (config-subif)#ip address 172.16.1.254 255.255.255.0
```

//在子接口上配置 IP 地址，这个地址就是 VLAN 1 的网关了

```
R1(config)#int g0/0.2
R1(config-subif)#encapture dot1q 2
R1 (config-subif)#ip address 172.16.2.254 255.255.255.0
```

4. 实验调试

在 PC1 和 PC2 上配置 IP 地址和网关，PC1 的网关指向：17.16.1.254, PC1 的网关指向：17. 16.2.254。测试 PC1 和 PC2 的通信。注意：如果计算机有两个网卡，请去掉另一网卡上设置的网关。

【提示】S1 实际上是 catalyst 3560 交换机，该交换机具有三层功能，我们这里把它当作二层交换机使用了，有点大材小用。

15.2 实验 2：三层交换实现 VLAN 间路由

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解三层交换的概念

（2） 配置三层交换

2. 实验拓扑

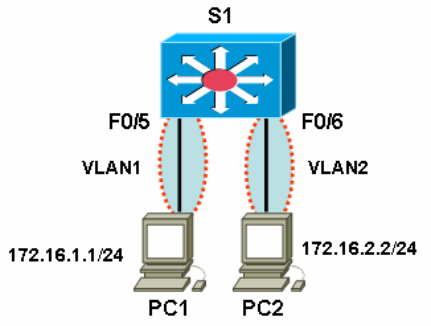

图 15-4 实验 2 拓扑图

3. 实验步骤

我们要用 S1 来实现分别处于 VLAN1 和 VLAN2 的 PC1 和 PC2 间的通信。

（1） 步骤 1：在 S1 上划分 VLAN

```
S1(config)#vlan 2
S1(config-vlan)#exit
S1(config)#int f0/5
S1(config-if)#switchport mode access
S1(config-if)#switchport access vlan 1
S1(config-if)#int f0/6
S1(config-if)#switchport mode access
S1(config-if)#switchport access vlan 2
```

（2） 步骤 2：配置三层交换

```
S1(config)#ip routing
```

//以上开启 S1 的路由功能，这时 S1 就启用了三层功能。

```
S1(config)#int vlan 1
S1(config-if)#no shutdown
S1(config-if)#ip address 172.16.1.254 255.255.255.0
S1(config)#int vlan 2
S1(config-if)#no shutdown
S1(config-if)#ip address 172.16.2.254 255.255.255.0
```

//在 vlan 接口上配置 IP 地址即可，VLAN 1 接口上的地址就是 PC1 的网关了，VLAN 2 接口上的地址就是 PC2 的网关了。

【提示】要在三层交换机上启用路由功能，还需要启用 CEF（命令为：ip cef），不过这是默认值。和路由器一样，三层交换机上同样可以运行路由协议。

4. 实验调试

（1） 检查 S1 上的路由表

```
S1#show ip route
```

（此处省略）

```
    172.16.0.0/24 is subnetted, 2 subnets
C       172.16.1.0 is directly connected, Vlan1
C       172.16.2.0 is directly connected, Vlan2
```

//和路由器一样，三层交换机上也有路由表。

（2） 测试 PC1 和 PC2 间的通信

在 PC1 和 PC2 上配置 IP 地址和网关，PC1 的网关指向：17.16.1.254, PC2 的网关指向：17. 16.2.254。测试 PC1 和 PC2 的通信。注意：如果计算机有两个网卡，请去掉另一网卡上设置的网关。

【提示】我们也可以把 f0/5 和 f0/6 接口作为路由接口使用，这时它们就和路由器的以太网接口一样了，可以在接口上配置 IP 地址。如果 S1 上的全部以太网都这样设置，S1 实际上成了具有 24 个以太网接口的路由器了，我们不建议这样做，这样太浪费接口了，。配置示例：

```
S1(config)#int f0/10
S1(config-if)#no switchport      //该接口不再是交换接口了，成为了路由接口
S1(config-if)#ip address 10.0.0.254  255.255.255.0
```

15.4 本章小结

本章介绍了实现不同 VLAN 间的计算机通信方法。可以使用单臂路由方法，在路由器的以太网接口上创建子接口。然而通常采用的是三层交换机来实现 VLAN 间的路由，三层交换机可以看成是交换机和路由器的集成，配置三层交换非常简单。表 15-1 是本章出现的命令。

表 15-1 本章命令汇总

命令	作用
int g0/0.1	创建子接口
encapture dot1q 1 native	指明子接口承载哪个VLAN的流量以及封装类型，同时该VLAN是native vlan
ip routing	打开路由功能
no switchport	接口不作为交换机接口
ip cef	开启 CEF 功能

第 16 章 网关冗余和负载平衡

为了减少交换机故障的影响，交换机上有 STP 技术。 然而作为网关的路由器故障了，又有什么办法？HSRP 和 VRRP 是最常用的网关冗余技术，HSRP 和 VRRP 类似，由多个路由器共同组成一个组，虚拟出一个网关，其中的一台路由器处于活动状态，当它故障时由备份路由器接替它的工作，从而实现对用户透明的切换。然而我们希望在冗余的同时，能同时实现负载平衡，以充分利用设备的能力，GLBP 同时提供了冗余和负载平衡的能力。本章将介绍它们的具体配置。

16.1 网关冗余和负载平衡简介

16.1.1 HSRP

HSRP 是 Cisco 的专有协议。HSRP（Hot Standby Router Protocol）把多台路由器组成一个“热备份组”，形成一个虚拟路由器。这个组内只有一个路由器是活动的（Active），并由它来转发数据包，如果活动路由器发生了故障，备份路由器将成为活动路由器。从网络内的主机来看，网关并没有改变。

HSRP 路由器利用 HELLO 包来互相监听各自的存在。当路由器长时间没有接收到 HELLO 包，就认为活动路由器故障，备份路由器就会成为活动路由器。HSRP 协议利用优先级决定哪个路由器成为活动路由器。如果一个路由器的优先级比其它路由器的优先级高，则该路由器成为活动路由器。路由器的缺省优先级是 100。 一个组中，最多有一个活动路由器和一个备份路由器。

HSRP 路由器发送的多播消息有以下三种：

（1） **HELLO**: HELLO 消息通知其它路由器发送路由器的 HSRP 优先级和状态信息，HSRP 路由器默认为每 3 秒钟发送一个 HELLO 消息；

（2） **Coup**：当一个备用路由器变为一个活动路由器时发送一个 coup 消息；

（3） **Resign**：当活动路由器要宕机或者当有优先级更高的路由器发送 HELLO 消息时，主动发送一个 resign 消息。

HSRP 路由器有以下六种状态：

（1） **Initial**：HSRP 启动时的状态，HSRP 还没有运行，一般是在改变配置或接口刚刚启动时进入该状态；

（2） **Learn**：路由器已经得到了虚拟 IP 地址，但是它既不是活动路由器也不是备份路由器。它一直监听从活动路由器和备份路由器发来的 HELLO 报文；

（3） **Listen**：路由器正在监听 HELLO 消息；

（4） **Speak**：在该状态下，路由器定期发送 HELLO 报文，并且积极参加活动路由器或备份路由器的竞选；

（5） **Standby**：当活动路由器失效时路由器准备接管数据传输功能；

（6） **Active**：路由器执行数据传输功能。

16.1.2 VRRP

VRRP 的工作原理和 HSRP 非常类似，不过 VRRP 是国际上的标准，允许在不同厂商的设备之间运行。VRRP 中虚拟网关的地址可以和接口上的地址相同，VRRP 中接口只有 3 个状态：初始状态(Initialize)、主状态(Master)、备份状态(Backup)。VRRP 有一种报文。

16.1.3 GLBP

HSRP和VRRP能实现网关的冗余，然而如果要实现负载平衡，需要创建多个组，并让客户端指向不同的网关。GLBP（Gateway Load Balance Protocol）也是Cisco的专有协议，不仅提供冗余网关功能，还在各网关之间提供负载均衡。GLBP也是由多个路由器组成一个组，虚拟一个网关出来。GLBP选举出一个AVG(Avtive Virtual Gateway)，AVG不是负责转发数据的。AVG分配最多四个MAC地址给一个虚拟网关，并在计算机进行ARP请求时，用不同的MAC进行响应，这样计算机实际就把数据发送给不同的路由器了，从而实现负载平衡。在GLBP中，真正负责转发数据的是AVF(Avtive Virtual Forwarder)，GLBP会控制GLBP组中哪个路由器是哪个MAC地址的活动路由器。

AVG的选举和HRSP中活动路由器的选举非常类似，优先级最高的路由器成为AVG，次之的为Backup AVG，其余的为监听状态。一个GLBP组只能有一个AVG和一个Backup AVG，主的AVG失败，备份AVG顶上。一台路由器可以同时是AVG和AVF。AVF是某些MAC的活动路由器，也就是说如果计算机把数据发往这个MAC，它将接收。当某一MAC的活动路由器故障，其它AVF将成为这一MAC的新的活动路由器，从而实现冗余功能。

GLBP 的负载平衡策略可以是根据不同主机、简单的轮询或者根据路由器的权重平衡，默认是轮询方式。

16.2 实验 1：HSRP

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 HSRP 的工作原理

（2） 掌握 HSRP 的配置

2. 实验拓扑

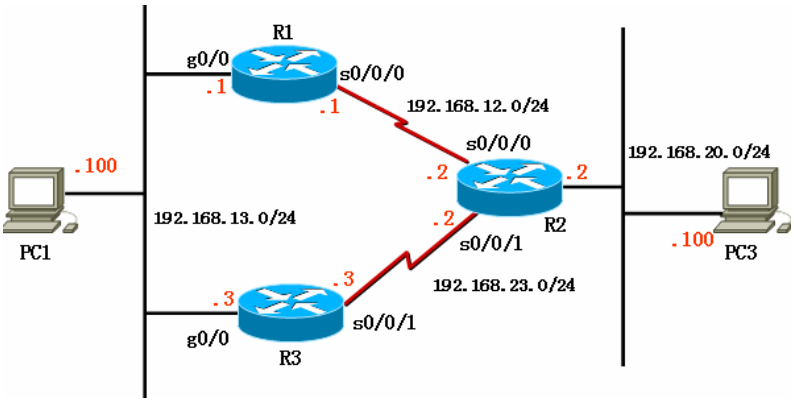

图 16-1 实验 1、实验 2 拓扑图

3. 实验步骤

（1） 步骤 1：配置 IP 地址、路由协议等

```
R1(config)#interface GigabitEthernet0/0
R1(config-if)#ip address 192.168.13.1 255.255.255.0
R1(config)#interface Serial0/0/0
```

```
R1(config-if)#ip address 192.168.12.1 255.255.255.0
R1(config)#router rip
R1(config-router)#network 192.168.12.0
R1(config-router)#network 192.168.13.0
R1(config-router)#passive-interface GigabitEthernet0/0
```
//之所以把 g0/0 接口设为被动接口，是防止从该接口发送 RIP 信息给 R3。

```
R2(config)#interface GigabitEthernet0/0
R2(config-if)#ip address 192.168.20.2 255.255.255.0
R2(config)#interface Serial0/0/0
R2(config-if)#clock rate 128000
R2(config-if)#ip address 192.168.12.2 255.255.255.0
R2(config)#interface Serial0/0/1
R2(config-if)#clock rate 128000
R2(config-if)#ip address 192.168.23.2 255.255.255.0
R2(config)#router rip
R2(config-router)#network 192.168.12.0
R2(config-router)#network 192.168.23.0
R2(config-router)#network 192.168.20.0
R2(config-router)#passive-interface GigabitEthernet0/0
```

```
R3(config)#interface GigabitEthernet0/0
R3(config-if)#ip address 192.168.13.3 255.255.255.0
R3(config)#interface Serial0/0/1
R3(config-if)#ip address 192.168.23.3 255.255.255.0
R3(config)#router rip
R3(config-router)#network 192.168.23.0
R3(config-router)#network 192.168.13.0
R3(config-router)#passive-interface GigabitEthernet0/0
```
（2） 步骤 2：配置 HSRP
```
R1(config)#interface g0/0
R1(config-if)#standby 1 ip 192.168.13.254
```
//启用 HSRP 功能，并设置虚拟 IP 地址，1 为 standby 的组号。相同组号的路由器属于同一个 HSRP 组，所有属于同一个 HSRP 组的路由器的虚拟地址必须一致
```
R1(config-if)#standby 1 priority 120
```
//配置 HSRP 的优先级，如果不设置该项，缺省优先级为 100，该值大抢占为活动路由器的优先权越高。
```
R1(config-if)#standby 1 preempt
```
//该设置允许该路由器在优先级是最高时成为活动路由器。如果不设置，即使该路由器权值再高，也不会成为活动路由器。
```
R1(config-if)#standby 1 timers 3 10
```
//其中 3 为 HELLO time，表示路由器每间隔多长时间发送 HELLO 信息。10 为 HOLD time，表示在多长时间内同组的其它路由器没有收到活动路由器的信息，则认为活动路由器故障。该设置的缺省值分别为 3 秒和 10 秒。如果要更改缺省值，所有同 HSRP 组的路由器的该项设

置必须一致。

```
R1(config-if)#standby 1 authentication md5 key-string cisco
```

//以上是配置认证密码，防止非法设备加入到 HSRP 组中，同一个组的密码必须一致。

```
R2(config)#interface g0/0
R2(config-if)#standby 1 ip 192.168.13.254
R2(config-if)#standby 1 preempt
R2(config-if)#standby 1 timers 3 10
R2(config-if)#standby 1 authentication md5 key-string cisco
```

//R2 上我们没有配置优先级，默认为 100。

（3） 步骤 3：检查、测试 HSRP

```
R1#show standby brief
                     P indicates configured to preempt.
                     |
Interface Grp  Pri P State   Active          Standby         Virtual IP
Gi0/0     1    120 P Active  local           192.168.13.3    192.168.13.254
```

//以上表明 R1 就是活动路由器，备份路由器为 192.168.13.3

```
R3#show standby brief
                     P indicates configured to preempt.
                     |
Interface Grp  Pri P State   Active          Standby         Virtual IP
Gi0/0     1    100 P Standby 192.168.13.1    local           192.168.13.254
```

//以上表明 R3 是备份路由器，活动路由器为 192.168.13.1

在 PC1 上配置 IP 地址 192.168.13.100/24，网关指向 192.168.13.254；在 PC3 上配置 IP 地址 192.168.20.100/24，网关指向 192.168.20.254。注意去掉另一网卡的网关。

在 PC1 上连续 ping PC3 上，在 R1 上关闭 g0/0 接口，观察 PC1 上 ping 的结果。如下：

```
C:\>ping -t 192.168.20.100
Reply from 192.168.20.100: bytes=32 time=9ms TTL=254
Reply from 192.168.20.100: bytes=32 time=9ms TTL=254
Reply from 192.168.20.100: bytes=32 time=9ms TTL=254
Request timed out.
Reply from 192.168.20.100: bytes=32 time=9ms TTL=254
Reply from 192.168.20.100: bytes=32 time=9ms TTL=254
Reply from 192.168.20.100: bytes=32 time=11ms TTL=254
Reply from 192.168.20.100: bytes=32 time=9ms TTL=254
```

//以上可以看到，R1 故障时，R3 很快就替代了 R1，计算机的通信只受到短暂的影响。

```
R3#show standby  brief
                     P indicates configured to preempt.
                     |
Interface Grp  Pri P State   Active          Standby         Virtual IP
Gi0/0     1    100 P Active  local           unknown         192.168.13.254
```

//以上表明 R3 成为了活动路由器了。

（4） 步骤 4：配置端口跟踪

图 16-1 中，按照以上步骤的配置，如果 R1 的 s0/0/0 接口出现问题，R1 将没有到达 PC3 所在网段的路由。然而 R1 和 R3 之间的以太网仍然没有问题，HSRP 的 HELLO 包正常发送和接收。因此 R1 仍然是虚拟网关 192.168.13.254 的活动路由器，PC1 的数据会发送给 R1，这样会造成 PC1 无法 ping 通 PC3。我们可以配置端口跟踪解决这个问题，端口跟踪使得 R1 发现 s0/0/0 上的链路出现问题后，把自己的优先级（我们设为了 120）减去一个数字（例如 30），成为了 90。由于 R3 的优先级为默认值 100，R3 就成为了活动路由器。配置如下：

R1(config)#int g0/0

R1(config-if)#standby 1 track s0/0/0 30

//以上表明跟踪的是 s0/0/0 接口，如果该接口故障，优先级降低 30。降低的值应该选取合适的值，使得其它路由器能成为活动路由器。按照步骤 3 测试 HSRP 的端口跟踪是否生效。

（5） 步骤 5：配置多个 HSRP 组

之前的步骤已经虚拟了 192.168.13.254 网关，对于这个网关只能有一个活动路由器，于是这个路由器将承担全部的数据流量。我们可以又创建一个 HSRP 组，虚拟出另一个网关 192.168.13.253，这时 R3 是活动路由器，让一部分计算机指向这个网关。这样就能做到负载平衡。以下是有 2 个 HSRP 组的完整配置：

R1 上：

```
interface GigabitEthernet0/0
     standby 1 ip 192.168.13.254
     standby 1 priority 120
     standby 1 preempt
     standby 1 authentication md5 key-string cisco
     standby 1 track Serial0/0/0 30
     standby 2 ip 192.168.13.253
     standby 2 preempt
     standby 2 authentication md5 key-string cisco
```

R3 上：

```
interface GigabitEthernet0/0
     standby 1 ip 192.168.13.254
     standby 1 preempt
     standby 1 authentication md5 key-string cisco
     standby 2 ip 192.168.13.253
     standby 2 priority 120
     standby 2 preempt
     standby 2 authentication md5 key-string cisco
     standby 2 track Serial0/0/0 30
```

【技术要点】我们这里是创建了两个 HSRP 组，第一个组的 IP 为 192.168.13.254，活动路由器为 R1，一部分计算机的网关指向 192.168.13.254。第二个组的 IP 为 192.168.13.253，活动路由器为 R2，另一部分计算机的网关指向 192.168.13.253。这样，如果网络全部正常时，一部分数据是 R1 转发的，另一部分数据是 R2 转发，实现了负载平衡。如果一个路由器出现问题，则另一个路由器就成为两个 HSRP 组的活动路由器，承担全部的数据转发功能。

通过这种方式实现负载平衡，需要计算机在设置网关时有所不同，如果计算机的 IP 是 DHCP 分配的，就不太方便。

【技术要点】HSRP 实际上在局域网用得较多，由于局域网内大多使用三层交换机，所以这时 HSRP 是在交换机上配置的。

16.3 实验 2：VRRP

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 VRRP 的工作原理

（2） 掌握 VRRP 的配置

2. 实验拓扑

如图 16-1。

3. 实验步骤

VRRP 的配置和 HSRP 的配置非常相同，不再赘述重复的步骤。

（1） 步骤 1：配置 IP 地址、路由协议等，参见实验 1

（2） 步骤 2：配置多个 VRRP 组，并跟踪接口

R1 上：

```
R1(config)#track 100 interface Serial0/0/0 line-protocol
R1(config)#interface GigabitEthernet0/0
R1(config-if)#vrrp 1 ip 192.168.13.254
R1(config-if)#vrrp 1 priority 120
R1(config-if)#vrrp 1 preempt
R1(config-if)#vrrp 1 authentication md5 key-string cisco
R1(config-if)#vrrp 1 track 100 decrement 30
R1(config-if)#vrrp 2 ip 192.168.13.253
R1(config-if)#vrrp 2 preempt
R1(config-if)#vrrp 2 authentication md5 key-string cisco
```

//VRRP 的端口跟踪和 HSRP 有些不同，需要在全局配置模式下先定义跟踪目标，才配置 vrrp 中跟踪该目标，我们这里定义了目标 100 是 s0/0/0 接口。

R3 上：

```
R3config)#track 100 interface Serial0/0/0 line-protocol
R3(config)#interface GigabitEthernet0/0
R3(config-if)#vrrp 1 ip 192.168.13.254
R3(config-if)#vrrp 1 preempt
R3(config-if)#vrrp 1 authentication md5 key-string cisco
R3(config-if)#vrrp 2 ip 192.168.13.253
R3(config-if)#vrrp 2 priority 120
R3(config-if)#vrrp 2 preempt
R3(config-if)#vrrp 2 authentication md5 key-string cisco
```

```
R3(config-if)#vrrp 2 track 100 decrement 30
```

```
R1#show vrrp brief
Interface          Grp Pri Time  Own Pre State   Master addr     Group addr
Gi0/0              1   120 3531        Y  Master  192.168.13.1    192.168.13.254
Gi0/0              2   100 3609        Y  Backup  192.168.13.3    192.168.13.253
```

//以上表明 R1 是 192.168.13.254 虚拟网关的 Master 路由器，是 192.168.13.253 虚拟网关的 Backup 路由器。

```
R3#show vrrp brief
Interface          Grp Pri Time  Own Pre State   Master addr     Group addr
Gi0/0              1   100 3609        Y  Backup  192.168.13.1    192.168.13.254
Gi0/0              2   120 3531        Y  Master  192.168.13.3    192.168.13.253
```

//以上表明 R3 是 192.168.13.253 虚拟网关的 Master 路由器，是 192.168.13.254 虚拟网关的 Backup 路由器。

（3） 步骤 3：检查、测试 HSRP，请参见实验 1。

16.4 实验 3：GLBP

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 GLBP 的工作原理

（2） 掌握 GLBP 的配置

2. 实验拓扑

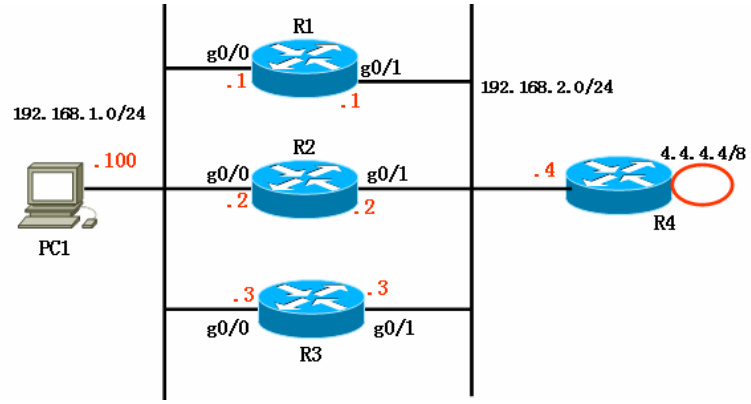

图 16-2 实验 3 拓扑

3. 实验步骤

（1） 步骤 1：配置 IP 地址、路由协议等

```
R1(config)#interface GigabitEthernet0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config)#interface GigabitEthernet0/1
R1(config-if)#ip address 192.168.2.1 255.255.255.0
```

```
R1(config)#router rip
R1(config-router)#network 192.168.1.0
R1(config-router)#network 192.168.2.0
R1(config-router)#passive-interface GigabitEthernet0/0

R2(config)#interface GigabitEthernet0/0
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config)#interface GigabitEthernet0/1
R2(config-if)#ip address 192.168.2.2 255.255.255.0
R2(config)#router rip
R2(config-router)#network 192.168.1.0
R2(config-router)#network 192.168.2.0
R2(config-router)#passive-interface GigabitEthernet0/0

R3(config)#interface GigabitEthernet0/0
R3(config-if)#ip address 192.168.1.3 255.255.255.0
R3(config)#interface GigabitEthernet0/1
R3(config-if)#ip address 192.168.2.3 255.255.255.0
R3(config)#router rip
R3(config-router)#network 192.168.1.0
R3(config-router)#network 192.168.2.0
R3(config-router)#passive-interface GigabitEthernet0/0

R4(config)#interface Loopback0
R4(config-if)#ip address 4.4.4.4 255.0.0.0
R4(config)#interface GigabitEthernet0/1
R4(config-if)#ip address 192.168.2.4 255.255.255.0
R4(config)#router rip
R4(config-router)#network 4.0.0.0
R4(config-router)#network 192.168.2.0
```

（2） 步骤 2：配置 GLBP

```
R1(config)#interface GigabitEthernet0/0
R1(config-if)#glbp 1 ip 192.168.1.254
//和 HSRP 类似，创建 GLBP 组，虚拟网关的 IP 为 192.168.1.254
R1(config-if)#glbp 1 priority 200
//配置优先级，优先级高的路由器成为 AVG，默认为 100
R1(config-if)#glbp 1 preempt
//配置 AVG 抢占，否则即使优先级再高，也不会成为 AVG
R1(config-if)#glbp 1 authentication md5 key-string cisco
//以上是配置认证，防止非法设备接入

R2(config)#interface GigabitEthernet0/0
R2(config-if)#glbp 1 ip 192.168.1.254
R2(config-if)#glbp 1 priority 180
```

```
R2(config-if)#glbp 1 preempt
R2(config-if)#glbp 1 authentication md5 key-string cisco

R3(config)#interface GigabitEthernet0/0
R3(config-if)#glbp 1 ip 192.168.1.254
R3(config-if)#glbp 1 priority 160
R3(config-if)#glbp 1 preempt
R3(config-if)#glbp 1 authentication md5 key-string cisco
```

（3） 步骤 3：查看 GLBP 信息

```
R1#show glbp
GigabitEthernet0/0 - Group 1
  State is Active
    4 state changes, last state change 00:18:16
  Virtual IP address is 192.168.1.254
```

//以上是虚拟的网关 IP 地址

```
  HELLO time 3 sec, hold time 10 sec
    Next HELLO sent in 1.896 secs
  Redirect time 600 sec, forwarder time-out 14400 sec
  Authentication MD5, key-string "cisco"
  Preemption enabled, min delay 0 sec
  Active is local
```

//以上说明 R1 是活动 AVG

```
  Standby is 192.168.1.2, priority 180 (expires in 9.892 sec)
```

//以上说明 R2 是备份 AVG

```
  Priority 200 (configured)
  Weighting 100 (default 100), thresholds: lower 1, upper 100
  Load balancing: round-robin
  Group members:
    0019.5535.b548 (192.168.1.3) authenticated
    0019.5535.b828 (192.168.1.1) local
    0019.5566.6320 (192.168.1.2) authenticated
```

//以上显示 GLBP 组中的成员

```
  There are 3 forwarders (1 active)
  Forwarder 1
    State is Listen
      4 state changes, last state change 00:17:08
    MAC address is 0007.b400.0101 (learnt)
```

//这是虚拟网关的其中一个 MAC

```
    Owner ID is 0019.5535.b548
    Redirection enabled, 599.984 sec remaining (maximum 600 sec)
    Time to live: 14399.984 sec (maximum 14400 sec)
    Preemption enabled, min delay 30 sec
    Active is 192.168.1.3 (primary), weighting 100 (expires in 9.984 sec)
    Client selection count: 1
```

```
  Forwarder 2
    State is Active
      3 state changes, last state change 00:18:28
    MAC address is 0007.b400.0102 (default)
```
//以上说明 R1 是 0007.b400.0102 的活动路由器，也就是说如果计算机把数据发往 0007.b400.0102，将由 R1 接收数据，再进行转发。
```
    Owner ID is 0019.5535.b828
    Redirection enabled
    Preemption enabled, min delay 30 sec
    Active is local, weighting 100
    Client selection count: 1
  Forwarder 3
    State is Listen
      2 state changes, last state change 00:18:06
    MAC address is 0007.b400.0103 (learnt)
    Owner ID is 0019.5566.6320
    Redirection enabled, 597.980 sec remaining (maximum 600 sec)
    Time to live: 14397.980 sec (maximum 14400 sec)
    Preemption enabled, min delay 30 sec
    Active is 192.168.1.2 (primary), weighting 100 (expires in 7.980 sec
```

通过查看，可以知道：
R1：0007.b400.0102 的活动路由器
R2：0007.b400.0103 的活动路由器
R3：0007.b400.0101 的活动路由器
（4） 步骤 4：检查 GLBP 的负载平衡功能
在 PC1 上配置 IP 地址，网关指向 192.168.1.254。并进行如下操作：
C:\>**ping 4.4.4.4**
C:\>**arp -a**
```
Interface: 192.168.1.100 --- 0x10006
  Internet Address      Physical Address      Type
  192.168.1.254         00-07-b4-00-01-01     dynamic
```
以上表明 PC1 的 ARP 请求获得网关(192.168.1.254)的 MAC 为 **00-07-b4-00-01-01**。
C:\>**arp -d**
//以上是删除 ARP 缓冲表
C:\>**ping 4.4.4.4**
C:\>**arp -a**
```
Interface: 192.168.1.100 --- 0x10006
  Internet Address      Physical Address      Type
  192.168.1.254         00-07-b4-00-01-02     dynamic
```
以上表明 PC1 的再次 ARP 请求获得网关(192.168.1.254)的 MAC 为 **00-07-b4-00-01-02** 了，也就是说 GLBP 响应 ARP 请求时，每次会用不同的 MAC 响应，从而实现负载平衡。

【提示】默认时 GLBP 的负载平衡策略是轮询方式，可以在接口下使用“**glbp 1**

load-balancing”命令修改，有以下选项：

- host-dependent：根据不同主机的源 MAC 地址进行平衡
- round-robin：轮询方式，即每响应一次 ARP 请求，轮换一个地址
- weighted：根据路由器的权重分配，权重高的被分配的可能性越大。

（5） 步骤 5：检查 GLBP 的冗余功能

首先在 PC1 上用“arp -a”命令确认 192.168.1.254 的 MAC 地址是什么，从而确定出当前究竟是哪个路由器在实际转发数据。我们这里 192.168.1.254 的 MAC 地址为 00-07-b4-00-01-02，从步骤 3 得知是 R1 在转发数据。

在 PC1 上连续 ping 4.4.4.4，并在 R1 上关闭 g0/0 接口，观察 PC1 的通信情况：

```
C:\ >ping -t 4.4.4.4
Reply from 4.4.4.4: bytes=32 time<1ms TTL=254
Reply from 4.4.4.4: bytes=32 time<1ms TTL=254
Request timed out.
Request timed out.
Reply from 4.4.4.4: bytes=32 time<1ms TTL=254
Reply from 4.4.4.4: bytes=32 time<1ms TTL=254
```

//可以看到在 R1 故障后，其它路由器很快接替了它的工作，计算机的通信只受到短暂的影响。因此 GLBP 不仅有负载平衡的能力，也有冗余的能力。可以使用“show glbp”命令查看一下谁是 00-07-b4-00-01-02 这个 MAC 的新的活动路由器。

16.5 本章小结

本章介绍了 HSRP 和 VRRP 的目的和基本工作原理。HSRP 和 VRRP 都是为了实现网关的冗余，它们把多个路由器组成一个小组，选出活动路由器，当它故障时，其它路由器接替它的工作。GLBP 则不仅具有网络冗余功能，还可以提供负载平衡的功能。本章详细介绍了它们的配置。表 16-1 是本章出现的命令。

表 16-1 本章命令汇总

命令	作用
standby 1 ip 192.168.13.254	启用 HSRP 功能，并设置虚拟 IP 地址
standby 1 priority 120	配置本路由器的 HSRP 优先级
standby 1 preempt	配置 HSRP 抢占
standby 1 timers 3 10	设置 HSRP 的 HELLO time 和 HOLD time
standby 1 authentication md5 key-string cisco	配置 HSRP 认证密码，认证方式为 MD5
show standby brief	查看 HSRP 的简要情况
standby 1 track Serial0/0/0 30	跟踪 s0/0/0 接口，当接口故障时，HSRP 优先级降低 30
vrrp 1 ip 192.168.13.254	启用 VRRP 功能，并设置虚拟 IP 地址
vrrp 1 priority 120	配置本路由器的 VRRP 优先级
vrrp 1 preempt	配置 VRRP 抢占
vrrp 1 authentication md5 key-string cisco	配置 VRRP 认证密码，认证方式为 MD5
track 100 interface Serial0/0/0	定义一个跟踪目标号，被跟踪对象为 s0/0/0

line-protocol	接口
vrrp 1 track 100 decrement 30	跟踪目标 100，当目标故障时，优先级降低 30
show vrrp brief	查看 VRRP 的简要情况
glbp 1 ip 192.168.1.254	启用 GLBP 功能，并设置虚拟 IP 地址
glbp 1 priority 200	配置本路由器的 GLBP 优先级
glbp 1 preempt	配置 GLBP 抢占
glbp 1 authentication md5 key-string cisco	配置 GLBP 认证密码，认证方式为 MD5
show glbp	查看 GLBP 情况

第 17 章　帧中继上的 OSPF

帧中继是典型的 NBMA（NonBroadcast Multiple Access）网络，其拓扑结构通常有两种：Full Mesh(全互联)和 Hub-and-Spoke（中心－分支）。由于 Hub-and-Spoke 结构具有节约费用、简化配置等优点，在实际网络工程中有着广泛的应用。本章重点讨论的就是在 Hub-and-Spoke 结构上，网络类型为 NBMA 模式、广播模式、点到点模式和点到多点模式的 OSPF 配置。

17.1　实验 1：帧中继环境下 NBMA 模式

1. 实验目的

通过本实验可以掌握：

（1）帧中继静态映射及 broadcast 参数的含义

（2）NBMA 模式下的 DR 选举

（3）手工配置 OSPF 邻居

（4）NBMA 模式下 OSPF 的配置和调试

2. 拓扑结构

实验拓扑如图 17-1 所示。

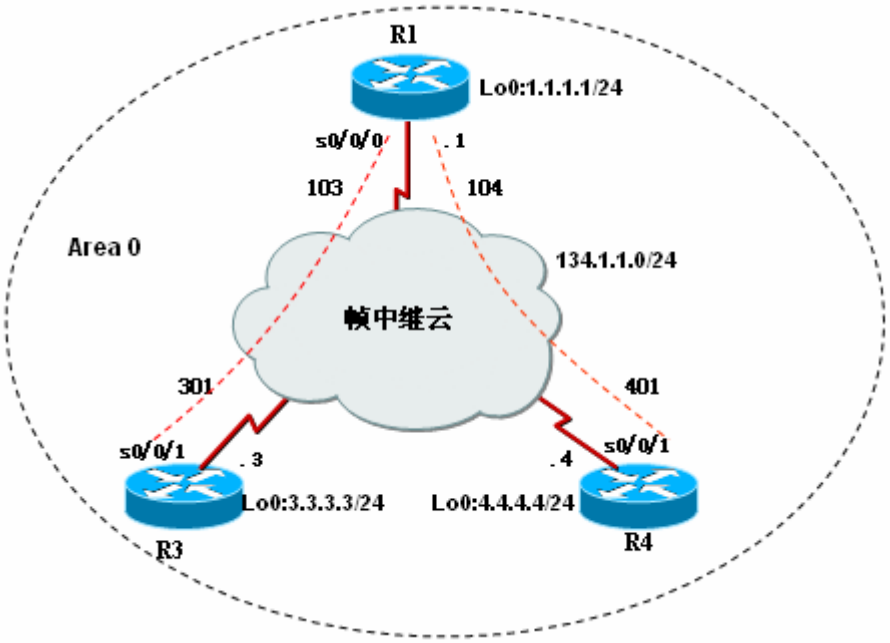

图 17-1 帧中继环境下 NBMA 模式

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-if)#ip ospf network point-to-point
R1(config-if)#interface Serial0/0/0
R1(config-if)#ip address 134.1.1.1 255.255.255.0
R1(config-if)#encapsulation frame-relay
```

```
R1(config-if)#frame-relay map ip 134.1.1.3 103 broadcast//帧中继静态映射
R1(config-if)#frame-relay map ip 134.1.1.4 104 broadcast
R1(config-if)#frame-relay map ip 134.1.1.1 103 //使得可以 PING 通自己
R1(config-if)#no frame-relay inverse-arp       //关闭帧中继动态 ARP 解析
R1(config-if)#no shutdown
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
R1(config-router)#network 134.1.1.0 0.0.0.255 area 0
R1(config-router)#neighbor 134.1.1.3     //手工指 OSPF 邻居
R1(config-router)#neighbor 134.1.1.4
```

（2）步骤 2：配置路由器 R3

```
R3(config)#interface Loopback0
R3(config-if)#ip address 3.3.3.3 255.255.255.0
R3(config-if)#ip ospf network point-to-point
R3(config-if)#interface Serial0/0/1
R3(config-if)#ip address 134.1.1.3 255.255.255.0
R3(config-if)#encapsulation frame-relay
R3(config-if)#ip ospf priority 0  // 配置 spoke 端 OSPF 接口优先级为 0
R3(config-if)#frame-relay map ip 134.1.1.1 301 broadcast
R3(config-if)#frame-relay map ip 134.1.1.4 301 broadcast
R3(config-if)#frame-relay map ip 134.1.1.3 301
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#no shutdown
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 134.1.1.0 0.0.0.255 area 0
```

（3）步骤 3：配置路由器 R4

```
R4(config)#interface Loopback0
R4(config-if)#ip address 4.4.4.4 255.255.255.0
R4(config-if)#ip ospf network point-to-point
R4(config-if)#interface Serial0/0/1
R4(config-if)#ip address 134.1.1.4 255.255.255.0
R4(config-if)#encapsulation frame-relay
R4(config-if)#ip ospf priority 0
R4(config-if)#frame-relay map ip 134.1.1.1 401 broadcast
R4(config-if)#frame-relay map ip 134.1.1.3 401 broadcast
R4(config-if)#frame-relay map ip 134.1.1.4 401
R4(config-if)#no frame-relay inverse-arp
R4(config-if)#no shutdown
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 0.0.0.255 area 0
```

R4(config-router)#network 134.1.1.0 0.0.0.255 area 0

【技术要点】

（1）在帧中继网络上，OSPF 接口缺省的网络类型为 NON_BROADCAST。在这种模式下，OSPF 不会在帧中继接口上发送 Hello 包，因此无法建立最基本的邻接关系。可以手工使用“neighbor”命令来指定邻居，这时 Hello 包以单播形式传送；

（2）NBMA 属于多路访问网络，所以要进行 DR 选举。由于 Hello 包只能传 1 跳，所以在 Hub-and-Spoke 结构中，必须控制处于“Hub”端的路由器为 DR,最保险的办法就是将“Spoke”端接口优先级配置为 0，使之不参与 DR 选举，“Hub”端的路由器自然就成为 DR。否则，可能会导致路由学习不正常。

4.实验调试

（1）show ip ospf interface

```
R1#show ip ospf interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 134.1.1.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type NON_BROADCAST, Cost: 64
//接口网络类型为 NBMA 模式
  Transmit Delay is 1 sec, State DR, Priority 1
//自己是 DR,接口优先级为 1
  Designated Router (ID) 1.1.1.1, Interface address 134.1.1.1
//DR 的 ID 和接口地址
  No backup designated router on this network
//没有 BDR
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
//NBMA 模式下，Hello 周期为 30 秒
    oob-resync timeout 120
    Hello due in 00:00:22
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 3.3.3.3
    Adjacent with neighbor 4.4.4.4
//与路由器 R3 和 R4 形成邻接关系
  Suppress hello for 0 neighbor(s)
```

（2）show ip route

```
R3#show ip route ospf
     1.0.0.0/24 is subnetted, 1 subnets
O       1.1.1.0 [110/65] via 134.1.1.1, 00:01:47, Serial0/0/0
     4.0.0.0/24 is subnetted, 1 subnets
O       4.4.4.0 [110/65] via 134.1.1.4, 00:01:47, Serial0/0/0
```

从以上输出表明，到达网络“4.4.4.0/24”的路由条目的下一跳地址为“134.1.1.4”，而不是“134.1.1.1”，所以，在 R3 的 s0/0/1 的接口上必须有到 134.1.1.4 的映射“frame-relay map ip 134.1.1.4 301 broadcast”。

（3）show ip ospf neighbor detail

```
R1#show ip ospf neighbor detail
 Neighbor 3.3.3.3, interface address 134.1.1.3
    In the area 0 via interface Serial0/0/0
    Neighbor priority is 0, State is FULL, 9 state changes
    DR is 134.1.1.1 BDR is 0.0.0.0
    Poll interval 120
    Options is 0x52
    LLS Options is 0x1 (LR)
    Dead timer due in 00:01:53
    Neighbor is up for 00:06:54
    Index 1/1, retransmission queue length 0, number of retransmission 1
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
    Last retransmission scan length is 1, maximum is 1
    Last retransmission scan time is 0 msec, maximum is 0 msec
 Neighbor 4.4.4.4, interface address 134.1.1.4
    In the area 0 via interface Serial0/0/0
    Neighbor priority is 0, State is FULL, 9 state changes
    DR is 134.1.1.1 BDR is 0.0.0.0
    Poll interval 120
    Options is 0x52
    LLS Options is 0x1 (LR)
    Dead timer due in 00:01:43
    Neighbor is up for 00:06:54
    Index 2/2, retransmission queue length 0, number of retransmission 1
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
    Last retransmission scan length is 1, maximum is 1
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

以上输出表明 R1 的两个邻居的接口优先级为 0。同时本网络的 BDR 为 0.0.0.0，这是可以的。

17.2 实验 2：帧中继环境下 BMA 模式

1.实验目的

通过本实验可以掌握：

（1）帧中继静态映射及 broadcast 参数的含义

（2）BMA 模式下的 DR 选举

（3）BMA 模式下 OSPF 的配置和调试

2.拓扑结构

实验拓扑如图 17-1 所示。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-if)#ip ospf network point-to-point
R1(config-if)#interface Serial0/0/0
R1(config-if)#ip address 134.1.1.1 255.255.255.0
R1(config-if)#encapsulation frame-relay
R1(config-if)#frame-relay map ip 134.1.1.3 103 broadcast
R1(config-if)#frame-relay map ip 134.1.1.4 104 broadcast
R1(config-if)#frame-relay map ip 134.1.1.1 103
R1(config-if)#no frame-relay inverse-arp
R1(config-if)#ip ospf network broadcast
R1(config-if)#no shutdown
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
R1(config-router)#network 134.1.1.0 0.0.0.255 area 0
```

（2）步骤 2：配置路由器 R3

```
R3(config)#interface Loopback0
R3(config-if)#ip address 3.3.3.3 255.255.255.0
R3(config-if)#ip ospf network point-to-point
R3(config-if)#interface Serial0/0/1
R3(config-if)#ip address 134.1.1.3 255.255.255.0
R3(config-if)#encapsulation frame-relay
R3(config-if)#ip ospf priority 0
R3(config-if)#frame-relay map ip 134.1.1.1 301 broadcast
R3(config-if)#frame-relay map ip 134.1.1.4 301 broadcast
R3(config-if)#frame-relay map ip 134.1.1.3 301
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#ip ospf network broadcast
R3(config-if)#no shutdown
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 134.1.1.0 0.0.0.255 area 0
```

（3）步骤 3：配置路由器 R4

```
R4(config)#interface Loopback0
R4(config-if)#ip address 4.4.4.4 255.255.255.0
R4(config-if)#ip ospf network point-to-point
R4(config-if)#interface Serial0/0/1
R4(config-if)#ip address 134.1.1.4 255.255.255.0
R4(config-if)#encapsulation frame-relay
R4(config-if)#ip ospf priority 0
R4(config-if)#frame-relay map ip 134.1.1.1 401 broadcast
R4(config-if)#frame-relay map ip 134.1.1.3 401 broadcast
```

```
R4(config-if)#frame-relay map ip 134.1.1.4 401
R4(config-if)#no frame-relay inverse-arp
R4(config-if)#ip ospf network broadcast
R4(config-if)#no shutdown
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 0.0.0.255 area 0
R4(config-router)#network 134.1.1.0 0.0.0.255 area 0
```

【技术要点】

（1）在 Hub-and-Spoke 结构中，BMA 也要控制 DR 选举，确保处于“Hub”端的路由器为 DR，实施方法和实验 1 一样；

（2）BMA 模式下，邻居关系自动通过 Hello 包建立和维持。

4.实验调试

（1）show ip ospf interface

```
R1#show ip ospf interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Internet Address 134.1.1.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type BROADCAST, Cost: 64
//网络类型为 BROADCAST
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 1.1.1.1, Interface address 134.1.1.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
//BMA 模式下，Hello 周期为 10 秒
    oob-resync timeout 40
    Hello due in 00:00:07
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 3.3.3.3
    Adjacent with neighbor 4.4.4.4
  Suppress hello for 0 neighbor(s)
```

（2）show ip route

```
R4#show ip route ospf
     1.0.0.0/24 is subnetted, 1 subnets
O       1.1.1.0 [110/65] via 134.1.1.1, 00:03:19, Serial0/0/1
     3.0.0.0/24 is subnetted, 1 subnets
O       3.3.3.0 [110/65] via 134.1.1.3, 00:03:19, Serial0/0/1
```

17.3 实验 3：帧中继环境下点到点模式

1. 实验目的

（1）帧中继子接口下静态映射

（2）点到点模式的特征

（2）点到点模式下 OSPF 的配置和调试

2. 拓扑结构

实验拓扑如图 17-2 所示。

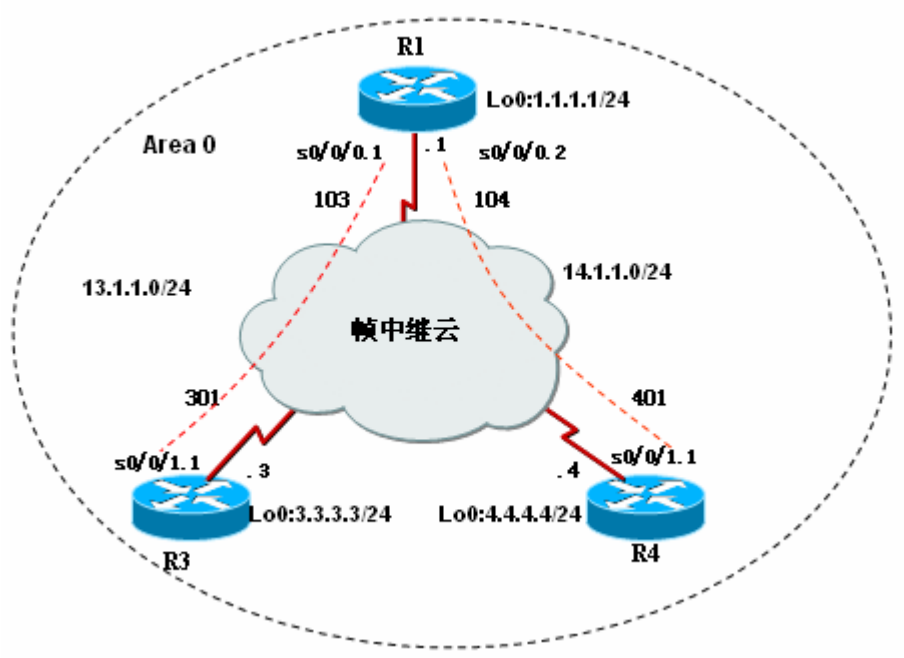

图 17-2 帧中继环境下点到点模式

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-if)#ip ospf network point-to-point
R1(config)#interface Serial0/0/0
R1(config-if)#no ip address
R1(config-if)#encapsulation frame-relay
R1(config-if)#no frame-relay inverse-arp
R1(config-if)#no shutdown
R1(config)#interface Serial0/0/0.1 point-to-point
R1(config-subif)#ip address 13.1.1.1 255.255.255.0
R1(config-subif)#frame-relay interface-dlci 103
R1(config)#interface Serial0/0/0.2 point-to-point
R1(config-subif)#ip address 14.1.1.1 255.255.255.0
R1(config-subif)#frame-relay interface-dlci 104
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
```

R1(config-router)#network 13.1.1.0 0.0.0.255 area 0
R1(config-router)#network 14.1.1.0 0.0.0.255 area 0

（2）步骤 2：配置路由器 R3

R3(config)#interface Serial0/0/1
R3(config-if)#no ip address
R3(config-if)#encapsulation frame-relay
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#no shutdown
R3(config)#interface Serial0/0/1.1 point-to-point
R3(config-subif)#ip address 13.1.1.3 255.255.255.0
R3(config-subif)#frame-relay interface-dlci 301
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 13.1.1.0 0.0.0.255 area 0

（3）步骤 3：配置路由器 R4

R4(config)#interface Serial0/0/1
R4(config-if)#no ip address
R4(config-if)#encapsulation frame-relay
R4(config-if)#no frame-relay inverse-arp
R4(config-if)#no shutdown
R4(config)#interface Serial0/0/1.1 point-to-point
R4(config-subif)#ip address 14.1.1.4 255.255.255.0
R4(config-subif)#frame-relay interface-dlci 401
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 0.0.0.255 area 0
R4(config-router)#network 14.1.1.0 0.0.0.255 area 0

4.实验调试

（1）show ip ospf interface

R1#show ip ospf interface s0/0/0.1
Serial0/0/0.1 is up, line protocol is up
  Internet Address 13.1.1.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type POINT_TO_POINT, Cost: 64
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
//POINT_TO_POINT 模式下，Hello 周期为 10 秒
oob-resync timeout 40
    Hello due in 00:00:09
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 4 msec
  Neighbor Count is 1, Adjacent neighbor count is 1

```
   Adjacent with neighbor 3.3.3.3
 Suppress hello for 0 neighbor(s)
```

（2）**show ip ospf neighbor detail**

```
R1#show ip ospf neighbor detail
Neighbor 4.4.4.4, interface address 14.1.1.4
   In the area 0 via interface Serial0/0/0.2
   Neighbor priority is 0, State is FULL, 6 state changes
   DR is 0.0.0.0 BDR is 0.0.0.0
   Options is 0x52
   LLS Options is 0x1 (LR)
   Dead timer due in 00:00:34
   Neighbor is up for 00:07:21
   Index 2/2, retransmission queue length 0, number of retransmission 1
   First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
   Last retransmission scan length is 1, maximum is 1
   Last retransmission scan time is 0 msec, maximum is 0 msec
Neighbor 3.3.3.3, interface address 13.1.1.3
   In the area 0 via interface Serial0/0/0.1
   Neighbor priority is 0, State is FULL, 6 state changes
   DR is 0.0.0.0 BDR is 0.0.0.0
   Options is 0x52
   LLS Options is 0x1 (LR)
   Dead timer due in 00:00:32
   Neighbor is up for 00:08:51
   Index 1/1, retransmission queue length 0, number of retransmission 1
   First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
   Last retransmission scan length is 1, maximum is 1
   Last retransmission scan time is 0 msec, maximum is 0 msec
```

以上输出表明路由器 R1 通过两个子接口分别与路由器 R3 和 R4 建立邻接关系。

【技术要点】

① 点到点模式的 DR 和 BDR 是“0.0.0.0”；

② 点到点模式下，每个子接口需要配置不同的网络；

③ 点到点模式下，Hello 周期为 10 秒。

17.4 实验 4：帧中继环境下点到多点模式

1.实验目的

（1）帧中继子接口下静态映射

（2）点到多点模式的特征

（3）点到多点模式下 OSPF 的配置和调试

2.拓扑结构

实验拓扑如图 17-3 所示。

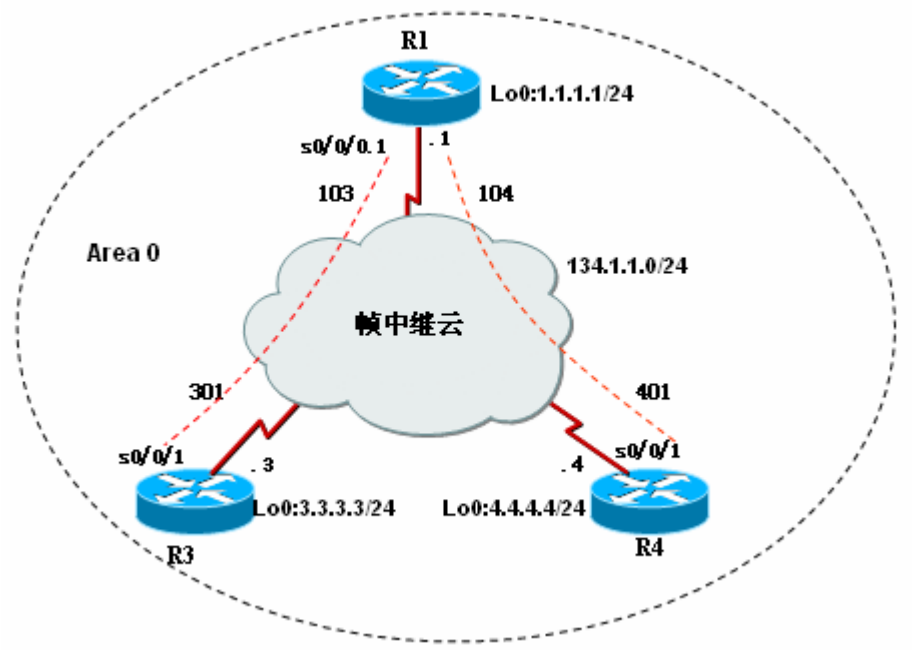

图 17-3 帧中继环境下点到多点模式

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#interface Serial0/0/0
R1(config-if)#no ip address
R1(config-if)#encapsulation frame-relay
R1(config-if)#no frame-relay inverse-arp
R1(config-if)#no shutdown
R1(config)#interface Serial0/0/0.1 multipoint
R1(config-subif)#ip address 134.1.1.1 255.255.255.0
R1(config-subif)#ip ospf network point-to-multipoint
R1(config-subif)#frame-relay map ip 134.1.1.3 103 broadcast
R1(config-subif)#frame-relay map ip 134.1.1.4 104 broadcast
R1(config-subif)#no frame-relay inverse-arp
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
R1(config-router)#network 134.1.1.0 0.0.0.255 area 0
```

（2）步骤 2：配置路由器 R3

```
R3(config)#interface Serial0/0/1
R3(config-if)#ip address 134.1.1.3 255.255.255.0
R3(config-if)#encapsulation frame-relay
R3(config-if)#ip ospf network point-to-multipoint
R3(config-if)#frame-relay map ip 134.1.1.1 301 broadcast
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#no shutdown
R3(config)#router ospf 1
```

```
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 134.1.1.0 0.0.0.255 area 0
```

（3）步骤 3：配置路由器 R4

```
R3(config)#interface Serial0/0/1
R3(config-if)#ip address 134.1.1.4 255.255.255.0
R3(config-if)#encapsulation frame-relay
R3(config-if)#ip ospf network point-to-multipoint
R3(config-if)#frame-relay map ip 134.1.1.1 401 broadcast
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#no shutdown
R3(config)#router ospf 1
R3(config-router)#router-id 4.4.4.4
R3(config-router)#network 4.4.4.0 0.0.0.255 area 0
R3(config-router)#network 134.1.1.0 0.0.0.255 area 0
```

4.实验调试

（1）show ip ospf interface

```
R1#show ip ospf interface s0/0/0.1
Serial0/0/0.1 is up, line protocol is up
  Internet Address 134.1.1.1/24, Area 0
  Process ID 1, Router ID 1.1.1.1, Network Type POINT_TO_MULTIPOINT, Cost: 64
  Transmit Delay is 1 sec, State POINT_TO_MULTIPOINT,
  Timer intervals configured, Hello 30, Dead 120, Wait 120, Retransmit 5
// POINT_TO_MULTIPOINT 模式下，Hello 周期为 30 秒
    oob-resync timeout 120
    Hello due in 00:00:00
  Index 2/2, flood queue length 0
  Next 0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 2, Adjacent neighbor count is 2
    Adjacent with neighbor 4.4.4.4
    Adjacent with neighbor 3.3.3.3
  Suppress hello for 0 neighbor(s)
```

（2）show ip route

```
R1#show ip route ospf
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/65] via 134.1.1.3, 00:02:11, Serial0/0/0.1
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/65] via 134.1.1.4, 00:02:11, Serial0/0/0.1
     134.1.0.0/16 is variably subnetted, 3 subnets, 2 masks
O       134.1.1.4/32 [110/64] via 134.1.1.4, 00:02:11, Serial0/0/0.1
O       134.1.1.3/32 [110/64] via 134.1.1.3, 00:02:11, Serial0/0/0.1
```

```
R3#show ip route ospf
      1.0.0.0/32 is subnetted, 1 subnets
O        1.1.1.1 [110/65] via 134.1.1.1, 00:03:41, Serial0/0/1
     4.0.0.0/32 is subnetted, 1 subnets
O        4.4.4.4 [110/129] via 134.1.1.1, 00:03:41, Serial0/0/1
     134.1.0.0/16 is variably subnetted, 3 subnets, 2 masks
O        134.1.1.4/32 [110/128] via 134.1.1.1, 00:03:41, Serial0/0/1
O        134.1.1.1/32 [110/64] via 134.1.1.1, 00:03:41, Serial0/0/1
```

以上输出表明在点到多点模式中，在路由表中会产生该网段其他各个接口的主机路由，因此在做帧中继映射的时候，只做到中心点的就可以了。

【技术要点】

（1）点到多点广播模式可以被看成多个点到点接口的集合，然而和点到点不同的是帧中继接口是在同一子网上；

（2）在点到多点模式中，不需要选举 DR/BDR；

（3）Hello 包每 30 秒发送一次，无需手工配置邻居。

17.5 帧中继上的 OSPF 命令汇总

表 17-1 列出了本章涉及到的主要的命令。

表 17-1 本章命令汇总

命令	作用
show ip route	查看路由表
show ip ospf interface	查看运行 OSPF 的接口的相关信息
show ip ospf neighbor detail	查看 OSPF 邻居路由器的详细信息
ip ospf network	配置 OSPF 网络类型
encapsulation frame-relay	接口封装帧中继
no frame-relay inverse-arp	关闭帧中继逆向 ARP 解析
frame-relay interface-dlci	帧中继映射
frame-relay map ip	帧中继映射

第 18 章 多区域 OSPF

在一个大型 OSPF 网络中，SPF 算法的反复计算，庞大的路由表和拓扑表的维护以及 LSA 的泛洪等都会占用路由器的资源，因而会降低路由器的运行效率。OSPF 协议可以利用区域的概念来减小这些不利的影响。因为在一个区域内的路由器将不需要了解它们所在区域外的拓扑细节。OSPF 多区域的拓扑结构有如下的优势：

1. 降低 SPF 计算频率
2. 减小路由表
3. 降低了通告 LSA 的开销
4. 将不稳定限制在特定的区域

18.1 多区域 OSPF 概述

18.1.1 OSPF 路由器类型

当一个 AS 划分成几个 OSPF 区域时，根据一个路由器在相应的区域之内的作用，可以将 OSPF 路由器作如下分类，如图 18-1 所示。

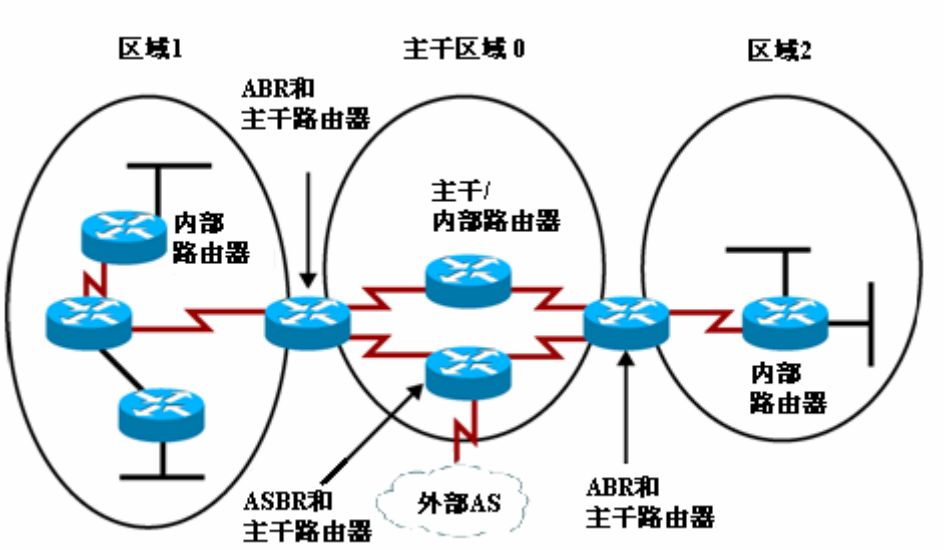

图 18-1 OSPF 路由器类型

1. 内部路由器：OSPF 路由器上所有直连的链路都处于同一个区域；
2. 主干路由器：具有连接区域 0 接口的路由器；
3. 区域边界路由器(ABR)：路由器与多个区域相连；
4. 自治系统边界路由器(ASBR)：与 AS 外部的路由器相连并互相交换路由信息；

18.1.2 LSA 类型

一台路由器中所有有效的 LSA 通告都被存放在它的链路状态数据库中，正确的 LSA 通告可以描述一个 OSPF 区域的网络拓扑结构。常见的 LSA 有 6 类，相应的描述如表 18-1 所示。

表 18-1 LSA 类型及相应的描述

类型代码	名称及路由代码	描述
1	路由器 LSA（O）	所有的 OSPF 路由器都会产生这种数据包，用于描述路由器上连接到某一个区域的链路或是某一接口的状态信息。该 LSA 只会在某一个特定的区域内扩散，而不会扩散至其它的区域。
2	网络 LSA（O）	由 DR 产生，只会在包含 DR 所处的广播网络的区域中扩散，不会扩散至其它的 OSPF 区域。
3	网络汇总 LSA（O IA）	由 ABR 产生，描述 ABR 和某个本地区域的内部路由器之间的链路信息。这些条目通过主干区域被扩散到其它的 ABR。
4	ASBR 汇总 LSA（O IA）	由 ABR 产生，描述到 ASBR 的可达性，由主干区域发送到其它 ABR。
5	外部 LSA（O E1 或 E2）	由 ASBR 产生，含有关于自治系统外的链路信息。
7	NSSA 外部 LSA（O N1 或 N2）	由 ASBR 产生的关于 NSSA 的信息，可以在 NSSA 区域内扩散，ABR 可以将类型 7 的 LSA 转换为类型 5 的 LSA。

18.1.3 区域类型

一个区域所设置的特性控制着它所能接收到的链路状态信息的类型。区分不同 OSPF 区域类型的关键在于它们对外部路由的处理方式。OSPF 区域类型如下：

1. 标准区域：可以接收链路更新信息和路由汇总；

2. 主干区域：连接各个区域的中心实体，所有其它的区域都要连接到这个区域上交换路由信息；

3. 末节区域（Stub Area）：不接受外部自治系统的路由信息；

4. 完全末节区域（Totally Stubby Area）：它不接受外部自治系统的路由以及自治系统内其它区域的路由汇总，完全末节区域是 Cisco 专有的特性；

5. 次末节区域(Not-So-Stubby Area,NSSA):允许接收以 7 类 LSA 发送的外部路由信息，并且 ABR 要负责把类型 7 的 LSA 转换成类型 5 的 LSA。

18.2 实验 1：多区域 OSPF 基本配置

1.实验目的

通过本实验可以掌握：

（1）在路由器上启动 OSPF 路由进程

（2）启用参与路由协议的接口，并且通告网络及所在的区域

（3）LSA 的类型和特征

（4）不同路由器类型的功能

（5）OSPF 拓扑结构数据库的特征和含义

（6）E1 路由和 E2 路由的区别

（7）查看和调试 OSPF 路由协议相关信息

2.实验拓扑

本实验的拓扑结构如图 18-2 所示。

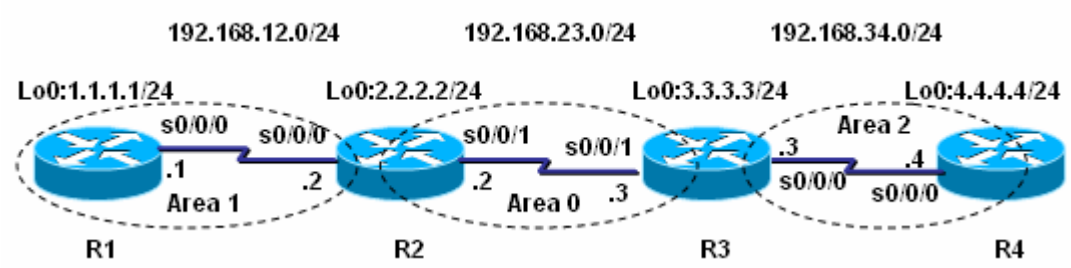

图 18-2 多区域 OSPF 基本配置

配置时采用环回接口尽量靠近区域 0 的原则。路由器 R4 的环回接口不在 OSPF 进程中通告，通过重分布的方法进入 OSPF 网络。

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 255.255.255.0 area 1
R1(config-router)#network 192.168.12.0 255.255.255.0 area 1
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 192.168.12.0 255.255.255.0 area 1
R2(config-router)#network 192.168.23.0 255.255.255.0 area 0
R2(config-router)#network 2.2.2.0 255.255.255.0 area 0
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 192.168.23.0 255.255.255.0 area 0
R3(config-router)#network 192.168.34.0 255.255.255.0 area 2
R3(config-router)#network 3.3.3.0 255.255.255.0 area 0
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 192.168.34.0 0.0.0.255 area 2
R4(config-router)#redistribute connected subnets
```

//将直连路由重分布到 OSPF 网络，重分布的内容在后面的章节详细介绍

4. 实验调试

（1）show ip route

```
R2#show ip route ospf

     1.0.0.0/24 is subnetted, 1 subnets
O       1.1.1.0 [110/65] via 192.168.12.1, 00:04:36, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
O       3.3.3.0 [110/65] via 192.168.23.3, 00:02:46, Serial0/0/1
     4.0.0.0/24 is subnetted, 1 subnets
O E2    4.4.4.0 [110/20] via 192.168.23.3, 00:02:22, Serial0/0/1
```

O IA 192.168.34.0/24 [110/128] via 192.168.23.3, 00:02:46, Serial0/0/1

以上输出表明路由器 R2 的路由表中既有区域内的路由“1.1.1.0”和“3.3.3.0”，又有区域间的路由“192.168.34.0”，还有外部区域的路由“4.4.4.0”。这就是为什么在 R4 上要用重分布，就是为了构造自治系统外的路由。

【技术要点】

OSPF 的外部路由分为：类型 1（在路由表中用代码“E1”表示）和类型 2（在路由表中用代码“E2”表示）。它们计算外部路由度量值的方式不同：

① 类型 1（E1）:外部路径成本＋数据包在 OSPF 网络所经过各链路成本；

② 类型 2（E2）:外部路径成本，即 ASBR 上的缺省设置。

在重分布的时候可以通过“**metric-type**”参数设置是类型 1 或 2，也可以通过“metric”参数设置外部路径成本，默认为 20。下面的是一个具体的实例：

R4(config-router)**#redistribute connected subnets metric 50 metric-type 1**

则在 R2 上关于“4.4.4.0”路由条目的信息如下：

O E1 4.4.4.0 [110/**178**] via 192.168.23.3, 00:01:27, Serial0/0/1

（2）**show ip ospf database**

R1**#show ip ospf database**

OSPF Router with ID (1.1.1.1) (Process ID 1)

Router Link States (Area 1) //区域 1 类型 1 的 LSA

Link ID	ADV Router	Age	Seq#	Checksum	Link count
1.1.1.1	1.1.1.1	595	0x80000007	0x00A0ED	3
2.2.2.2	2.2.2.2	459	0x80000004	0x002E71	2

Summary Net Link States (Area 1) //区域 1 类型 3 的 LSA

Link ID	ADV Router	Age	Seq#	Checksum
2.2.2.0	2.2.2.2	459	0x80000002	0x000D20
3.3.3.0	2.2.2.2	459	0x80000002	0x006B7E
192.168.23.0	2.2.2.2	459	0x80000002	0x001E55
192.168.34.0	2.2.2.2	459	0x80000002	0x002701

Summary ASB Link States (Area 1) //区域 1 类型 4 的 LSA

Link ID	ADV Router	Age	Seq#	Checksum
4.4.4.4	2.2.2.2	459	0x80000002	0x008919

Type-5 AS External Link States//类型 5 的 LSA

Link ID	ADV Router	Age	Seq#	Checksum	Tag
4.4.4.0	4.4.4.4	349	0x80000003	0x008460	0

R2#**show ip ospf database**

OSPF Router with ID (2.2.2.2) (Process ID 1)

Router Link States (Area 0)//区域0类型1的LSA

Link ID	ADV Router	Age	Seq#	Checksum	Link count
2.2.2.2	2.2.2.2	1712	0x80000004	0x006208	3
3.3.3.3	3.3.3.3	1677	0x80000004	0x00F56C	3

Summary Net Link States (Area 0) //区域0类型3的LSA

Link ID	ADV Router	Age	Seq#	Checksum
1.1.1.0	2.2.2.2	1785	0x80000001	0x00B53B
192.168.12.0	2.2.2.2	1785	0x80000001	0x0099E5
192.168.34.0	3.3.3.3	1673	0x80000001	0x0088DC

Summary ASB Link States (Area 0) //区域0类型4的LSA

Link ID	ADV Router	Age	Seq#	Checksum
4.4.4.4	3.3.3.3	1652	0x80000001	0x00EAF4

Router Link States (Area 1) //区域1类型1的LSA

Link ID	ADV Router	Age	Seq#	Checksum	Link count
1.1.1.1	1.1.1.1	1794	0x80000006	0x00A2EC	3
2.2.2.2	2.2.2.2	1786	0x80000003	0x003070	2

Summary Net Link States (Area 1) //区域1类型3的LSA

Link ID	ADV Router	Age	Seq#	Checksum
2.2.2.0	2.2.2.2	1782	0x80000001	0x000F1F
3.3.3.0	2.2.2.2	1698	0x80000001	0x006D7D
192.168.23.0	2.2.2.2	1738	0x80000001	0x002054
192.168.34.0	2.2.2.2	1672	0x80000001	0x0029FF

Summary ASB Link States (Area 1) //区域1类型4的LSA

Link ID	ADV Router	Age	Seq#	Checksum
4.4.4.4	2.2.2.2	1653	0x80000001	0x008B18

Type-5 AS External Link States// 类型5的LSA

Link ID	ADV Router	Age	Seq#	Checksum	Tag

```
4.4.4.0         4.4.4.4         203         0x80000002 0x00865F 0
```

以上输出结果包含了区域 1 的 LSA 类型 1、LSA 类型 3、LSA 类型 4、LSA 类型 5 的链路状态信息,以及区域 0 的 LSA 类型 1，LSA 类型 3，LSA 类型 4 的链路状态信息。同时看到路由器 R1 和 R2 的区域 1 的链路状态数据库完全相同。

【技术要点】

① 相同区域内的路由器具有相同的链路状态数据库，只是在虚链路的时候略有不同；

② 命令“**show ip ospf database**”所显示的内容并不是数据库中存储的关于每条 LSA 的全部信息，而仅仅是 LSA 的头部信息。要看 LSA 的全部信息，该命令后面还有跟详细的参数，如“**show ip ospf database router**”，结果显示如下：

```
R1#show ip ospf database router

            OSPF Router with ID (1.1.1.1) (Process ID 1)

                Router Link States (Area 1)

  LS age: 1355
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 1.1.1.1
  Advertising Router: 1.1.1.1
  LS Seq Number: 80000008
  Checksum: 0x9EEE
  Length: 60
  Number of Links: 3

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 1.1.1.0
     (Link Data) Network Mask: 255.255.255.0
      Number of TOS metrics: 0
       TOS 0 Metrics: 1

    Link connected to: another Router (point-to-point)
     (Link ID) Neighboring Router ID: 2.2.2.2
     (Link Data) Router Interface address: 192.168.12.1
      Number of TOS metrics: 0
       TOS 0 Metrics: 64

    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 192.168.12.0
     (Link Data) Network Mask: 255.255.255.0
      Number of TOS metrics: 0
       TOS 0 Metrics: 64
```

```
Routing Bit Set on this LSA
LS age: 1267
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 2.2.2.2
Advertising Router: 2.2.2.2
LS Seq Number: 80000005
Checksum: 0x2C72
Length: 48
Area Border Router
Number of Links: 2

  Link connected to: another Router (point-to-point)
   (Link ID) Neighboring Router ID: 1.1.1.1
   (Link Data) Router Interface address: 192.168.12.2
    Number of TOS metrics: 0
     TOS 0 Metrics: 64

  Link connected to: a Stub Network
   (Link ID) Network/subnet number: 192.168.12.0
   (Link Data) Network Mask: 255.255.255.0
    Number of TOS metrics: 0
     TOS 0 Metrics: 64
```

以上输出是路由器 R1 在区域 1 的 LSA 类型 1 的全部信息。

（3）show ip ospf

```
R4#show ip ospf 1
 Routing Process "ospf 1" with ID 4.4.4.4
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 It is an autonomous system boundary router
 Redistributing External Routes from,
 ......
```

以上信息表明路由器 R4 是一台 ASBR。

18.3 多区域 OSPF 高级配置

18.3.1 实验 2：OSPF 手工汇总

1.实验目的

通过本实验可以掌握：

（1）路由汇总的目的

（2）区域间路由汇总

（3）外部自治系统路由汇总

2.实验拓扑

本实验的拓扑结构如图 18-3 所示。

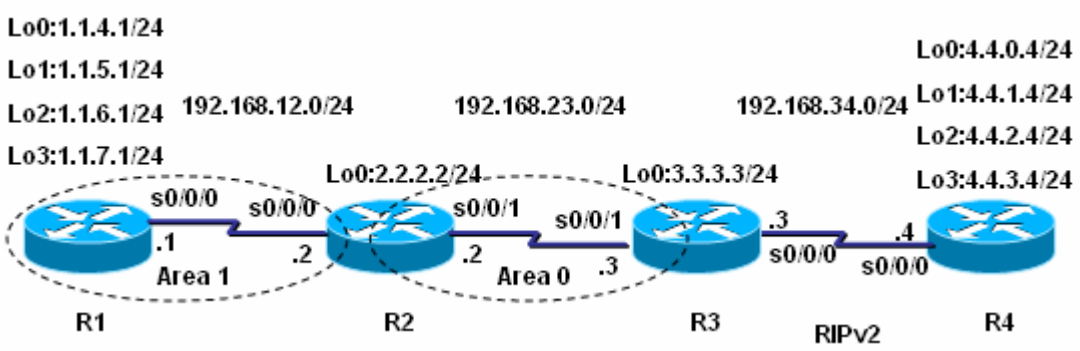

图 18-3 OSPF 手工汇总

路由器 R1、R2 和 R3 之间运行 OSPF,路由器 R3 和 R4 之间运行 RIPv2，路由器 R1 上的四个环回接口是为在路由器 R2 上做区域间路由汇总准备的，路由器 R4 上的四个环回接口是为在路由器 R3 上做外部路由汇总准备的。由于路由器 R3 是边界路由器，所以要完成双向重分布。

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.4.0 255.255.252.0 area 1
R1(config-router)#network 192.168.12.0 255.255.255.0 area 1
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 192.168.12.0 255.255.255.0 area 1
R2(config-router)#network 192.168.23.0 255.255.255.0 area 0
R2(config-router)#network 2.2.2.0 255.255.255.0 area 0
R2(config-router)#area 1 range 1.1.4.0 255.255.252.0    //配置区域间路由汇总
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 192.168.23.0 0.0.0.255 area 0
R3(config-router)#summary-address 4.4.0.0 255.255.252.0
//配置外部自治系统路由汇总
R3(config-router)#redistribute rip subnets //将 RIP 路由重分布到 OSPF 中
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#no auto-summary
R3(config-router)#network 192.168.34.0
R3(config-router)#redistribute ospf 1 metric 2  //将 OSPF 路由重分布到 RIP 中
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router rip
```

```
R4(config-router)#version 2
R4(config-router)#no auto-summary
R4(config-router)#network 4.0.0.0
R4(config-router)#network 192.168.34.0
```

【技术要点】

（1）区域间路由汇总必须在 ABR 上完成；

（2）外部路由汇总必须在 ASBR 上完成。

4.实验调试

（1）在 R2 上查看路由表，显示如下：

```
R2#show ip route ospf

     1.0.0.0/8 is variably subnetted, 5 subnets, 3 masks
O       1.1.5.1/32 [110/65] via 192.168.12.1, 00:17:16, Serial0/0/0
O       1.1.4.0/24 [110/65] via 192.168.12.1, 00:17:16, Serial0/0/0
O       1.1.4.0/22 is a summary, 00:17:16, Null0
O       1.1.7.1/32 [110/65] via 192.168.12.1, 00:17:16, Serial0/0/0
O       1.1.6.1/32 [110/65] via 192.168.12.1, 00:17:16, Serial0/0/0

     3.0.0.0/24 is subnetted, 1 subnets
O       3.3.3.0 [110/65] via 192.168.23.3, 00:12:14, Serial0/0/1
     4.0.0.0/22 is subnetted, 1 subnets
O E2    4.4.0.0 [110/20] via 192.168.23.3, 00:11:09, Serial0/0/1
O E2 192.168.34.0/24 [110/20] via 192.168.23.3, 00:12:15, Serial0/0/1
```

以上输出表明 R2 对 R1 的四条环回接口的路由汇总后，会产生一条指向 Null0 的路由；同时收到经路由器 R3 汇总的路由，因为是重分布进来的外部路由，所以路由代码为“O E2”。

（2）在 R3 上查看路由表，显示如下：

```
R3#show ip route ospf

O IA 192.168.12.0/24 [110/128] via 192.168.23.2, 00:23:20, Serial0/0/1
     1.0.0.0/22 is subnetted, 1 subnets
O IA    1.1.4.0 [110/129] via 192.168.23.2, 00:23:20, Serial0/0/1
     2.0.0.0/24 is subnetted, 1 subnets
O       2.2.2.0 [110/65] via 192.168.23.2, 00:23:20, Serial0/0/1
     4.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
O       4.4.0.0/22 is a summary, 00:20:29, Null0
```

以上输出表明 R3 对四条环回接口的 RIP 路由汇总后，会产生一条指向 Null0 的路由；同时收到经路由器 R2 汇总的路由，由于是区域间路由汇总，所以路由代码为“O IA”。

18.3.2 实验 3：OSPF 末节区域和完全末节区域

1.实验目的

通过本实验可以掌握：

（1）末节区域的条件

（2）末节区域的特征
（3）完全末节区域的特征
（4）末节区域的配置
（5）完全末节区域的配置

2. 实验拓扑

本实验的拓扑结构如图 18-4 所示。

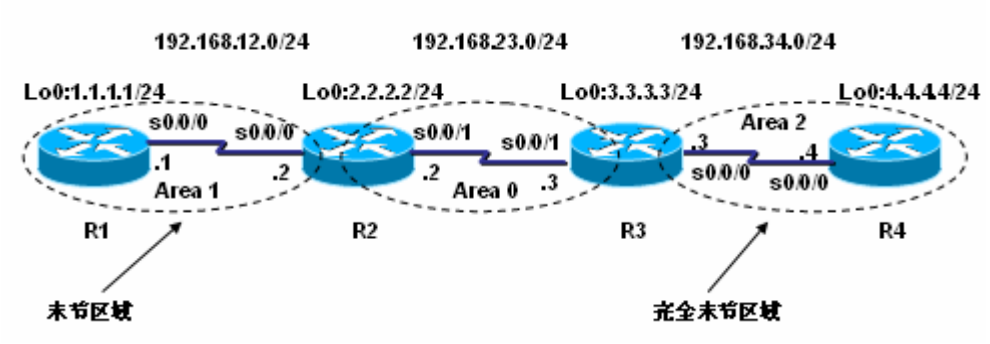

图 18-4 OSPF 末节区域配置

本实验在路由器 R2 上将环回接口 0 以重分布的方式注入 OSPF 区域，用来构造 5 类的 LSA。把区域 1 配置成末节区域，将区域 2 配置成完全末节区域。

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 255.255.255.0 area 1
R1(config-router)#network 192.168.12.0 255.255.255.0 area 1
R1(config-router)#area 1 stub // 把区域 1 配置成末节区域
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 192.168.12.0 255.255.255.0 area 1
R2(config-router)#network 192.168.23.0 255.255.255.0 area 0
R2(config-router)#redistribute connected subnets //将直连重分布进 OSPF 区域
R2(config-router)#area 1 stub
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 192.168.23.0 0.0.0.255 area 0
R3(config-router)#network 192.168.34.0 0.0.0.255 area 2
R3(config-router)#area 2 stub no-summary// 把区域 2 配置成完全末节区域
```

【技术要点】

“no-summary”阻止区域间的路由进入末节区域，所以叫完全末节区域。只需在 ABR 上启用本参数即可。

（4）步骤 4：配置路由器 R4

```
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 0.0.0.255 area 2
R4(config-router)#network 192.168.34.0 0.0.0.255 area 2
R4(config-router)#area 2 stub
```

【技术要点】

末节和完全末节区域需要满足如下的条件：

（1）区域只有一个出口；

（2）区域不需要作为虚链路的过渡区；

（3）区域内没有 ASBR；

（4）区域不是主干区域。

4.实验调试

（1）在 R1 上查看路由表，显示如下：

```
R1#show ip route ospf

     3.0.0.0/24 is subnetted, 1 subnets
O IA    3.3.3.0 [110/129] via 192.168.12.2, 00:12:29, Serial0/0/0
     4.0.0.0/32 is subnetted, 1 subnets
O IA    4.4.4.4 [110/193] via 192.168.12.2, 00:12:29, Serial0/0/0
O IA 192.168.23.0/24 [110/128] via 192.168.12.2, 00:12:29, Serial0/0/0
O IA 192.168.34.0/24 [110/192] via 192.168.12.2, 00:12:29, Serial0/0/0
O*IA 0.0.0.0/0 [110/65] via 192.168.12.2, 00:12:29, Serial0/0/0
```

以上的输出表明 R2 重分布进来的环回接口的路由并没有在 R1 的路由表中出现，说明末节区域不接收类型 5 的 LSA，也就是外部路由；同时末节区域 1 的 ABR R2 自动向该区域内传播 0.0.0.0/0 的默认路由；末节区域可以接收区域间路由。

（2）在 R4 上查看路由表，显示如下：

```
R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 192.168.34.3 to network 0.0.0.0

     4.0.0.0/24 is subnetted, 1 subnets
C       4.4.4.0 is directly connected, Loopback0
C    192.168.34.0/24 is directly connected, Serial0/0/0
```

O*IA 0.0.0.0/0 [110/65] via 192.168.34.3, 00:24:26, Serial0/0/0

以上输出表明在完全末节区域 2 中，R4 的路由表中除了直连和区域内路由，全部被默认路由代替，证明完全末节区域不接收外部路由和区域间路由,只有区域内的路由和一条由 ABR 向该区域注入的默认路由。

18.3.3 实验 4：OSPF NSSA 区域

1.实验目的

通过本实验可以掌握：

（1）NSSA 的特征

（2）NSSA 的配置

（3）NSSA 产生默认路由的方法

2.实验拓扑

本实验的拓扑结构如图 18-5 所示。

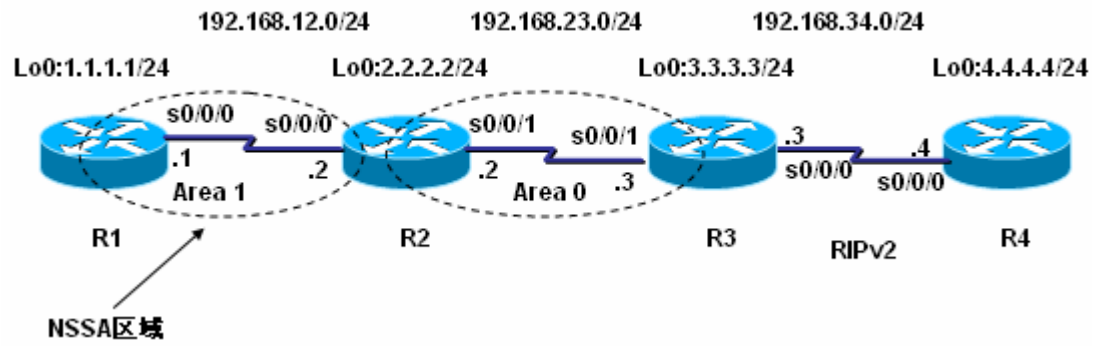

图 18-5 OSPF NSSA 区域配置

本实验在路由器 R1 上将环回接口 0 以重分布的方式注入 OSPF 区域，用来验证 5 类的 LSA 在 NSSA 区域的传递方式。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 192.168.12.0 255.255.255.0 area 1
R1(config-router)#redistribute connected subnets
R1(config-router)#area 1 nssa       //将区域 1 配置成 NSSA
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 192.168.12.0 255.255.255.0 area 1
R2(config-router)#network 192.168.23.0 255.255.255.0 area 0
R2(config-router)#network 2.2.2.0 255.255.255.0 area 0
R2(config-router)#area 1 nssa
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 192.168.23.0 0.0.0.255 area 0
```

R3(config-router)#redistribute rip subnets //将 RIP 路由重分布到 OSPF 区域
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#no auto-summary
R3(config-router)#network 192.168.34.0
R3(config-router)#redistribute ospf 1 metric 2
（4）步骤 4：配置路由器 R4
R4(config)#router rip
R4(config-router)#version 2
R4(config-router)#no auto-summary
R4(config-router)#network 4.0.0.0
R4(config-router)#network 192.168.34.0

4.实验调试

（1）在 R1 上查看路由表，显示如下：

```
R1#show ip route ospf

     2.0.0.0/24 is subnetted, 1 subnets
O IA    2.2.2.0 [110/65] via 192.168.12.2, 00:06:11, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
O IA    3.3.3.0 [110/129] via 192.168.12.2, 00:06:11, Serial0/0/0
O IA 192.168.23.0/24 [110/128] via 192.168.12.2, 00:06:11, Serial0/0/0
```

以上的输出表明区域间的路由是可以进入到 NSSA 区域的;但是在 R1 的路由表中并没有出现在 R3 上把 RIP 重分布进来的路由,因此说明 LSA 类型为 5 的外部路由不能在 NSSA 区域中传播，ABR 也没有能力把类型 5 的 LSA 转成类型 7 的 LSA。

【技术要点】

如果不想在 NSSA 区域中出现区域间的路由,则在 ABR 的路由器上配置 NSSA 区域时加上“no-summary”参数即可。这时 ABR 也会自动向 NSSA 区域注入一条“O IA”的默认路由，配置如下：

R2(config-router)#area 1 nssa no-summary

R1 的路由表如下：

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 192.168.12.2 to network 0.0.0.0

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
```

```
C       1.1.1.0 is directly connected, Loopback0
O*IA 0.0.0.0/0 [110/65] via 192.168.12.2, 00:00:32, Serial0/0/0
```

本实验中，如果在路由器 R2 配置 NSSA 时没有加“no-summary”参数，那么对路由器 R1 来讲，RIP 部分的路由是不可达的，为了解决此问题，我们在路由器 R2 上配置 NSSA 区域时加上“default-information-originate”参数即可，此时 ABR 路由器 R2 会向 NSSA 区域注入一条“O N2”的默认路由，配置如下：

```
R2(config-router)#area 1 nssa default-information-originate
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 192.168.12.2 to network 0.0.0.0

     2.0.0.0/24 is subnetted, 1 subnets
O IA    2.2.2.0 [110/65] via 192.168.12.2, 00:01:57, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
O IA    3.3.3.0 [110/129] via 192.168.12.2, 00:01:57, Serial0/0/0
O IA 192.168.23.0/24 [110/128] via 192.168.12.2, 00:01:57, Serial0/0/0
O*N2 0.0.0.0/0 [110/1] via 192.168.12.2, 00:01:49, Serial0/0/0
```

如果在 R2 配置 NSSA 时 “no-summary”参数和“default-information-originate”参数都加，如下所示：

```
R2(config-router)#area 1 nssa default-information-originate  no-summary
```

则 R1 的路由表如下：

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 192.168.12.2 to network 0.0.0.0

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
O*IA 0.0.0.0/0 [110/65] via 192.168.12.2, 00:00:20, Serial0/0/0
```

以上输出表明“O IA”的路由优于“O N2”的路由。

（2）在 R2 上查看路由表，显示如下：

```
R2#show ip route ospf

     1.0.0.0/24 is subnetted, 1 subnets
O N2    1.1.1.0 [110/20] via 192.168.12.1, 00:04:11, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
O       3.3.3.0 [110/65] via 192.168.23.3, 00:04:11, Serial0/0/1
     4.0.0.0/24 is subnetted, 1 subnets
O E2    4.4.4.0 [110/20] via 192.168.23.3, 00:04:11, Serial0/0/1
O E2 192.168.34.0/24 [110/20] via 192.168.23.3, 00:04:11, Serial0/0/1
```

以上输出表明 NSSA 区域的路由代码为“O N2”或”O N1”。

（3）在 R2 上查看拓扑表，显示如下：

```
R2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
2.2.2.2         2.2.2.2         89          0x80000014 0x004810 3
3.3.3.3         3.3.3.3         85          0x8000000C 0x005BFD 3

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
192.168.12.0    2.2.2.2         89          0x8000000A 0x0087EE

                Router Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         16          0x80000009 0x002D6B 2
2.2.2.2         2.2.2.2         89          0x80000010 0x00C1C9 2

                Summary Net Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum
0.0.0.0         2.2.2.2         419         0x80000001 0x00FC31

                Type-7 AS External Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum Tag
0.0.0.0         2.2.2.2         657         0x80000001 0x00B978 0
1.1.1.0         1.1.1.1         275         0x80000002 0x00E92E 0
```

Type-5 AS External Link States

Link ID	ADV Router	Age	Seq#	Checksum	Tag
1.1.1.0	2.2.2.2	90	0x80000002	0x0060BD	0
4.4.4.0	3.3.3.3	1863	0x80000001	0x00FC8B	0
192.168.34.0	3.3.3.3	87	0x80000002	0x0062A5	0

从输出结果中表明，路由器 R2 将类型 7 的 LSA 转换成类型 5 的 LSA，并且继续在网络上扩散到路由器 R3。

18.4 OSPF 虚链路

在实际网络中，可能会存在主干区域不连续或者某一个区域与主干区域物理不相连的情况，在这两种情况下，可以通过虚链路来解决。

18.4.1 实验 5：不连续区域 0 的虚链路

1. 实验目的

通过本实验可以掌握：

（1）不连续区域 0 虚链路的特征

（2）虚链路的配置

2. 实验拓扑

本实验的拓扑结构如图 18-6 所示。

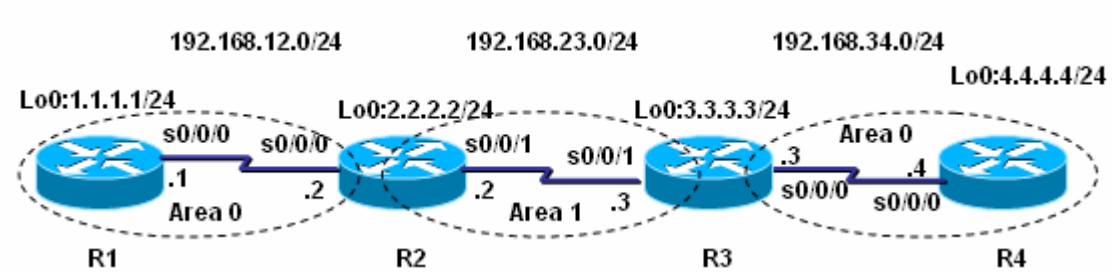

图 18-6 不连续区域 0 虚链路

本实验中区域 1 为转接区域。

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 0
R1(config-router)#network 192.168.12.0 0.0.0.255 area 0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 2.2.2.0 0.0.0.255 area 0
```

```
R2(config-router)#network 192.168.12.0 0.0.0.255 area 0
R2(config-router)#network 192.168.23.0 0.0.0.255 area 1
R2(config-router)#area 1 virtual-link 3.3.3.3  //配置虚链路
```

【技术要点】

配置虚链路的时候，“virtual-link”后一定要互指对方的路由器 ID。

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 192.168.23.0 0.0.0.255 area 1
R3(config-router)#network 192.168.34.0 0.0.0.255 area 0
R3(config-router)#area 1 virtual-link 2.2.2.2
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 0.0.0.255 area 0
R4(config-router)#network 192.168.34.0 0.0.0.255 area 0
```

4.实验调试

（1）show ip route

```
R1#show ip route ospf

     2.0.0.0/24 is subnetted, 1 subnets
O       2.2.2.0 [110/65] via 192.168.12.2, 00:04:42, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
O       3.3.3.0 [110/129] via 192.168.12.2, 00:04:42, Serial0/0/0
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/193] via 192.168.12.2, 00:04:42, Serial0/0/0
O IA 192.168.23.0/24 [110/128] via 192.168.12.2, 00:04:42, Serial0/0/0
O    192.168.34.0/24 [110/192] via 192.168.12.2, 00:04:42, Serial0/0/0
```

从以上输出可以看出通过虚拟链路将两个不连续的区域 0 连接起来。

（2）show ip ospf virtual-links

```
R2#show ip ospf virtual-links
Virtual Link OSPF_VL0 to router 3.3.3.3 is up
  Run as demand circuit
  DoNotAge LSA allowed.
  Transit area 1, via interface Serial0/0/1, Cost of using 64
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:03
    Adjacency State FULL (Hello suppressed)
    Index 2/3, retransmission queue length 0, number of retransmission 1
    First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
```

```
    Last retransmission scan length is 1, maximum is 1
    Last retransmission scan time is 0 msec, maximum is 0 msec
```

以上输出表明了虚链路的基本信息。

（3）**show ip ospf database**

```
R2#show ip ospf database

            OSPF Router with ID (2.2.2.2) (Process ID 1)

                Router Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum Link count
1.1.1.1         1.1.1.1         668         0x80000003 0x00ABE6 3
2.2.2.2         2.2.2.2         537         0x80000007 0x00EEB6 4
3.3.3.3         3.3.3.3         1     (DNA) 0x80000014 0x00C591 4
4.4.4.4         4.4.4.4         6     (DNA) 0x80000003 0x00AB8E 3

                Summary Net Link States (Area 0)

Link ID         ADV Router      Age         Seq#       Checksum
192.168.23.0    2.2.2.2         608         0x80000001 0x002054
192.168.34.0    3.3.3.3         16    (DNA) 0x80000001 0x00026E

                Router Link States (Area 1)

Link ID         ADV Router      Age         Seq#       Checksum Link count
2.2.2.2         2.2.2.2         562         0x80000002 0x00ED95 2
3.3.3.3         3.3.3.3         553         0x80000003 0x008BF1 2
```

以上输出表明虚链路的路由被拉进区域 0，并带有“(DNA)”标记，表示不老化。

18.4.1 实验 6：远离区域 0 的虚链路

1. 实验目的

通过本实验可以掌握：

（1）远离区域 0 虚链路的特征

（2）虚链路的配置

2. 实验拓扑

本实验的拓扑结构如图 18-7 所示。

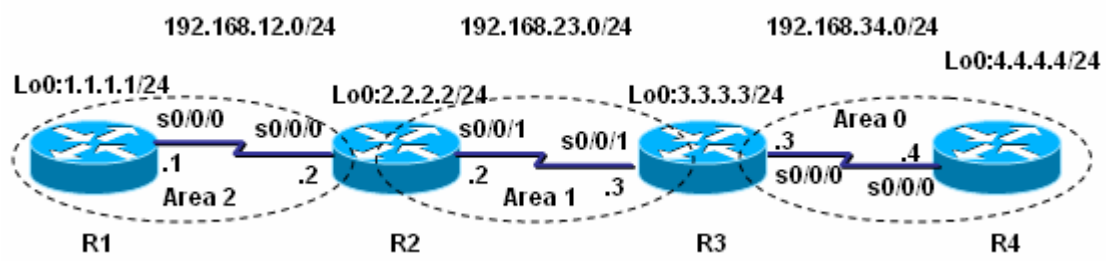

图 18-7 远离区域 0 虚链路

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 1.1.1.0 0.0.0.255 area 2
R1(config-router)#network 192.168.12.0 0.0.0.255 area 2
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 2.2.2.0 0.0.0.255 area 1
R2(config-router)#network 192.168.12.0 0.0.0.255 area 2
R2(config-router)#network 192.168.23.0 0.0.0.255 area 1
R2(config-router)#area 1 virtual-link 3.3.3.3
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 192.168.23.0 0.0.0.255 area 1
R3(config-router)#network 192.168.34.0 0.0.0.255 area 0
R3(config-router)#area 1 virtual-link 2.2.2.2
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 0.0.0.255 area 0
R4(config-router)#network 192.168.34.0 0.0.0.255 area 0
```

4.实验调试

在路由器 R4 上查看路由表：

```
R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

O IA 192.168.12.0/24 [110/192] via 192.168.34.3, 00:02:19, Serial0/0/0
     1.0.0.0/32 is subnetted, 1 subnets
O IA    1.1.1.1 [110/193] via 192.168.34.3, 00:02:19, Serial0/0/0
     2.0.0.0/32 is subnetted, 1 subnets
O IA    2.2.2.2 [110/129] via 192.168.34.3, 00:02:19, Serial0/0/0
```

```
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/65] via 192.168.34.3, 00:02:19, Serial0/0/0
     4.0.0.0/24 is subnetted, 1 subnets
C       4.4.4.0 is directly connected, Loopback0
O IA 192.168.23.0/24 [110/128] via 192.168.34.3, 00:02:19, Serial0/0/0
C    192.168.34.0/24 is directly connected, Serial0/0/0
```

从 R4 的路由表的输出，可以看出路由器 R1 能够通过使用转接区域 1 的虚拟链路到达区域 0。

【技术要点】

虚链路属于区域 0，所以在进行区域 0 认证的时候，不要忘记虚链路的认证，例如如果区域 0 采用 MD5 认证，则在虚链路上配置如下：

R3(config-router)**#area 1 virtual-link 2.2.2.2 message-digest-key 1 md5 cisco**

18.5 OSPF 命令汇总

表 18-2 列出了本章涉及到的主要的命令。

表 18-2 本章命令汇总

命令	**作用**
show ip route	查看路由表
show ip ospf neighbor	查看 OSPF 邻居的基本信息
show ip ospf database	查看 OSPF 拓扑结构数据库
show ip ospf interface	查看 OSPF 路由器接口的信息
show ip ospf	查看 OSPF 进程及其细节
show ip ospf database router	查看类型 1 的 LSA 的全部信息
redistribute	路由协议重分布
area *area-id* range	区域间路由汇总
summary-address	外部路由汇总
area *area-id* stub	把某区域配置成末节区域
area *area-id* stub no-summary	把某区域配置成完全末节区域
area *area-id* nssa	把某区域配置成 NSSA 区域
area *area-id* virtual-link	配置虚链路

第 19 章 IS-IS

近几年来，随着在 ISP 中的广泛应用，IS（Intermediate System，中间系统）-IS 路由协议已经变得很普及。IS-IS 最初是由国际标准化组织制定的一个 OSI（开放系统互联）路由协议，被设计成工作在 OSI 无连接网络服务（CLNS）的环境中。

19.1 IS-IS 概述

19.1.1 IS-IS 特点

IS-IS 是一个非常灵活的路由协议，具有很好地可扩展性，而且已经整合了诸如 MPLS（多协议标记交换）之类的特性，其主要特点如下：

1. 维护一个链路状态数据库，并使用 SPF 算法来计算最佳路径；
2. 用 Hello 包建立和维护邻居关系；
3. 使用区域来构造两级层次化的拓扑结构；
4. 在区域之间可以使用路由汇总来减少路由器的负担；
5. 支持 VLSM 和 CIDR；
6. 在广播多路访问网络，通过选举指定 IS（DIS）来管理和控制网络上的泛洪扩散；
7. 具有认证功能；
8. IS-IS 采用 cost 作为度量值；
9. IS-IS 管理距离为 115；
10. 快速收敛；
11. 适合大型网络。

19.1.2 术语

1. CLNS（ Connectionless Network Service，无连接网络服务）：使用数据报传输服务，在数据传输之前不需要建立连接，它描述提供给传输层的服务；

2. CLNP（ Connectionless Network Protocol, 无连接网络协议）：是 OSI 模型中网络层中的一种无连接的网络协议，和 IP 有相同的特质；

3. ES（End system,端系统）：没有路由能力的网络节点；

4. IS（Intermediate System，中间系统）：有数据包转发能力的网络节点，即路由器；

5. NSAP（Network Service Access Point ，网络服务访问点），是网络层和传输层边界上概念性的点。每一个传输层实体都会分配得到唯一的 NSAP 地址；

6. Level 1 路由器：类似 OSPF 的内部路由器；

7. Level 1/2 路由器：类似 OSPF 的 ABR；

8. Level 2 路由器：类似 OSPF 的主干路由器；

9. SNPA（Subnetwork Point of Attachment，子网连接点）：是和三层地址对应的二层地址，如 MAC 地址、DLCI 等；

10. ISO 地址：ISO 地址有两种形式，NET（网络实体标题）和 NSAP 地址，其中 NET 是 NSEL 的值为 0x00 时的 NSAP 地址，NSAP 地址长度为 8-20 个字节，包括区域、系统 ID 和 NSEL 三个部分，其中前两部分可以分得更细。

19.2 实验 1：集成 IS-IS 的基本配置

1.实验目的

通过本实验可以掌握

（1）在路由器上启动 IS-IS 路由进程

（2）启用参与路由协议的接口

（3）度量值 cost 的计算

（4）NET 地址配置

（5）DIS 选举的控制

（6）查看和调试 IS-IS 路由协议相关信息

2. 拓扑结构

实验拓扑如图 19-1 所示。

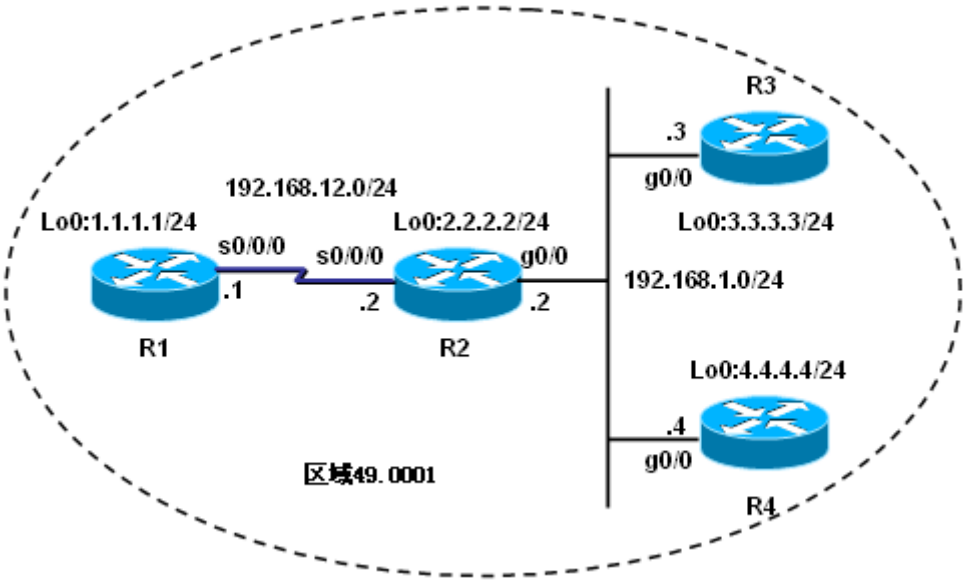

图 19-1 集成 IS-IS 的基本配置

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router isis                              //启动 IS-IS 路由进程
R1(config-router)#net 49.0001.1111.1111.1111.00     //配置 NET 地址
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-if)#ip router isis                        //接口下启用 IS-IS
R1(config-if)#interface Serial0/0/0
R1(config-if)#ip address 192.168.12.1 255.255.255.0
R1(config-if)#ip router isis
R1(config-if)#no shutdown
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router isis
R2(config-router)#net 49.0001.2222.2222.2222.00
R2(config)#interface Loopback0
R2(config-if)#ip address 2.2.2.2 255.255.255.0
R2(config-if)#ip router isis
R2(config)#interface gigabitethernet0/0
```

```
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#ip router isis
R2(config-if)#no shutdown
R2(config)#interface Serial0/0/0
R2(config-if)#ip address 192.168.12.2 255.255.255.0
R2(config-if)#clockrate 128000
R2(config-if)#ip router isis
R2(config-if)#no shutdown
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router isis
R3(config-router)#net 49.0001.3333.3333.3333.00
R3(config)#interface Loopback0
R3(config-if)#ip address 3.3.3.3 255.255.255.0
R3(config-if)#ip router isis
R3(config)#interface gigabitethernet0/0
R3(config-if)#ip address 192.168.1.3 255.255.255.0
R3(config-if)#ip router isis
R3(config-if)#no shutdown
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router isis
R4(config-router)#net 49.0001.4444.4444.4444.00
R4(config)#interface Loopback0
R4(config-if)#ip address 4.4.4.4 255.255.255.0
R4(config-if)#ip router isis
R4(config)#interface gigabitethernet0/0
R4(config-if)#ip address 192.168.1.4 255.255.255.0
R4(config-if)#ip router isis
R4(config-if)#no shutdown
```

4.实验调试

（1）show clns neighbors

该命令用来显示 IS-IS 的邻居。

```
R2#show clns neighbors

System Id      Interface   SNPA            State  Holdtime  Type Protocol
R1             Se0/0/0     *HDLC*          Up     24        L1L2 IS-IS
R3             Gi0/0       ca02.0f78.0000  Up     26        L1L2 IS-IS
R4             Gi0/0       ca03.0f78.0000  Up     8         L1L2 IS-IS
```

从以上输出可以看出，路由器 R2 有 3 个邻居,而且都是“L1L2”类型的，这也是启动 IS-IS 的路由器的默认类型。由于 R1 和 R2 是通过串行连接的，所以 SNPA 为“*HDLC*”，而 R2 与 R3 和 R4 是通过以太网连接的,所以 SNPA 分别是 R3 和 R4 以太口“gigabitethernet0/0”的 MAC 地址。

【提示】

从IOS12.0(5)版本开始,Cisco路由器支持动态主机名字映射,可以通过命令“**show isis hostname**”查看：

```
R2#show isis hostname
Level  System ID      Dynamic Hostname  (notag)
 1     4444.4444.4444 R4
 2     3333.3333.3333 R3
 1     1111.1111.1111 R1
    *  2222.2222.2222 R2
```

上面输出清楚的显示了系统 ID 和动态主机名的映射关系，其中“*” 表示本地路由器。

【注意】

缺省情况下，Hello 包每 10 秒中发送一次，holddown 时间为 30 秒，即 3 倍的关系。可以在接口下通过“**isis hello-interval**” 命令修改 Hello 包发送的周期，同时通过“**isis hello-multiplier**”命令定义 holddown 是 Hello 周期的倍数。

（2）**show clns protocol**

该命令显示和 CLNS 路由协议相关的信息。

```
R2#show clns protocol
IS-IS Router: <Null Tag>
  System Id: 2222.2222.2222.00  IS-Type: level-1-2
//系统 ID 以及 IS-IS 路由器类型
  Manual area address(es):
        49.0001
  Routing for area address(es):
        49.0001
  Interfaces supported by IS-IS:
        GigabitEthernet0/0 - IP
        Serial0/0/0 - IP
        Loopback0  -  IP
//以上四行表示运行 IS-IS 路由协议的接口
  Redistribute:
    static (on by default)
  Distance for L2 CLNS routes: 110
//管理距离
  RRR level: none
  Generate narrow metrics: level-1-2
  Accept narrow metrics:   level-1-2
//使用“窄”度量
  Generate wide metrics:   none
  Accept wide metrics:     none
```

（3）**show clns interface**

该命令显示 clns 接口状态的基本信息。

```
R2#show clns interface s0/0/0
Serial0/0/0 is up, line protocol is up
```

```
  Checksums enabled, MTU 1500, Encapsulation HDLC
  ERPDUs enabled, min. interval 10 msec.
  CLNS fast switching enabled
//CLNS 快速交换启动
  CLNS SSE switching disabled
//CLNS SSE 交换关闭
  DEC compatibility mode OFF for this interface
  Next ESH/ISH in 47 seconds
  Routing Protocol: IS-IS
    Circuit Type: level-1-2    //电路类型
    Interface number 0x1, local circuit ID 0x100
    Neighbor System-ID: R1
    Level-1 Metric: 10, Priority: 64, Circuit ID: R2.00
// 接口 Level-1 的度量值、接口优先级以及电路 ID
    Level-1 IPv6 Metric: 10
    Number of active level-1 adjacencies: 1
    Level-2 Metric: 10, Priority: 64, Circuit ID: R2.00
// 接口 Level-2 的度量值、接口优先级以及电路 ID
    Level-2 IPv6 Metric: 10
    Number of active level-2 adjacencies: 1
    Next IS-IS Hello in 7 seconds
    if state UP
```

（4）show clns route

该命令查看 CLNS 第二层路由信息。

```
R2#show clns route
Codes: C - connected, S - static, d - DecnetIV
       I - ISO-IGRP,  i - IS-IS,  e - ES-IS
       B - BGP,       b - eBGP-neighbor

C  49.0001.2222.2222.2222.00 [1/0], Local IS-IS NET
C  49.0001 [2/0], Local IS-IS Area
```

因为这条命令用于 OSI 路由选择，所以以上输出没有太多的信息。

（5）show isis topology

该命令显示 IS-IS 的拓扑结构信息，包含到其它中间系统的路径信息。

```
R2#show isis topology
IS-IS IP paths to level-1 routers
System Id        Metric     Next-Hop         Interface    SNPA
R1               10         R1               Se0/0/0       *HDLC*
R2               --
R3               10         R3               Gi0/0        ca02.0f78.0000
R4               10         R4               Gi0/0        ca03.0f78.0000

IS-IS IP paths to level-2 routers
System Id        Metric     Next-Hop         Interface     SNPA
```

R1	10	R1	Se0/0/0	*HDLC*
R2	--			
R3	10	R3	Gi0/0	ca02.0f78.0000
R4	10	R4	Gi0/0	ca03.0f78.0000

以上输出表明 IS-IS 为 L1 路由器和 L2 路由器分别存放拓扑结构数据库的，其中“metric”是到达目标的 cost 之和。

（6）**show isis database**

该命令显示 IS-IS 链路状态数据库。

R2#**show isis database**

IS-IS **Level-1** Link State Database:

LSPID		LSP Seq Num	LSP Checksum	LSP Holdtime	ATT/P/OL
R1.00-00		0x0000001C	0x5F4F	1077	0/0/0
R2.00-00	*	0x0000001E	0xF57A	974	0/0/0
R3.00-00		0x0000001C	0x0608	855	0/0/0
R4.00-00		0x0000001B	0xA5FA	701	0/0/0
R4.02-00		0x00000018	0x9BE2	592	0/0/0

IS-IS **Level-2** Link State Database:

LSPID		LSP Seq Num	LSP Checksum	LSP Holdtime	ATT/P/OL
R1.00-00		0x00000020	0xFD8C	882	0/0/0
R2.00-00	*	0x00000020	0x125F	493	0/0/0
R3.00-00		0x0000001E	0x658B	869	0/0/0
R4.00-00		0x0000001B	0x1870	542	0/0/0
R4.02-00		0x00000018	0x4DB9	916	0/0/0

以上输出表明：

① IS-IS 为第一层路由和第二层路由分别维护独立的链路状态数据库。由于 IS-IS 是链路状态路由协议，而且四台路由器具有相同区域，所以它们的链路状态数据库是相同的；

② 路由器 R4 是 DIS,LSPID（链路状态协议数据单元 ID）由三个部分构成：

第一部分是系统 ID,长度为 6 个字节；

第二部分是伪节点 ID,长度为一个字节，它代表了一个 LAN,当这个值非 0 时，表示该路由器为 DIS；

第三部分是 LSP 分段号，长度为一个字节，如果是 00 表示所有的数据都在单个的 LSP 中；

③ 系统 ID 和伪节点就构成了电路 ID（Circuit ID），如：“R4.02”。

【技术要点】

DIS 的选举原则如下：

① 只有形成邻接关系的路由器才有资格参与选举；

② 接口优先级最高成为 DIS；

③ 如果接口优先级相同，则最高的 SNPA 地址成为 DIS；

④ DIS 选举是抢占的。

修改接口优先级的命令是“**isis priority**”，默认是 64，取值范围为 0-127。在本例中

可以将 R2 的以太口的接口优先级改为 100，则 R2 马上被选为 DIS，显示如下：

```
R2#show isis database
IS-IS Level-1 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
R1.00-00              0x00000020   0x5753        1196              0/0/0
R2.00-00            * 0x00000023   0xD562        1197              0/0/0
R2.02-00            * 0x00000001   0x63FE        1189              0/0/0
R3.00-00              0x00000021   0xC80D        1191              0/0/0
R4.00-00              0x00000020   0x68FF        1191              0/0/0
IS-IS Level-2 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
R1.00-00              0x00000024   0xF590        1196              0/0/0
R2.00-00            * 0x00000025   0x58E0        1198              0/0/0
R2.02-00            * 0x00000001   0x15D5        1198              0/0/0
R3.00-00              0x00000024   0xB0B5        1196              0/0/0
R4.00-00              0x00000021   0x639A        1196              0/0/0
```

（7）show isis route

该命令查看 CLNS 第一层路由信息。

```
R2#show isis route
IS-IS not running in OSI mode (*) (only calculating IP routes)
(*) Use "show isis topology" command to display paths to all routers
```

由于该命令是针对 OSI 路由选择协议的，所以没有具体的输出。

（8）show ip protocols

该命令显示和 IP 路由协议相关的信息。

```
R2#show ip protocols
Routing Protocol is "isis"
  Invalid after 0 seconds, hold down 0, flushed after 0
//更新计时器全部为 0，表示 IS-IS 路由协议采用出发更新
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: isis
  Address Summarization:
    None
  Maximum path: 4
  Routing for Networks:
    Loopback0
    Serial0/0/0
    Gigabitethernet0/0
//以上四行表示运行 IS-IS 路由协议的接口
  Routing Information Sources:
    Gateway         Distance      Last Update
    4.4.4.4              115      00:06:51
    3.3.3.3              115      00:06:51
    1.1.1.1              115      00:06:51
```

```
//以上五行表示路由信息源
  Distance: (default is 115)
//默认管理距离
```

（9）show ip route

```
R2#show ip route isis

     1.0.0.0/24 is subnetted, 1 subnets
i L1    1.1.1.0 [115/20] via 192.168.12.1, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
i L1    3.3.3.0 [115/20] via 192.168.1.3, GigabitEthernet0/0
     4.0.0.0/24 is subnetted, 1 subnets
i L1    4.4.4.0 [115/20] via 192.168.1.4, GigabitEthernet0/0
```

以上输出表明，如果路由器类型为“L1/L2”,区域内的路由用“i L1”表示，即 level-1 路由。

【提示】

默认情况下,IS-IS 使用窄度量计算度量值,所有链路都使用 10 作为度量值,因为 IS-IS 不能象 OSPF 那样基于带宽自动的计算度量值。

19.3 实验 2：多区域集成的 IS-IS

1. 实验目的

通过本实验可以掌握

（1）在路由器上启动 IS-IS 路由进程

（2）启用参与路由协议的接口

（3）L1 和 L2 路由的区别

（4）配置 L1 或 L2 路由器

（5）配置电路类型

（6）配置区域间路由汇总

（7）通告默认路由

（8）配置 IS-IS 认证

（9）查看和调试多区域 IS-IS 路由协议相关信息

2. 拓扑结构

实验拓扑如图 19-2 所示。

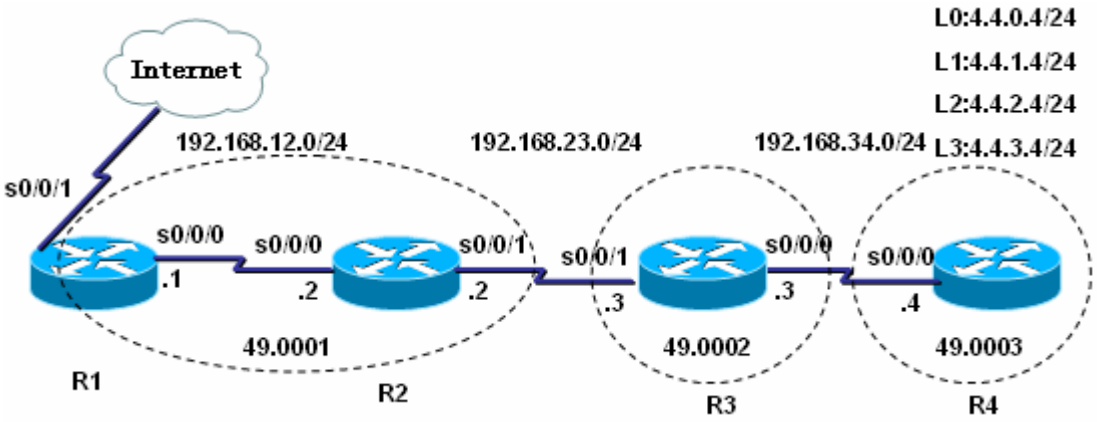

图 19-2 多区域集成的 IS-IS

【说明】

IS-IS 区域的划分是基于路由器的，也就是说一个路由器只能属于一个区域，而 OSPF 区域的划分是基于链路的。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router isis
R1(config-router)#net 49.0001.1111.1111.1111.00
R1(config-router)#is-type level-1  //将 R1 配置成 L1 路由器
R1(config-router)#area-password area //启用区域认证
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-if)#ip router isis
R1(config)#interface Serial0/0/0
R1(config-if)#ip address 192.168.12.1 255.255.255.0
R1(config-if)#ip router isis
R1(config-if)#isis password neighbor level-1 //启用 level 1 邻居认证
R1(config-if)#no shutdown
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router isis
R2(config-router)#net 49.0001.2222.2222.2222.00
R2(config-router)#default-information originate //向 IS-IS 区域注入默认路由
R2(config-router)#area-password area
R2(config-router)#domain-password domain //启用域认证
R2(config)#interface Serial0/0/0
R2(config-if)#ip address 192.168.12.2 255.255.255.0
R2(config-if)#clockrate 128000
R2(config-if)#ip router isis
R2(config-if)#isis password neighbor level-1
R2(config-if)#no shutdown
R2(config)#interface Serial0/0/1
R2(config-if)#ip address 192.168.23.2 255.255.255.0
R2(config-if)#ip router isis
R2(config-if)#clockrate 128000
R2(config-if)#no shutdown
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router isis
R3(config-router)#net 49.0002.3333.3333.3333.00
R3(config-router)#is-type level-2-only    //将 R3 配置成 L2 路由器
R3(config-router)#domain-password domain
R3(config)#interface Serial0/0/0
R3(config-if)#ip address 192.168.34.3 255.255.255.0
```

```
R3(config-if)#ip router isis
R3(config-if)#isis circuit-type level-2-only //配置接口电路类型
R3(config-if)#isis password neighborpassword level-2 //启用 level 2 邻居认证
R3(config-if)#clockrate 128000
R3(config-if)#no shutdown
R3(config)#interface Serial0/0/1
R3(config-if)#ip address 192.168.23.3 255.255.255.0
R3(config-if)#ip router isis
R3(config-if)#no shutdown
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router isis
R4(config-router)#net 49.0003.4444.4444.4444.00
R4(config-router)#summary-address 4.4.0.0 255.255.252.0//配置区域间路由汇总
R4(config-router)#is-type level-2-only
R4(config-router)#domain-password domain
R4(config)#interface Loopback0
R4(config-if)#ip address 4.4.0.4 255.255.255.0
R4(config-if)#ip router isis
R4(config)#interface Loopback1
R4(config-if)#ip address 4.4.1.4 255.255.255.0
R4(config-if)#ip router isis
R4(config)#interface Loopback2
R4(config-if)#ip address 4.4.2.4 255.255.255.0
R4(config-if)#ip router isis
R4(config)#interface Loopback3
R4(config-if)#ip address 4.4.3.4 255.255.255.0
R4(config-if)#ip router isis
R4(config-if)#interface Serial0/0/0
R4(config-if)#ip address 192.168.34.4 255.255.255.0
R4(config-if)#ip router isis
R4(config-if)#isis circuit-type level-2-only
R4(config-if)#isis password neighborpassword level-2
R4(config-if)#no shutdown
```

【技术要点】

IS-IS 的认证只限于明文口令，Cisco 的 IOS 支持 3 个级别的认证：

（1）邻居认证：相互连接的路由器接口必须配置相同的口令，同时必须为 L1 和 L2 类型的邻居关系配置各自的认证，L1 邻居认证的密码和 L2 邻居认证的密码可以不同。邻居认证通过命令“isis password”配置。本实验中 R1 和 R2 之间的串行链路启用 Level-1 的邻居认证，而 R3 和 R4 之间的串行链路启用 Level-2 的邻居认证；

（2）区域认证：区域内的每台路由器必须执行认证，并且必须使用相同的口令。区域认证通过命令“area-password”配置。本实验中区域“49.0001”启用区域认证；

（3）域认证：域内的每一个 L2 和 L1/L2 类型的路由器必须执行认证，并且必须使用相同的口令。域认证通过命令“domain-password”配置。本实验中 R2、R3 和 R4 都配置域认

证，因为路由器 R1 是 L1 路由器，所以不用配置域认证。

4.实验调试

（1）show isis database

```
R1#show isis database

IS-IS Level-1 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
R1.00-00            * 0x0000003B   0x1F72        659               0/0/0
R2.00-00              0x00000039   0x9DEE        658               1/0/0
R2#show isis database

IS-IS Level-1 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
R1.00-00              0x00000008   0x8161        993               1/0/0
R2.00-00            * 0x00000007   0x02BC        902               1/0/0
IS-IS Level-2 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
R1.00-00              0x00000009   0xE455        1120              0/0/0
R2.00-00            * 0x00000006   0xFFFD        903               0/0/0
R3.00-00              0x00000009   0xE9C4        1039              0/0/0
R4.00-00              0x0000000B   0xC040        1110              0/0/0
R3#show isis database

IS-IS Level-2 Link State Database:
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
R2.00-00              0x0000003B   0xC0D2        1036              0/0/0
R3.00-00            * 0x00000044   0x0FE3        1043              0/0/0
R4.00-00              0x00000046   0xEC58        1041              0/0/0
```

以上输出表明：

① R1 路由器为 L1 路由器，只维护 L1 的链路状态数据库；

② R2 路由器为 L1/L2 路由器，同时为 L1 和 L2 维护单独的链路状态数据库，也表明所在区域有另一台路由器 R1；

③ R3 和 R4 路由器为 L2 路由器，只维护 L2 的链路状态数据库。

（2）show ip route

```
R1#show ip route isis
i L1 192.168.23.0/24 [115/20] via 192.168.12.2, Serial0/0/0
i*L1 0.0.0.0/0 [115/10] via 192.168.12.2, Serial0/0/0
R2#show ip route isis
     1.0.0.0/24 is subnetted, 1 subnets
i L1    1.1.1.0 [115/20] via 192.168.12.1, Serial0/0/0
     4.0.0.0/22 is subnetted, 1 subnets
i L2    4.4.0.0 [115/30] via 192.168.23.3, Serial0/0/1
i L2 192.168.34.0/24 [115/20] via 192.168.23.3, Serial0/0/1
```

```
R3#show ip route isis
i L2 192.168.12.0/24 [115/20] via 192.168.23.2, Serial0/0/1
     1.0.0.0/24 is subnetted, 1 subnets
i L2    1.1.1.0 [115/30] via 192.168.23.2, Serial0/0/1
     4.0.0.0/22 is subnetted, 1 subnets
i L2    4.4.0.0 [115/20] via 192.168.34.4, Serial0/0/0
i*L2 0.0.0.0/0 [115/10] via 192.168.23.2, Serial0/0/1
R4#show ip route isis
i L2 192.168.12.0/24 [115/30] via 192.168.34.3, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
i L2    1.1.1.0 [115/40] via 192.168.34.3, Serial0/0/0
     4.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
i su    4.4.0.0/22 [115/10] via 0.0.0.0, Null0
i L2 192.168.23.0/24 [115/20] via 192.168.34.3, Serial0/0/0
i*L2 0.0.0.0/0 [115/20] via 192.168.34.3, Serial0/0/0
```

以上输出表明：

① 由于 R1 为 L1 路由器，所以只有“i L1”的路由和一条到最近的 L1/L2 路由器的默认路由“i*L1”；

② 由于 R1 和 R2 在一个区域，所以 R2 既有“i L1”的路由,又有“i L2”的路由；

③ R3 和 R4 都是 L2 路由器，所以只有“i L2”的路由；

④ R3 和 R4 都收到一条由 R2 注入的默认路由“i*L2”；

⑤ R2 和 R3 都收到 R4 的汇总路由，同时 R4 的路由表自动生成一条“i su”的路由条目，主要是为了避免路由环路。

（3）show clns interface

```
R3#show clns interface s0/0/0
Serial0/0/0 is up, line protocol is up
 Checksums enabled, MTU 1500, Encapsulation HDLC
  ERPDUs enabled, min. interval 10 msec.
  CLNS fast switching enabled
  CLNS SSE switching disabled
  DEC compatibility mode OFF for this interface
  Next ESH/ISH in 12 seconds
  Routing Protocol: IS-IS
    Circuit Type: level-2
  ......
```

以上输出可以看到接口的电路类型为“level-2”。

19.4 帧中继上集成 IS-IS

19.4.1 实验 3: NBMA 上集成的 IS-IS

1.实验目的

通过本实验可以掌握

（1）帧中继上 CLNS 映射

（2）在 NBMA 下 IS-IS 的配置和调试

2. 拓扑结构

实验拓扑如图 19-3 所示。

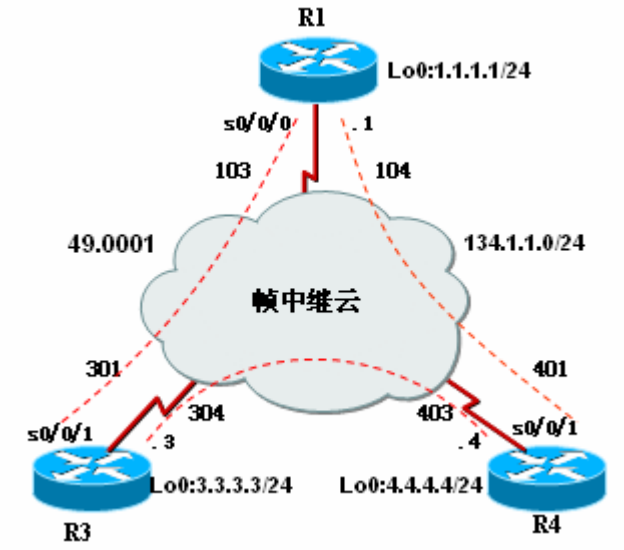

图 19-3 NBMA 上集成 IS-IS

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router isis
R1(config-router)#net 49.0001.1111.1111.1111.00
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-if)#ip router isis
R1(config)#interface Serial0/0/0
R1(config-if)#ip address 134.1.1.1 255.255.255.0
R1(config-if)#encapsulation frame-relay
R1(config-if)#frame-relay map clns 104 broadcast //配置 CLNS 映射
R1(config-if)#frame-relay map clns 103 broadcast
R1(config-if)#frame-relay map ip 134.1.1.3 103 broadcast
R1(config-if)#frame-relay map ip 134.1.1.4 104 broadcast
R1(config-if)#no frame-relay inverse-arp
R1(config-if)#ip router isis
R1(config-if)#no shutdown
```

（2）步骤 2：配置路由器 R3

```
R3(config)#router isis
R3(config-router)#net 49.0001.3333.3333.3333.00
R3(config)#interface Loopback0
R3(config-if)#ip address 3.3.3.3 255.255.255.0
R3(config-if)#ip router isis
R3(config)#interface Serial0/0/1
```

```
R3(config-if)#ip address 134.1.1.3 255.255.255.0
R3(config-if)#encapsulation frame-relay
R3(config-if)#frame-relay map clns 304 broadcast
R3(config-if)#frame-relay map clns 301 broadcast
R3(config-if)#frame-relay map ip 134.1.1.1 301 broadcast
R3(config-if)#frame-relay map ip 134.1.1.4 304 broadcast
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#ip router isis
R3(config-if)#no shutdown
```

（3）步骤 3：配置路由器 R4

```
R4(config)#router isis
R4(config-router)#net 49.0001.4444.4444.4444.00
R4(config)#interface Loopback0
R4(config-if)#ip address 4.4.4.4 255.255.255.0
R4(config-if)#ip router isis
R4(config)#interface Serial0/0/1
R4(config-if)#ip address 134.1.1.4 255.255.255.0
R4(config-if)#encapsulation frame-relay
R4(config-if)#frame-relay map clns 403 broadcast
R4(config-if)#frame-relay map clns 401 broadcast
R4(config-if)#frame-relay map ip 134.1.1.1 401 broadcast
R4(config-if)#frame-relay map ip 134.1.1.3 403 broadcast
R4(config-if)#ip router isis
R4(config-if)#no frame-relay inverse-arp
R4(config-if)#no shutdown
```

4.实验调试

（1）show clns neighbors

```
R1#show clns neighbors
```

System Id	Interface	SNPA	State	Holdtime	Type	Protocol
R3	Se0/0/0	**DLCI 103**	Up	23	L1L2	IS-IS
R4	Se0/0/0	**DLCI 104**	Up	8	L1L2	IS-IS

以上输出表明 NBMA 网络中 IS-IS 的 SNPA 为“DLCI”号码。

（2）show isis database

```
R1#show isis database
IS-IS Level-1 Link State Database:
```

LSPID		LSP Seq Num	LSP Checksum	LSP Holdtime	ATT/P/OL
R1.00-00	*	0x00000012	0x6472	1133	0/0/0
R3.00-00		0x00000011	0xA159	716	0/0/0
R4.00-00		0x00000010	0x414C	1169	0/0/0
R4.02-00		**0x0000000F**	**0xAD40**	**1171**	**0/0/0**

```
IS-IS Level-2 Link State Database:
```

LSPID		LSP Seq Num	LSP Checksum	LSP Holdtime	ATT/P/OL
R1.00-00	*	0x00000014	0x0D6F	1046	0/0/0

```
R3.00-00              0x00000014   0xDEC6      666            0/0/0
R4.00-00              0x00000012   0x8FAC      528            0/0/0
R4.02-00              0x0000000E   0x6116      556            0/0/0
```

以上输出表明在NBMA网络中，需要DIS选举，由于各个接口没有配置优先级，所以R4被选举成DIS。

（3）show ip route

```
R1#show ip route isis
     3.0.0.0/24 is subnetted, 1 subnets
i L1    3.3.3.0 [115/20] via 134.1.1.3, Serial0/0/0
     4.0.0.0/24 is subnetted, 1 subnets
i L1    4.4.4.0 [115/20] via 134.1.1.4, Serial0/0/0
```

以上输出表明路由器R1的路由表中有两条“i L1”的路由，因为它们都在区域“49.0001”中。

（3）show frame-relay map

```
R1#show frame-relay map
Serial0/0/0 (up): CLNS dlci 103(0x67,0x1870), static,
              broadcast,
              CISCO, status defined, active
Serial0/0/0 (up): CLNS dlci 104(0x68,0x1880), static,
              broadcast,
              CISCO, status defined, active
Serial0/0/0 (up): ip 134.1.1.3 dlci 103(0x67,0x1870), static,
              broadcast,
              CISCO, status defined, active
Serial0/0/0 (up): ip 134.1.1.4 dlci 104(0x68,0x1880), static,
              broadcast,
              CISCO, status defined, active
```

以上输出表明接口s0/0/0上即需要CLNS的映射，又需要IP的映射。CLNS的映射是必须的，如果没有CLNS的映射，CLNS的邻居关系都不能建立。

【技术要点】

（1）在主接口和多点子接口下，IP 映射和 CNLS 映射应该分别配置，命令分别是“frame-relay map clns”和“frame-relay map ip”；

（2）在点到点子接口下，命令“frame-relay interface-dlci”同时启动 IP 映射和CNLS映射，所以不需要额外的CLNS映射。

19.4.2 实验4:帧中继上点到点子接口下集成的IS-IS

1.实验目的

通过本实验可以掌握

（1）帧中继上CLNS映射

（2）在帧中继上点到点子接口下IS-IS的配置和调试

2.拓扑结构

实验拓扑如图19-4所示。

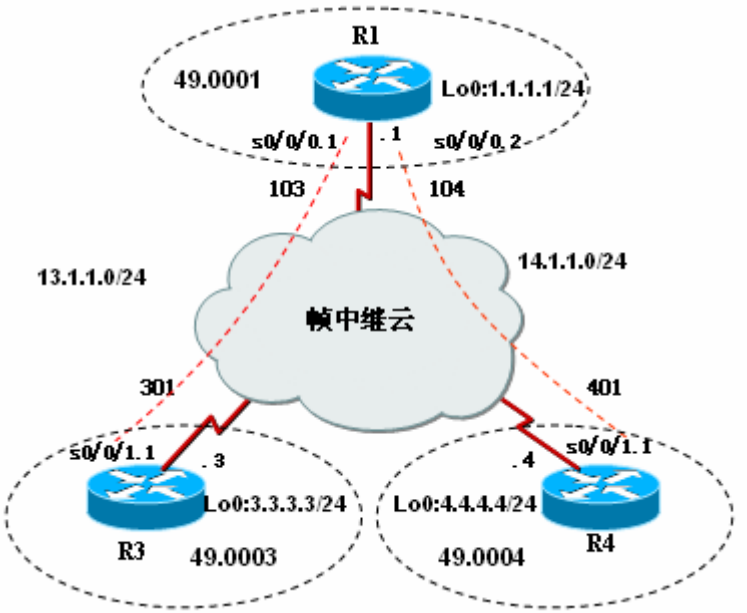

图 19-4 帧中继点到点子接口下集成的 IS-IS

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router isis
R1(config-router)#net 49.0001.1111.1111.1111.00
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-if)#ip router isis
R1(config)#interface Serial0/0/0
R1(config-if)#encapsulation frame-relay
R1(config-if)#no frame-relay inverse-arp
R1(config-if)#no shutdown
R1(config)#interface Serial0/0/0.1 point-to-point
R1(config-subif)#ip address 13.1.1.1 255.255.255.0
R1(config-subif)#frame-relay interface-dlci 103
R1(config-subif)#ip router isis
R1(config)#interface Serial0/0/0.2 point-to-point
R1(config-subif)#ip address 14.1.1.1 255.255.255.0
R1(config-subif)#frame-relay interface-dlci 104
R1(config-subif)#ip router isis
```

（2）步骤 2：配置路由器 R3

```
R3(config)#router isis
R3(config-router)#net 49.0003.3333.3333.3333.00
R3(config)#interface Loopback0
R3(config-if)#ip address 3.3.3.3 255.255.255.0
R3(config-if)#ip router isis
R3(config)#interface Serial0/0/1
```

```
R3(config-if)#encapsulation frame-relay
R3(config-if)#no frame-relay inverse-arp
R3(config-if)#no shutdown
R3(config)#interface Serial0/0/1.1 point-to-point
R3(config-subif)#ip address 13.1.1.3 255.255.255.0
R3(config-subif)#frame-relay interface-dlci 301
R3(config-subif)#ip router isis
```

（3）步骤 3：配置路由器 R4

```
R4(config)#router isis
R4(config-router)#net 49.0004.4444.4444.4444.00
R4(config)#interface Loopback0
R4(config-if)#ip address 4.4.4.4 255.255.255.0
R4(config-if)#ip router isis
R4(config-if)#interface Serial0/0/1
R4(config-if)#encapsulation frame-relay
R4(config-if)#no frame-relay inverse-arp
R4(config-if)#no shutdown
R4(config)#interface Serial0/0/1.1 point-to-point
R4(config-subif)#ip address 14.1.1.4 255.255.255.0
R4(config-subif)#frame-relay interface-dlci 401
R4(config-subif)#ip router isis
```

4.实验调试

（1）show ip route

```
R1#show ip route isis
     3.0.0.0/24 is subnetted, 1 subnets
i L2    3.3.3.0 [115/20] via 13.1.1.3, Serial0/0/0.1
     4.0.0.0/24 is subnetted, 1 subnets
i L2    4.4.4.0 [115/20] via 14.1.1.4, Serial0/0/0.2
```

以上输出表明路由器 R1 的路由表中有两条“i L2”的路由，路由表是正确的。

（2）show frame-relay map

```
R1#show frame-relay map
Serial0/0/0.1 (up): point-to-point dlci, dlci 103(0x67,0x1870), broadcast
      status defined, active
Serial0/0/0.2 (up): point-to-point dlci, dlci 104(0x68,0x1880), broadcast
      status defined, active
```

以上输出表明点到点子接口的映射根本没有区分 IP 或者 CLNS，所以命令“frame-relay interface-dlci”同时启动 IP 映射和 CNLS 映射。

19.5 IS-IS 命令汇总

表 19-1 列出了本章涉及到的主要的命令。

表 19-1 本章命令汇总

命令	作用
show clns neighbors	查看 CLNS 邻居

show clns protocols	查看 CLNS 路由协议相关的信息
show clns interface	查看 CLNS 接口状态的信息
show clns route	查看 CLNS L2 路由
clear clns route	清除 CLNS 路由表
clear isis *	清除 IS-IS 链路状态数据库
show clns traffic	查看 CLNS 协议的统计信息
show isis hostname	查看主机名和系统 ID 的动态对应关系
show isis database	查看 IS-IS 链路状态数据库
show isis topology	查看 IS-IS 拓扑结构信息
show isis route	查看 CLNS L1 的路由表
show frame-relay map	查看帧中继映射
show ip protocols	查看和 IP 路由协议相关的信息
router isis	启动 IS-IS 路由进程
net	配置 NET 地址
ip router isis	接口下启用 IS-IS
is-type	配置 IS-IS 路由器类型
area-password	配置区域认证
isis password	配置邻居认证
domain-password domain	配置域认证
default-information originate	向 IS-IS 网络注入默认路由
summary-address	配置区域间路由汇总
isis circuit-type	配置接口电路类型
frame-relay map clns	配置 CLNS 映射

第 20 章 路由重分布

当许多运行多路由的网络要集成到一起时，必须在这些不同的路由选择协议之间共享路由信息。在路由选择协议之间交换路由信息的过程被称为路由重分布（Route Redistribution）。

20.1 路由重分布概述

路由重分布为在同一个互联网络中高效地支持多种路由协议提供了可能，执行路由重分布的路由器被称为边界路由器，因为它们位于两个或多个自治系统的边界上。

路由重分布时计量单位和管理距离是必须要考虑的。每一种路由协议都有自己度量标准，所以在进行重分布时必须转换度量标准，使得它们兼容。种子度量值（seed metric）是定义在路由重分布里的，它是一条从外部重分布进来的路由的初始度量值。路由协议默认的种子度量值如表 20-1 所示。

表 20-1 路由协议默认的种子度量值

路由协议	默认种子度量值
RIP	无限大
EIGRP	无限大
OSPF	BGP 为 1,其它为 20
IS-IS	0
BGP	IGP 的度量值

路由重分布应该考虑到如下的一些问题：

1．路由环路

路由器有可能从一个自治系统学到的路由信息发送回该自治系统，特别是在做双向重分布的时候，一定要注意；

2．路由信息的兼容问题

每一种路由协议的度量标准不同，所以路由器通过重分布所选择的路径可能并非最佳路径；

3．不一致的收敛时间

因为不同的路由协议收敛的时间不同。

20.2 实验 1：RIP、EIGRP 和 OSPF 重分布

1.实验目的

通过本实验可以掌握

（1）种子度量值的配置

（2）路由重分布参数的配置

（3）静态路由重分布

（4）RIP 和 EIGRP 的重分布

（5）EIGRP 和 OSPF 的重分布

（6）重分布路由的查看和调试

2.拓扑结构

实验拓扑如图 20-1 所示。

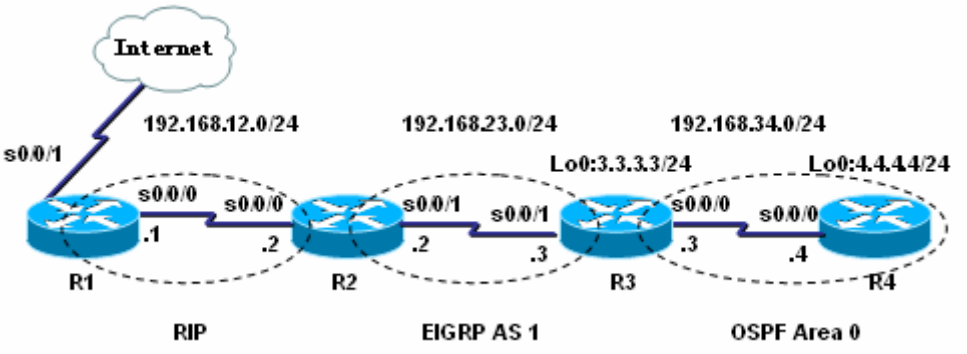

图 20-1 RIP、EIGRP 和 OSPF 重分布

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 192.168.12.0
R1(config-router)#redistribute static metric 3 //重分布静态路由
R1(config)#ip route 0.0.0.0 0.0.0.0 Serial0/0/1
```

【注意】

在向 RIP 区域重分布路由的时候，必须指定度量值，或者通过“default-metric”命令设置缺省种子度量值，因为 RIP 默认种子度量值为无限大，但是只有重分布静态特殊，可以不指定种子度量值。

（2）步骤 2：配置路由器 R2

```
R2(config)#router eigrp 1
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.23.0
R2(config-router)#redistribute rip metric 1000 100 255 1 1500
//将 RIP 重分布到 EIGRP 中
```

【提示】

因为 EIGRP 的度量相对复杂，所以重分布时需要分别指定带宽、延迟、可靠性、负载以及 MTU 参数的值。

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 192.168.12.0
R2(config-router)#redistribute eigrp 1 //将 EIGRP 重分布到 RIP 中
R2(config-router)#default-metric 4      //配置默认种子度量值
```

【注意】

在“redistribute”命令中用参数“metric”指定的种子度量值优先于路由模式下使用“default-metric”命令设定的缺省的种子度量值。

（3）步骤 3：配置路由器 R3

```
R3(config)#router eigrp 1
R3(config-router)#no auto-summary
R3(config-router)#network 3.3.3.0 0.0.0.255
R3(config-router)#network 192.168.23.0
R3(config-router)#redistribute ospf 1 metric 1000 100 255 1 1500
//将 OSPF 重分布到 EIGRP 中
R3(config-router)#distance eigrp 90 150 //配置 EIGRP 默认管理距离
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 192.168.34.0 0.0.0.255 area 0
R3(config-router)#redistribute eigrp 1 metric 30 metric-type 1 subnets
//将 EIGRP 重分布到 OSPF 中
R3(config-router)#default-information originate always
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 0.0.0.255 area 0
R4(config-router)#network 192.168.34.0 0.0.0.255 area 0
```

4.实验调试

（1）在 R1 上查看路由表：

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 0.0.0.0 to network 0.0.0.0

C    192.168.12.0/24 is directly connected, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
R       3.3.3.0 [120/4] via 192.168.12.2, 00:00:08, Serial0/0/0
     4.0.0.0/32 is subnetted, 1 subnets
R       4.4.4.4 [120/4] via 192.168.12.2, 00:00:08, Serial0/0/0
C    202.96.134.0/24 is directly connected, Serial0/0/1
R    192.168.23.0/24 [120/4] via 192.168.12.2, 00:00:08, Serial0/0/0
R    192.168.34.0/24 [120/4] via 192.168.12.2, 00:00:08, Serial0/0/0
S*   0.0.0.0/0 is directly connected, Serial0/0/1
```

以上输出表明路由器 R1 通过 RIPv2 学到从路由器 R2 重分布进 RIP 的路由。

（2）在 R2 上查看路由表：

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 192.168.12.1 to network 0.0.0.0

C    192.168.12.0/24 is directly connected, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
D       3.3.3.0 [90/2297856] via 192.168.23.3, 00:00:21, Serial0/0/1
     4.0.0.0/32 is subnetted, 1 subnets
D EX    4.4.4.4 [170/3097600] via 192.168.23.3, 00:00:21, Serial0/0/1
C    192.168.23.0/24 is directly connected, Serial0/0/1
D EX 192.168.34.0/24 [170/3097600] via 192.168.23.3, 00:00:21, Serial0/0/1
R*   0.0.0.0/0 [120/3] via 192.168.12.1, 00:00:05, Serial0/0/0
```

以上输出表明从路由器 R1 上重分布进 RIP 的默认路由被路由器 R2 学习到，路由代码为“R*”；在路由器 R3 上重分布进来的 OSPF 路由也被路由器 R2 学习到，路由代码为“D EX”，这也说明 EIGRP 能够识别内部路由和外部路由，默认的时候，内部路由的管理距离是 90，外部路由的管理距离是 170。

（3）在 R3 上查看路由表：

```
R3#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 192.168.23.2 to network 0.0.0.0

D EX 192.168.12.0/24 [150/3097600] via 192.168.23.2, 00:13:43, Serial0/0/1
     3.0.0.0/24 is subnetted, 1 subnets
C       3.3.3.0 is directly connected, Loopback0
     4.0.0.0/32 is subnetted, 1 subnets
O       4.4.4.4 [110/65] via 192.168.34.4, 00:13:43, Serial0/0/0
C    192.168.23.0/24 is directly connected, Serial0/0/1
```

```
C    192.168.34.0/24 is directly connected, Serial0/0/0
D*EX 0.0.0.0/0 [150/3097600] via 192.168.23.2, 00:06:08, Serial0/0/1
```

以上输出表明，从路由器 R2 上重分布进 EIGRP 的路由被路由器 R3 学习到，路由代码为“D*EX”，同时 EIGRP 外部路由的管理距离被修改成 150。

（4）在 R4 上查看路由表：

```
R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 192.168.34.3 to network 0.0.0.0

O E1 192.168.12.0/24 [110/94] via 192.168.34.3, 00:25:26, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
O E1    3.3.3.0 [110/94] via 192.168.34.3, 00:25:26, Serial0/0/0
     4.0.0.0/24 is subnetted, 1 subnets
C       4.4.4.0 is directly connected, Loopback0
O E1 192.168.23.0/24 [110/94] via 192.168.34.3, 00:25:26, Serial0/0/0
C    192.168.34.0/24 is directly connected, Serial0/0/0
O*E2 0.0.0.0/0 [110/1] via 192.168.34.3, 00:25:26, Serial0/0/0
```

以上输出表明，从路由器 R3 上重分布进 OSPF 的路由被路由器 R4 学习到，路由代码为“O E1”；同时学到由 R3 注入的路由代码为“O E2”的默认路由。

（5）show ip protocols

```
R3#show ip protocols
Routing Protocol is "eigrp 1" //  运行 AS 为 1 的 EIGRP 进程
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
  Redistributing: eigrp 1, ospf 1 (internal, external 1 & 2, nssa-external 1 & 2)
  //将 OSPF 进程 1 重分布 EIGRP 中
  EIGRP NSF-aware route hold timer is 240s
  Automatic network summarization is not in effect
  Maximum path: 4
  Routing for Networks:
    3.3.3.0/24
    192.168.23.0
```

```
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.23.2          90      00:51:05
  Distance: internal 90 external 150

Routing Protocol is "ospf 1" //运行 OSPF 进程，进程号为 1
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 3.3.3.3
  It is an autonomous system boundary router //自治系统边界路由器（ASBR）
  Redistributing External Routes from,
    eigrp 1 with metric mapped to 30, includes subnets in redistribution
//将 EIGRP1 重分布 OSPF 中
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    192.168.34.0 0.0.0.255 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
    4.4.4.4              110      00:58:42
    3.3.3.3              110      00:58:42
  Distance: (default is 110)
```

以上输出表明路由器 R3 运行 EIGRP 和 OSPF 两种路由协议，而且实现了双向重分布。

20.3 实验 2：ISIS 和 OSPF 重分布

1. 实验目的

通过本实验可以掌握

（1）直连路由的重分布

（2）IS-IS 和 OSPF 的重分布

（3）重分布路由的查看和调试

2. 拓扑结构

实验拓扑如图 20-2 所示。

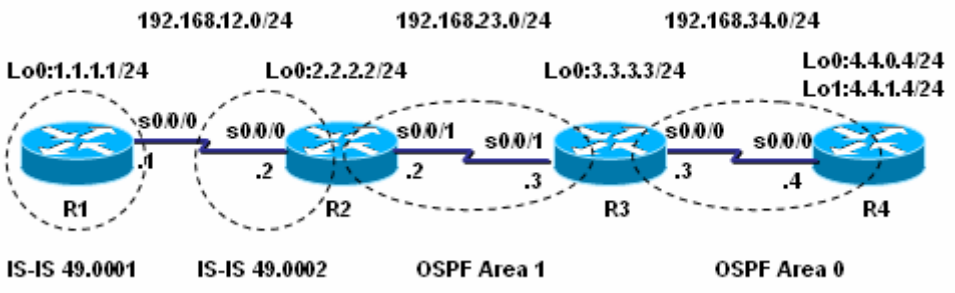

图 20-2 ISIS 和 OSPF 重分布

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router isis
R1(config-router)#net 49.0001.1111.1111.1111.00
R1(config-router)#is-type level-2-only
R1(config)#interface Loopback0
R1(config-if)#ip address 1.1.1.1 255.255.255.0
R1(config-if)#ip router isis
R1(config)#interface Serial0/0/0
R1(config-if)#ip address 192.168.12.1 255.255.255.0
R1(config-if)#ip router isis
R1(config-if)#no shutdown
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router isis
R2(config-router)#net 49.0002.2222.2222.2222.00
R2(config-router)#is-type level-2-only
R2(config-router)#redistribute ospf 1 metric 20 //将 OSPF 重分布到 IS-IS 中
R2(config)#interface Loopback0
R2(config-if)#ip address 2.2.2.2 255.255.255.0
R2(config-if)#ip router isis
R2(config)#interface Serial0/0/0
R2(config-if)#ip address 192.168.12.2 255.255.255.0
R2(config-if)#clockrate 128000
R2(config-if)#ip router isis
R2(config-if)#no shutdown
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 192.168.23.0 0.0.0.255 area 1
R2(config-router)#redistribute isis level-2 metric 50 subnets
//将 IS-IS 重分布到 OSPF 中
R2(config-router)#redistribute connected subnets
//将直连重分布到 OSPF 中
```

【技术要点】

在重分布 IS-IS 路由协议的时候，只能将 L1 和 L2 的路由重分布进来，而运行 IS-IS 路由协议的本地接口是不能被重分布进来的，要通过重分布直连才可以。本实验中，如果不重分布直连，那么 R3 和 R4 的路由表中将没有“192.168.12.0”的路由条目，造成局部网络不可达。

（3）步骤 3：配置路由器 R3

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 3.3.3.0 0.0.0.255 area 0
R3(config-router)#network 192.168.23.0 0.0.0.255 area 1
R3(config-router)#network 192.168.34.0 0.0.0.255 area 0
```

（4）步骤 4：配置路由器 R4

```
R4(config)#ip prefix-list 1 seq 5 permit 4.4.0.0/24 //定义前缀列表
R4(config)#ip prefix-list 2 seq 5 permit 4.4.1.0/24
R4(config)#route-map conn permit 10 //定义策略，21 章详细介绍
R4(config-route-map)#match ip address prefix-list 1 //匹配条件
R4(config-route-map)#set metric 50                  //执行行为
R4(config-route-map)#set metric-type type-1
R4(config)#route-map conn permit 20
R4(config-route-map)#match ip address prefix-list 2
R4(config-route-map)#set metric 100
R4(config)#router ospf 1
R4(config-router)#router-id 4.4.4.4
R4(config-router)#network 192.168.34.0 0.0.0.255 area 0
R4(config-router)#redistribute connected subnets route-map conn
//将直连重分布到 OSPF 中
```

【说明】

（1）路由器 R4 重分布直连的环回接口时，对“4.4.0.0”做的控制是种子度量值设为 50，度量值的类型为 1，而对“4.4.1.0”做的控制是种子度量值设为 100，度量值的类型采用默认，即类型 2；

（2）由于要重分布直连接口，所以一定不能在 OSPF 的路由进程中通告“4.4.0.0”和“4.4.1.0”；

（3）前缀列表（prefix-list）在路由过滤和路由控制中使用非常的广泛，它比访问控制列表具有更大的灵活性、匹配更加精确。

4.实验调试

（1）在 R1 上查看路由表：

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     2.0.0.0/24 is subnetted, 1 subnets
i L2    2.2.2.0 [115/20] via 192.168.12.2, Serial0/0/0
     3.0.0.0/32 is subnetted, 1 subnets
```

```
i L2    3.3.3.3 [115/30] via 192.168.12.2, Serial0/0/0
     4.0.0.0/24 is subnetted, 2 subnets
i L2    4.4.0.0 [115/30] via 192.168.12.2, Serial0/0/0
i L2    4.4.1.0 [115/30] via 192.168.12.2, Serial0/0/0
i L2 192.168.23.0/24 [115/30] via 192.168.12.2, Serial0/0/0
i L2 192.168.34.0/24 [115/30] via 192.168.12.2, Serial0/0/0
```

以上输出表明路由器R1学到整个网络的路由信息，其中路由条目“2.2.2.0”是IS-IS内部路由,而其它的“i L2”路由条目全部是通过路由器R2重分布进来的。

（2）在R2上查看路由表：

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
i L2    1.1.1.0 [115/20] via 192.168.12.1, Serial0/0/0
     2.0.0.0/24 is subnetted, 1 subnets
C       2.2.2.0 is directly connected, Loopback0
     3.0.0.0/32 is subnetted, 1 subnets
O IA    3.3.3.3 [110/65] via 192.168.23.3, 00:00:27, Serial0/0/1
     4.0.0.0/24 is subnetted, 2 subnets
O E1    4.4.0.0 [110/178] via 192.168.23.3, 00:00:27, Serial0/0/1
O E2    4.4.1.0 [110/100] via 192.168.23.3, 00:00:27, Serial0/0/1
C    192.168.23.0/24 is directly connected, Serial0/0/1
O IA 192.168.34.0/24 [110/128] via 192.168.23.3, 00:00:27, Serial0/0/1
```

以上输出表明路由器R2既学到了“i L2”的路由，又学到OSPF的“O IA”的路由，也学到从R4重分布进来的“O E1”和“O E2”路由。特别是对于R4两个环回接口的路由条目，确实达到了预期的控制要求。

（3）在R3上查看路由表：

```
R3#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
```

```
Gateway of last resort is not set

O E2 192.168.12.0/24 [110/20] via 192.168.23.2, 00:01:27, Serial0/0/1
     1.0.0.0/24 is subnetted, 1 subnets
O E2    1.1.1.0 [110/50] via 192.168.23.2, 00:01:23, Serial0/0/1
     2.0.0.0/24 is subnetted, 1 subnets
O E2    2.2.2.0 [110/20] via 192.168.23.2, 00:01:27, Serial0/0/1
     3.0.0.0/24 is subnetted, 1 subnets
C       3.3.3.0 is directly connected, Loopback0
     4.0.0.0/24 is subnetted, 2 subnets
O E1    4.4.0.0 [110/114] via 192.168.34.4, 00:01:28, Serial0/0/0
O E2    4.4.1.0 [110/100] via 192.168.34.4, 00:01:28, Serial0/0/0
C    192.168.23.0/24 is directly connected, Serial0/0/1
C    192.168.34.0/24 is directly connected, Serial0/0/0
```

以上输出值得注意的是路由条目“192.168.12.0”和“2.2.2.0”，如果在路由器 R2 上 OSPF 重分布的时候，没有将直连接口重分布进来，那么路由器 R3 是不能收到这些路由条目的。

（4）在 R4 上查看路由表：

```
R4#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

O E2 192.168.12.0/24 [110/20] via 192.168.34.3, 00:01:42, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
O E2    1.1.1.0 [110/50] via 192.168.34.3, 00:01:41, Serial0/0/0
     2.0.0.0/24 is subnetted, 1 subnets
O E2    2.2.2.0 [110/20] via 192.168.34.3, 00:01:42, Serial0/0/0
     3.0.0.0/32 is subnetted, 1 subnets
O       3.3.3.3 [110/65] via 192.168.34.3, 00:01:42, Serial0/0/0
     4.0.0.0/24 is subnetted, 2 subnets
C       4.4.0.0 is directly connected, Loopback0
C       4.4.1.0 is directly connected, Loopback1
O IA 192.168.23.0/24 [110/128] via 192.168.34.3, 00:01:42, Serial0/0/0
C    192.168.34.0/24 is directly connected, Serial0/0/0
```

以上输出表明路由器 R4 既学到了 OSPF 的“O”和“O IA”的路由，也学到从 R2 重分布进来的 “O E2”路由。

20.4 路由重分布命令汇总

表 20-2 列出了本章涉及到的主要的命令。

表 20-2 本章命令汇总

命令	作用
show ip route	查看路由表
show ip protocols	查看和路由协议相关的信息
redistribute	配置路由协议重分布
default-metric	配置默认种子度量值
ip prefix-list	定义前缀列表
distance eigrp	配置 EIGRP 默认管理距离

第 21 章　路由优化

在当今高性能的网络中，为了保证网络的伸缩性、稳定性、安全性和快速收敛，必须对网络进行优化。路由过滤和策略路由是路由优化的常用方法。

21.1　路由优化概述

路由过滤是指在路由更新中抑制某些路由不被发送和接收，被动接口、分布控制列表、重分布结合路由策略等都可以实现路由过滤。

策略路由提供了根据网络管理者制定的标准来进行数据包转发的一种机制。基于策略的路由比传统路由能力更强，使用更灵活，它使网络管理者不仅能够根据目的地址而且能够根据协议类型、报文大小、应用或 IP 源地址来选择转发路径。策略路由的策略由路由映射图（route map）来定义。“route map”命令中最为重要的是“match”和“set”。

match 用来定义匹配的条件，匹配语句在路由器的输入端口对数据包进行检测。常用的匹配条件包括 IP 地址、接口、度量值以及数据包长度等。

set 定义对符合匹配条件的语句采取的行为。通常的行为如表 21-1 所示。

表 21-1 set 的行为

set 行为	描述
set ip next hop	设定数据包的下一跳地址
set interface	设定数据包出接口
set ip default next hop	设定缺省的下一跳地址，用于当路由表里没有到数据包目的地址路由条目时
set default interface	设定缺省的出接口
set ip tos	设定 IP 数据包的 IP ToS 值
set ip precedence	设定 IP 数据包的优先级

【注意】

1. 一个 route map 的最后默认“deny any”。这个 deny 的使用结果依赖于这个 route map 是怎样使用的。如果一个数据包对于 route map 没有匹配项，它会按照正常的目的地址路由转发，而对于路由条目如果 route map 没有匹配项，则被拒绝；

2. 一个 route map 可以包含多个 route map 陈述，这些语句的执行顺序像 ACL 一样，从上到下被执行。

21.2 实验 1:用分布控制列表控制路由更新

1.实验目的

通过本实验可以掌握

（1）被动接口的配置

（2）分布控制列表的配置

2.拓扑结构

实验拓扑如图 21-1 所示。

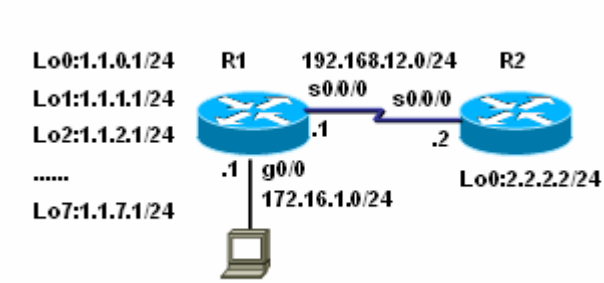

图 21-1 用分布控制列表控制路由更新

本实验通过使用分布控制列表控制路由器 R1 只发送环回接口中第三位为奇数的路由和 g0/0 接口的路由更新给 R2，整个网络运行 RIPv2 路由协议。

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#access-list 1 permit 172.16.1.0
R1(config)#access-list 1 permit 1.1.1.0 0.0.254.0 //允许第三位为奇数的路由
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 1.0.0.0
R1(config-router)#network 172.16.0.0
R1(config-router)#network 192.168.12.0
R1(config-router)#passive-interface default        //默认为被动接口
R1(config-router)#no passive-interface Serial0/0/0 //关闭被动接口
R1(config-router)#distribute-list 1 out Serial0/0/0//出方向配置分布控制列表
```

【注意】

“distribute-list”命令可以全局的在一个出或入方向的路由更新中过滤路由，也可以为一个路由进程所涉及到的每一个接口的入方向或出方向设置路由过滤。

（2）步骤 2：配置路由器 R2

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto-summary
R2(config-router)#network 2.0.0.0
R2(config-router)#network 192.168.12.0
```

4. 实验调试

（1）在 R2 上查看路由表：

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
```

```
        * - candidate default, U - per-user static route, o - ODR
        P - periodic downloaded static route

Gateway of last resort is not set

C    192.168.12.0/24 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 4 subnets
R       1.1.1.0 [120/1] via 192.168.12.1, 00:00:01, Serial0/0/0
R       1.1.3.0 [120/1] via 192.168.12.1, 00:00:01, Serial0/0/0
R       1.1.5.0 [120/1] via 192.168.12.1, 00:00:01, Serial0/0/0
R       1.1.7.0 [120/1] via 192.168.12.1, 00:00:01, Serial0/0/0
     2.0.0.0/24 is subnetted, 1 subnets
C       2.2.2.0 is directly connected, Loopback0
     172.16.0.0/24 is subnetted, 1 subnets
R       172.16.1.0 [120/1] via 192.168.12.1, 00:00:01, Serial0/0/0
```

以上输出表明路由器 R2 只收到 R1 的以太口和第三位为奇数的环回接口的路由。

（2）show ip protocols

```
R1#show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 2 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
    Serial0/0/0 filtered by 1 (per-user), default is 1
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 2, receive version 2
......
```

以上输出信息表明，全局下没有作用分布控制列表，在 s0/0/0 接口的出方向作用了分布控制列表 1。

21.3 策略路由

21.3.1 实验 2：基于源 IP 地址的策略路由

1.实验目的

通过本实验可以掌握

（1）用 route-map 定义路由策略

（2）在接口下应用路由策略

（3）基于源 IP 地址的策略路由的调试

2.拓扑结构

实验拓扑如图 21-2 所示。

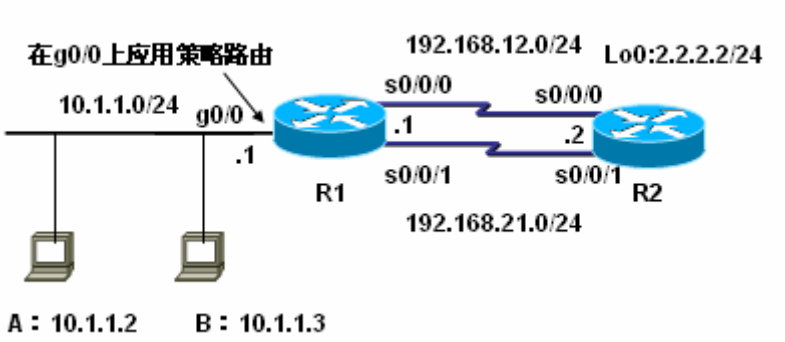

图 21-2 基于源 IP 地址的策略路由

本实验设计如下：在路由器 R1 的 g0/0 接口应用 IP 策略路由 CCNA,使得从主机 A 来的数据设置下一跳地址为 192.168.12.2;从主机 B 来的数据设置下一跳地址为 192.168.21.2,所有其它的数据包正常转发，整个网络运行 EIGRP 路由协议。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#access-list 1 permit 10.1.1.2
R1(config)#access-list 2 permit 10.1.1.3
R1(config)#route-map CCNA permit 10
R1(config-route-map)#match ip address 1
R1(config-route-map)#set ip next-hop 192.168.12.2
R1(config)#route-map CCNA permit 20
R1(config-route-map)#match ip address 2
R1(config-route-map)#set ip next-hop 192.168.21.2
R1(config)#interface g0/0
R1(config-if)#ip policy route-map CCNA
R1(config)#router eigrp 1
R1(config-router)#no auto-summary
R1(config-router)#network 10.1.1.0 255.255.255.0
R1(config-router)#network 192.168.12.0
R1(config-router)#network 192.168.21.0
```

4.实验调试

（1）主机 A 上 ping 地址 2.2.2.2，路由器 R1 上显示的调试信息如下：

```
R1#debug ip policy
*Feb  20  00:18:33.368: IP: s=10.1.1.2 (GigabitEthernet0/0), d=2.2.2.2, len 60, policy match
*Feb  20  00:18:33.368: IP: route map CCNA, item 10, permit
*Feb 20 00:18:33.372: IP: s=10.1.1.2 (GigabitEthernet0/0), d=2.2.2.2 (Serial0/0/0), len 60, policy routed
*Feb 20 00:18:36.381: IP: GigabitEthernet0/0 to Serial0/0/0 192.168.12.2
```

以上输出信息表明源地址为 10.1.1.2 的主机发送给目的主机 2.2.2.2 的数据包在接口“GigabitEthernet0/0”匹配 route-map CCNA 的序列号 10 所定义的策略，执行策略路由，设置数据包下一跳地址为 192.168.12.2。

（2）主机 B 上 ping 地址 2.2.2.2，路由器 R1 上显示的调试信息如下：

*Feb 20 00:21:12.449: IP: s=10.1.1.3 (GigabitEthernet0/0), d=2.2.2.2, len 60, **policy match**

*Feb 20 00:21:12.449: IP: route map **CCNA, item 20, permit**

*Feb 20 00:21:12.453: IP: s=10.1.1.3 (GigabitEthernet0/0), d=2.2.2.2 (Serial0/0/1), len 60, **policy routed**

*Feb 20 00:21:12.453: IP: GigabitEthernet0/0 to **Serial0/0/1 192.168.21.2**

数以上输出信息表明源地址为 10.1.1.3 的主机发送给目的主机 2.2.2.2 数据包在接口“GigabitEthernet0/0”匹配 route-map CCNA 的序列号 20 所定义的策略，执行策略路由，设置数据包下一跳地址为 192.168.21.2。

（3）在主机 10.1.1.6 上 ping 地址 2.2.2.2，路由器 R1 上显示的调试信息如下：

*Feb 20 00:14:17.416: IP: s=10.1.1.6 (GigabitEthernet0/0), d=2.2.2.2 (Serial0/0/0), len 60, **policy rejected -- normal forwarding**

以上输出信息表明源地址为 10.1.1.6 的主机发送到目的主机 2.2.2.2 数据包在接口“GigabitEthernet0/0”不匹配路由策略，数据包正常转发。

（4）**show ip policy**

该命令显示了在哪些接口上应用了哪些策略。

```
R1#show ip policy
Interface       Route map
Gi0/0           CCNA
```

以上输出信息表明在“Gi0/0”接口应用了路由策略“CCNA”。

21.3.2 实验 3：基于报文大小的策略路由

1.实验目的

通过本实验可以掌握

（1）用 route-map 定义路由策略

（2）在接口下应用路由策略

（3）基于报文大小的策略路由的调试

2.拓扑结构

实验拓扑如图 21-3 所示。

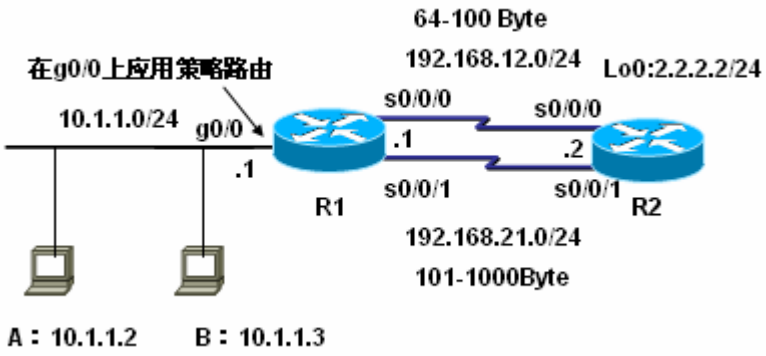

图 21-3 基于报文大小的策略路由

本实验设计如下：在路由器 R1 的 g0/0 接口应用 IP 策略路由 CCNP，使得对大小为 64-100 字节的数据包设置出接口为 s0/0/0；大小为 101-1000 字节的数据包设置出接口为 s0/0/1，所有其它的数据包正常转发，整个网络运行 EIGRP 路由协议。

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#route-map CCNP permit 10
R1(config-route-map)#match length 64 100
R1(config-route-map)#set interface s0/0/0
R1(config)#route-map CCNP permit 20
R1(config-route-map)#match length 101 1000
R1(config-route-map)#set interface s0/0/1
R1(config)#interface g0/0
R1(config-if)#ip policy route-map CCNP
R1(config)#ip local policy route-map CCNP
R1(config)#router eigrp 1
R1(config-router)#no auto-summary
R1(config-router)#network 10.1.1.0 255.255.255.0
R1(config-router)#network 192.168.12.0
R1(config-router)#network 192.168.21.0
```

【注意】

在接口下应用的策略路由对路由器本地产生的数据包不起作用。如果需要对本地路由器产生的数据包执行策略路由，要用“ip local policy route-map”命令配置本地策略。

4.实验调试

（1）执行扩展 ping 命令，数据包的长度为 90,源地址为 10.1.1.6（路由器 R3 的以太口地址）：

```
R3#ping
Protocol [ip]:
Target IP address: 2.2.2.2
Repeat count [5]: 1
Datagram size [100]: 90
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 10.1.1.6
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1, 90-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.6
```

路由器 R1 上显示的调试信息如下：

```
R1#debug ip policy
*Feb 20 20:57:06.535: IP: s=10.1.1.6 (GigabitEthernet0/0), d=2.2.2.2, len 90, FIB policy
```

match

*Feb 20 20:57:06.535: IP: s=10.1.1.6 (GigabitEthernet0/0), d=2.2.2.2 (**Serial0/0/0**), len 90, **FIB policy routed**

以上输出信息表明长度为 90 字节的数据包在接口“GigabitEthernet0/0”匹配策略，执行策略路由，设置数据包出接口为 s0/0/0。

（2）执行扩展 ping 命令，数据包的长度为 300，源地址为 10.1.1.6：

```
R3#ping
Protocol [ip]:
Target IP address: 2.2.2.2
Repeat count [5]: 1
Datagram size [100]: 300
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 10.1.1.6
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1, 300-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.6
```

路由器 R1 上显示的调试信息如下：

*Feb 20 20:58:39.311: IP: s=10.1.1.6 (GigabitEthernet0/0), d=2.2.2.2, **len 300, FIB policy match**

*Feb 20 20:58:39.311: IP: s=10.1.1.6 (GigabitEthernet0/0), d=2.2.2.2 (**Serial0/0/1**), len 300, **FIB policy routed**

以上输出信息表明长度为 300 字节的数据包在接口“GigabitEthernet0/0”匹配策略，执行策略路由，设置数据包出接口为 s0/0/1。

（3）执行扩展 ping 命令，数据包的长度为 1200，源地址为 10.1.1.6：

```
R3#ping
Protocol [ip]:
Target IP address: 2.2.2.2
Repeat count [5]: 1
Datagram size [100]: 1200
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 10.1.1.6
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
```

```
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1, 1200-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
Packet sent with a source address of 10.1.1.6
```

路由器 R1 上显示的调试信息如下：

```
*Feb 20 21:01:20.583: IP: s=10.1.1.6 (GigabitEthernet0/0), d=2.2.2.2, len 1200, FIB policy rejected(no match) - normal forwarding
```

以上输出信息表明长度为 1200 字节的数据包在接口“GigabitEthernet0/0”不匹配路由策略，数据包正常转发。

（4）show route-map

该命令显示定义的所有路由策略及路由策略匹配的情况。

```
R1#show route-map
route-map CCNP, permit, sequence 10
  Match clauses:
    length 64 100
  Set clauses:
    interface Serial0/0/0
  Policy routing matches: 3 packets, 292 bytes
route-map CCNP, permit, sequence 20
  Match clauses:
    length 101 1000
  Set clauses:
    interface Serial0/0/1
  Policy routing matches: 1 packets, 314 bytes
```

（5）show ip policy

```
R1#show ip policy
Interface      Route map
local          CCNP
Gi0/0          CCNP
```

以上输出信息表明在接口“Gi0/0”和本地应用了路由策略“CCNP”。

21.3.2 实验 4：基于应用的策略路由

1.实验目的

通过本实验可以掌握

（1）用 route-map 定义路由策略

（2）在接口下应用路由策略

（3）基于应用的策略路由的调试

2.拓扑结构

实验拓扑如图 21-4 所示。

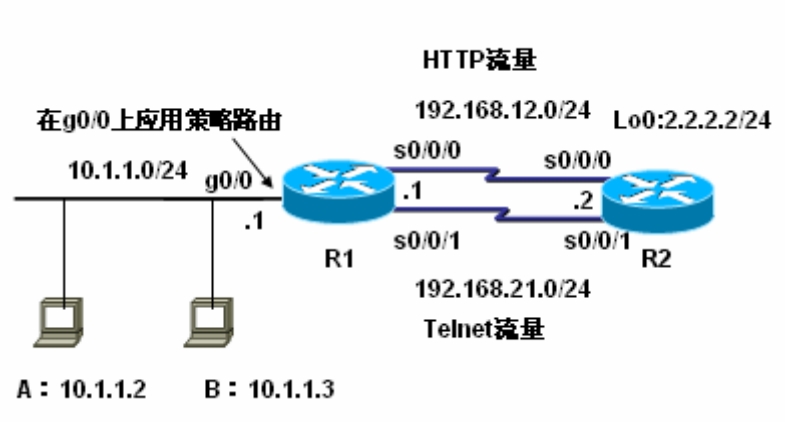

图 21-4 基于应用的策略路由

本实验设计如下：在路由器 R1 的 g0/0 接口应用 IP 策略路由 CCIE，使得对 HTTP 数据包设置下一跳地址为 192.168.12.2，并且设置 IP 数据包优先级为 flash，为 Telnet 数据包设置下一跳地址为 192.168.21.2， 并且设置 IP 数据包优先级为 critical，所有其它的数据包正常转发，整个网络运行 EIGRP 路由协议。

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#ip access-list extended HTTP
R1(config-ext-nacl)#permit tcp any any eq 80
R1(config)#ip access-list extended TELNET
R1(config-ext-nacl)#permit tcp any any eq 23
R1(config)#route-map CCIE permit 10
R1(config-route-map)#match ip address HTTP
R1(config-route-map)#set ip precedence flash
R1(config-route-map)#set ip next-hop 192.168.12.2
R1(config)#route-map CCIE permit 20
R1(config-route-map)#match ip address TELNET
R1(config-route-map)#set ip precedence critical
R1(config-route-map)#set ip next-hop 192.168.21.2
R1(config)#interface g0/0
R1(config-if)#ip policy route-map CCIE
R1(config)#ip local policy route-map CCIE
R1(config)#router eigrp 1
R1(config-router)#no auto-summary
R1(config-router)#network 10.1.1.0 255.255.255.0
R1(config-router)#network 192.168.12.0
R1(config-router)#network 192.168.21.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router eigrp 1
R2(config-router)#network 2.2.2.0 0.0.0.255
R2(config-router)#network 192.168.12.0
R2(config-router)#network 192.168.21.0
```

R2(config-router)#no auto-summary
R2(config)#ip http server
R2(config)#line vty 0 4
R2(config-line)#no login
R2(config-line)#privilege level 15

4.实验调试

（1）在主机 A 上访问 2.2.2.2 的 http 服务,路由器 R1 上显示的调试信息如下：

R1#debug ip policy

*Feb 20 22:32:44.407: IP: s=10.1.1.2 (GigabitEthernet0/0), d=2.2.2.2, len 48, **FIB policy match**

*Feb 20 22:32:44.407: IP: s=10.1.1.2 (GigabitEthernet0/0), d=2.2.2.2, **g=192.168.12.2**, len 48, **FIB policy routed**

以上输出信息表明 http 的数据包在接口“GigabitEthernet0/0”匹配策略，执行策略路由，设置数据包下一跳地址为 192.168.12.2。

（2）在主机 A 上访问 2.2.2.2 的 telnet 服务,路由器 R1 上显示的调试信息如下：

*Feb 20 22:29:28.127: IP: s=10.1.1.2 (GigabitEthernet0/0), d=2.2.2.2, len 43, **FIB policy match**

*Feb 20 22:29:28.131: IP: s=10.1.1.2 (GigabitEthernet0/0), d=2.2.2.2, **g=192.168.21.2**, len 43, **FIB policy routed**

以上输出信息表明 telnet 的数据包在接口“GigabitEthernet0/0”匹配策略，执行策略路由，设置数据包下一跳地址为 192.168.21.2。

（3）在主机 A 上 ping 2.2.2.2,路由器 R1 上显示的调试信息如下：

*Feb 20 22:35:12.431: IP: s=10.1.1.2 (FastEthernet0/0), d=2.2.2.2, len 100, **FIB policy rejected(no match) - normal forwarding**

以上输出信息表明 ping 的数据包在接口“GigabitEthernet0/0”不匹配路由策略，数据包正常转发。

21.4 路由优化命令汇总

表 21-2 列出了本章涉及到的主要的命令。

表 21-2 本章命令汇总

命令	作用
show ip policy	查看策略路由及作用的接口
show route-map	查看定义的所有路由策略及路由策略匹配的情况
debug ip policy	动态查看策略路由的匹配情况
passive-interface	配置被动接口
distribute-list	配置分布控制列表
route-map	定义路由策略
match	定义匹配的条件
set	定义对符合匹配条件的语句采取的行为
ip policy route-map	应用路由策略
ip local policy route-map	本地应用路由策略

第 22 章　IPv6

无论是 NAT，还是 CIDR 等都是缓解 IP 地址短缺的手段，而 IPv6 才是解决地址短缺的最终方法。IPv6 是由 IETF 设计的下一代互联网协议，目的是取代现有的互联网协议 IPv4。

22.1　IPv6 概述

22.1.1 IPv6 优点

IPv4 的设计思想成功地造就了目前的国际互联网，其核心价值体现在简单、灵活和开放性。但随着新应用的不断涌现，传统的 IPv4 协议已经难以支持互联网的进一步扩张和新业务的特性，比如实时应用和服务质量保证等。IPv6 能够解决 IPv4 存在的许多问题，如地址短缺、服务质量保证等。同时，IPv6 还对 IPv4 作了大量的改进，包括路由和网络自动配置等。IPv6 和 IPv4 将在过渡期内共存几年，并由 IPv6 渐渐取代 IPv4。IPv6 的特点如下：

1. 128 比特的地址方案，为将来数十年提供了足够的地址空间；
2. 充足的地址空间将极大地满足那些伴随着网络智能设备的出现而对地址增长的需求，例如个人数据助理、移动电话、家庭网络接入设备等；
3. 多等级编址层次有助于路由聚合，提高了路由选择的效率和可扩展性；
4. 自动配置使得在 Internet 上大规模布置新设备成为可能；
5. ARP 广播被本地链路多播代替；
6. IPv6 对数据包头作了简化，以减少处理器开销并节省网络带宽；
7. IPv6 中流标签字段可以提供流量区分；
8. IPv6 的组播可以区分永久性与临时性地址，更有利于组播功能的实现；
9. IPv6 地址本身的分层体系更加支持了域名解析体系中的地址集聚和地址更改；
10. IPv6 协议内置安全机制，并已经标准化；
11. IPv6 协议更好地支持移动性；
12. IPv6 提供了更加优秀的 QOS 保障；
13. IPv6 中没有广播地址，它的功能正在被组播地址所代替。

22.1.2 IPv6 地址

IPv4 地址表示为点分十进制格式，而 IPv6 采用冒号分十六进制格式。例如：

2007:00D3:0000:2F3B:02BB:00FF:FE28:2000 是一个完整的 IPv6 地址。

【提示】

1. IPv6 地址中每个 16 位分组中的前导零位可以去除做简化表示；
2. 可以将冒号十六进制格式中相邻的连续零位合并，用双冒号“::”表示；
3. 要在一个 URL 中使用文本 IPv6 地址，文本地址应该用符号“[”和“]”来封闭。

IPv6 地址有三种类型：单播、任意播和组播，在每种地址中又有一种或者多种类型的地址，如单播有本地链路地址、本地站点地址、可聚合全球地址、回环地址和未指定地址；

任意播有本地链路地址、本地站点地址和可聚合全球地址；多播有指定地址和请求节点地址。下面主要介绍几个常用地址类型：

1. 本地链路地址

当在一个节点上启用 IPv6 协议栈，启动时节点的每个接口自动配置一个本地链路地址，前缀为 FE80::/10。

2. 本地站点地址

本地站点地址与 RFC1918 所定义的私有 IPv4 地址空间类似，因此本地站点地址不能在全球 IPv6 因特网上路由，前缀为 FEC0::/10。

3. 可聚合全球单播地址

IANA 分配 IPv6 寻址空间中的一个 IPv6 地址前缀作为可聚合全球单播地址。

4. IPv4 兼容地址

IPv4 兼容的 IPv6 地址是由过渡机制使用的特殊单播 IPv6 地址，目的是在主机和路由器上自动创建 IPv4 隧道以在 IPv4 网络上传送 IPv6 数据包。

5. 回环地址

单播地址 0:0:0:0:0:0:0:1 称为回环地址。节点用它来向自身发送 IPv6 包。它不能分配给任何物理接口。

6. 不确定地址

单播地址 0:0:0:0:0:0:0:0 称为不确定地址。它不能分配给任何节点。

7. 多播指定地址

RFC2373 在多播范围内为 IPv6 协议的操作定义和保留了几个 IPv6 地址，这些保留地址称为多播指定地址。

8. 请求节点地址

对于节点或路由器的接口上配置的每个单播和任意播地址，都自动启动一个对应的被请求节点地址。被请求节点地址受限于本地链路。

22.2 IPv6 路由

22.2.1 实验 1： IPv6 静态路由

1. 实验目的

通过本实验可以掌握

（1）启用 IPv6 流量转发

（2）配置 IPv6 地址

（3）IPv6 静态路由配置和调试

（4）IPv6 默认路由配置和调试

2. 拓扑结构

实验拓扑如图 22-1 所示。

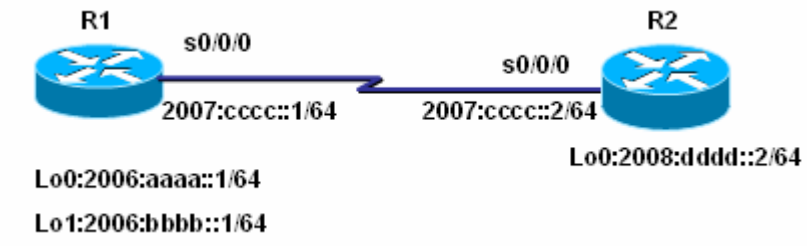

图 22-1 IPv6 静态路由

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#ipv6 unicast-routing                    //启用 IPv6 流量转发
R1(config)#interface Loopback0
R1(config-if)#ipv6 address 2006:AAAA::1/64         //配置 IPv6 地址
R1(config)#interface Loopback1
R1(config-if)#ipv6 address 2006:BBBB::1/64
R1(config)#interface Serial0/0/0
R1(config-if)#ipv6 address 2007:CCCC::1/64
R1(config-if)#no shutdown
R1(config)#ipv6 route 2008:DDDD::/64 Serial0/0/0   //配置 IPv6 静态路由
```

（2）步骤 2：配置路由器 R2

```
R2(config)#ipv6 unicast-routing
R2(config)#interface Loopback0
R2(config-if)#ipv6 address 2008:DDDD::2/64
R2(config)#interface Serial0/0/0
R2(config-if)#ipv6 address 2007:CCCC::2/64
R2(config-if)#clockrate 128000
R2(config-if)#no shutdown
R2(config)#ipv6 route ::/0 Serial0/0/0             //配置 IPv6 默认路由
```

4.实验调试

（1）show ipv6 interface

该命令用来查看 IPv6 的接口信息。

```
R1#show ipv6 interface s0/0/0
Serial0/0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::C800:BFF:FE80:0
//本接口启用 IPv6，本地链路地址自动配置
  Global unicast address(es):
    2007:CCCC::1, subnet is 2007:CCCC::/64
//全球聚合地址
  Joined group address(es):
    FF02::1
//表示本地链路上的所有节点和路由器
    FF02::2
//表示本地链路上的所有路由器
    FF02::1:FF00:1
//用于替换 ARP 机制的被请求节点的多播地址
    FF02::1:FF80:0
//与单播地址 2007:CCCC::1 相关的被请求节点多播地址
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
//启用 ICMP 重定向
```

```
  ND DAD is enabled, number of DAD attempts: 1
//邻居发现和重复地址检测启动
  ND reachable time is 30000 milliseconds
//ND 可达时间
  Hosts use stateless autoconfig for addresses.
//使用无状态自动配置地址
```

（2）**show ipv6 route**

该命令用来查看 IPv6 路由表。

```
R1#show ipv6 route
IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2006:AAAA::/64 [0/0]
     via ::, Loopback0
L   2006:AAAA::1/128 [0/0]
     via ::, Loopback0
C   2006:BBBB::/64 [0/0]
     via ::, Loopback1
L   2006:BBBB::1/128 [0/0]
     via ::, Loopback1
C   2007:CCCC::/64 [0/0]
     via ::, Serial0/0/0
L   2007:CCCC::1/128 [0/0]
     via ::, Serial0/0/0
S   2008:DDDD::/64 [1/0]
     via ::, Serial0/0/0
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
R2#show ipv6 route
IPv6 Routing Table - 7 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
S   ::/0 [1/0]
     via ::, Serial0/0/0
```

```
C    2007:CCCC::/64 [0/0]
     via ::, Serial0/0/0
L    2007:CCCC::2/128 [0/0]
     via ::, Serial0/0/0
C    2008:DDDD::/64 [0/0]
     via ::, Loopback0
L    2008:DDDD::2/128 [0/0]
     via ::, Loopback0
L    FE80::/10 [0/0]
     via ::, Null0
L    FF00::/8 [0/0]
     via ::, Null0
```

以上输出表明路由器 R1 上有一条 IPv6 的静态路由,R2 上有一条 IPv6 的默认路由,IPv6 中的默认路由是没有“*”的。

（3）ping

```
R2#ping ipv6 2006:AAAA::1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2006:AAAA::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/72/124 ms
```

22.2.2 实验 2: IPv6 RIPng

1.实验目的

通过本实验可以掌握

（1）启用 IPv6 流量转发

（2）向 RIPng 网络注入默认路由

（3）RIPng 配置和调试

2.拓扑结构

实验拓扑如图 22-2 所示。

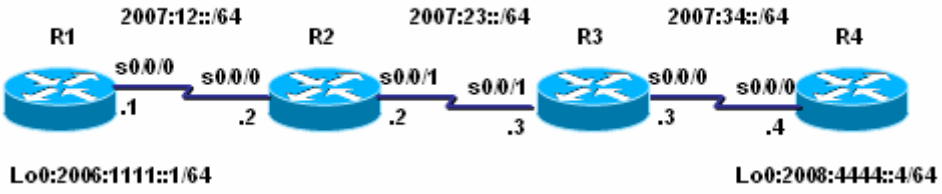

图 22-2 IPv6 RIPng 配置

3.实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#ipv6 unicast-routing
R1(config)#ipv6 router rip cisco        //启动 IPv6 RIPng 进程
R1(config-rtr)#split-horizon            //启用水平分割
R1(config-rtr)#poison-reverse           //启用毒化反转
R1(config)#interface Loopback0
```

```
R1(config-if)#ipv6 address 2006:1111::1/64
R1(config-if)#ipv6 rip cisco enable   //在接口上启用 RIPng
R1(config)#interface Serial0/0/0
R1(config-if)#ipv6 address 2007:12::1/64
R1(config-if)#ipv6 rip cisco enable
R1(config-if)#ipv6 rip cisco default-information originate
//向 IPv6 RIPng 区域注入一条默认路由（:: /0）
R1(config-if)#no shutdown
R1(config)#ipv6 route ::/0 Loopback0 //配置默认路由
```

【提示】

“ipv6 rip cisco default-information only ”命令也可以向 IPv6 RIPng 区域注入一条默认路由,但是该命令只从该接口发送默认的 IPv6 路由,而该接口其它的 IPv6 的 RIPng 路由都被抑制。

（2）步骤 2：配置路由器 R2

```
R2(config)#ipv6 unicast-routing
R2(config)#ipv6 router rip cisco
R2(config-rtr)#split-horizon
R2(config-rtr)#poison-reverse
R2(config)#interface Serial0/0/0
R2(config-if)#ipv6 address 2007:12::2/64
R2(config-if)#ipv6 rip cisco enable
R2(config-if)#clock rate 128000
R2(config-if)#no shutdown
R2(config)#interface Serial0/0/1
R2(config-if)#ipv6 address 2007:23::2/64
R2(config-if)#ipv6 rip cisco enable
R2(config-if)#clock rate 128000
R2(config-if)#no shutdown
```

（3）步骤 3：配置路由器 R3

```
R3(config)#ipv6 unicast-routing
R3(config)#ipv6 router rip cisco
R3(config-rtr)#split-horizon
R3(config-rtr)#poison-reverse
R3(config)#interface Serial0/0/0
R3(config-if)#ipv6 address 2007:34::3/64
R3(config-if)#ipv6 rip cisco enable
R3(config-if)#clockrate 128000
R3(config-if)#no shutdown
R3(config)#interface Serial0/0/1
R3(config-if)#ipv6 address 2007:23::3/64
R3(config-if)#ipv6 rip cisco enable
R3(config-if)#no shutdown
```

（4）步骤 4：配置路由器 R4

```
R4(config)#ipv6 unicast-routing
R4(config)#ipv6 router rip cisco
R4(config-rtr)#split-horizon
R4(config-rtr)#poison-reverse
R4(config)#interface Loopback0
R4(config-if)#ipv6 address 2008:4444::4/64
R4(config-if)#ipv6 rip cisco enable
R4(config)#interface Serial0/0/0
R4(config-if)#ipv6 address 2007:34::4/64
R4(config-if)#ipv6 rip cisco enable
R4(config-if)#no shutdown
```

4.实验调试

（1）show ipv6 route

```
R2#show ipv6 route
IPv6 Routing Table - 10 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
R   ::/0 [120/2]
     via FE80::C800:AFF:FE90:0, Serial0/0/0
R   2006:1111::/64 [120/2]
     via FE80::C800:AFF:FE90:0, Serial0/0/0
C   2007:12::/64 [0/0]
     via ::, Serial0/0/0
L   2007:12::2/128 [0/0]
     via ::, Serial0/0/0
C   2007:23::/64 [0/0]
     via ::, Serial0/0/1
L   2007:23::2/128 [0/0]
     via ::, Serial0/0/1
R   2007:34::/64 [120/2]
     via FE80::C802:AFF:FE90:0, Serial0/0/1
R   2008:4444::/64 [120/3]
     via FE80::C802:AFF:FE90:0, Serial0/0/1
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
```

以上输出表明 R1 确实向 IPv6 RIPng 网络注入一条 IPv6 的默认路由，同时收到 3 条 IPv6 RIPng 路由条目，而且所有 IPv6 RIPng 路由条目的下一跳地址均为邻居路由器接口的

“link-local”地址。可以通过“**show ipv6 rip next-hops**”命令查看 RIPng 的下一跳地址。

```
R2#show ipv6 rip next-hops
 RIP process "cisco", Next Hops
  FE80::C800:AFF:FE90:0/Serial0/0/0 [3 paths]
  FE80::C802:AFF:FE90:0/Serial0/0/1 [3 paths]
```

（2）show ip protocols

```
R2#show ipv6 protocols
IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "static"
IPv6 Routing Protocol is "rip cisco"
  Interfaces:
    Serial0/0/1
    Serial0/0/0
  Redistribution:
    None
```

以上输出表明启动的 IPv6 RIPng 进程为 cisco，同时在 Serial0/0/1 和 Serial0/0/0 接口上起用 RIPng。

（3）show ipv6 rip database

该命令用来查看 RIPng 的数据库。

```
R2#show ipv6 rip database
RIP process "cisco", local RIB
 2006:1111::/64, metric 2, installed
     Serial0/0/0/FE80::C800:AFF:FE90:0, expires in 178 secs
 2007:12::/64, metric 2
     Serial0/0/0/FE80::C800:AFF:FE90:0, expires in 178 secs
 2007:23::/64, metric 2
     Serial0/0/1/FE80::C802:AFF:FE90:0, expires in 168 secs
 2007:34::/64, metric 2, installed
     Serial0/0/1/FE80::C802:AFF:FE90:0, expires in 168 secs
 2008:4444::/64, metric 3, installed
     Serial0/0/1/FE80::C802:AFF:FE90:0, expires in 168 secs
 ::/0, metric 2, installed
     Serial0/0/0/FE80::C800:AFF:FE90:0, expires in 178 secs
```

以上输出显示了 R2 的 RIPng 的数据库。

（4）debug ipv6 rip

该命令用来动态查看 RIPng 的更新。

```
R2#debug ipv6 rip
RIP Routing Protocol debugging is on
R2#clear ipv6 route *
 *Feb 15 14:17:34.851: RIPng: Sending multicast update on Serial0/0/1 for cisco
 *Feb 15 14:17:34.851:        src=FE80::C801:AFF:FE90:0
 *Feb 15 14:17:34.855:        dst=FF02::9 (Serial0/0/1)
 *Feb 15 14:17:34.855:        sport=521, dport=521, length=92
```

```
*Feb 15 14:17:34.859:        command=2, version=1, mbz=0, #rte=4
*Feb 15 14:17:34.859:        tag=0, metric=2, prefix=2006:1111::/64
*Feb 15 14:17:34.859:        tag=0, metric=1, prefix=2007:12::/64
*Feb 15 14:17:34.863:        tag=0, metric=1, prefix=2007:23::/64
*Feb 15 14:17:34.863:        tag=0, metric=2, prefix=::/0
*Feb 15 14:17:34.867: RIPng: Sending multicast update on Serial0/0/0 for cisco
*Feb 15 14:17:34.867:        src=FE80::C801:AFF:FE90:0
*Feb 15 14:17:34.871:        dst=FF02::9 (Serial0/0/0)
*Feb 15 14:17:34.871:        sport=521, dport=521, length=92
*Feb 15 14:17:34.871:        command=2, version=1, mbz=0, #rte=4
*Feb 15 14:17:34.875:        tag=0, metric=1, prefix=2007:12::/64
*Feb 15 14:17:34.875:        tag=0, metric=1, prefix=2007:23::/64
*Feb 15 14:17:34.879:        tag=0, metric=2, prefix=2007:34::/64
*Feb 15 14:17:34.879:        tag=0, metric=3, prefix=2008:4444::/64
*Feb 15 14:17:43.439: RIPng: response received from FE80::C800:AFF:FE90:0 on Serial0/0/0
for cisco
*Feb 15 14:17:43.443:        src=FE80::C800:AFF:FE90:0 (Serial0/0/0)
*Feb 15 14:17:43.443:        dst=FF02::9
*Feb 15 14:17:43.447:        sport=521, dport=521, length=72
*Feb 15 14:17:43.447:        command=2, version=1, mbz=0, #rte=3
*Feb 15 14:17:43.447:        tag=0, metric=1, prefix=2006:1111::/64
*Feb 15 14:17:43.451:        tag=0, metric=1, prefix=2007:12::/64
*Feb 15 14:17:43.451:        tag=0, metric=1, prefix=::/0
R2#
*Feb 15 14:17:57.815: RIPng: response received from FE80::C802:AFF:FE90:0 on Serial0/0/1
for cisco
*Feb 15 14:17:57.819:        src=FE80::C802:AFF:FE90:0 (Serial0/0/1)
*Feb 15 14:17:57.819:        dst=FF02::9
*Feb 15 14:17:57.823:        sport=521, dport=521, length=72
*Feb 15 14:17:57.823:        command=2, version=1, mbz=0, #rte=3
*Feb 15 14:17:57.823:        tag=0, metric=1, prefix=2007:23::/64
*Feb 15 14:17:57.827:        tag=0, metric=1, prefix=2007:34::/64
*Feb 15 14:17:57.827:        tag=0, metric=2, prefix=2008:4444::/64
```

以上输出显示路由器 R2 发送和接收 RIPng 的信息。

22.2.3 实验 3：OSPFv3

1.实验目的

通过本实验可以掌握

（1）启用 IPv6 流量转发

（2）向 OSPFv3 网络注入默认路由

（3）OSPFv3 多区域配置和调试

2.拓扑结构

实验拓扑如图 22-3 所示。

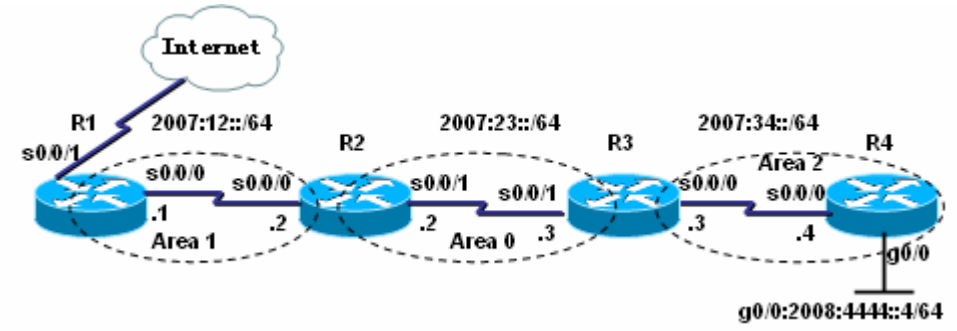

图 22-3 OSPFv3 配置

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#ipv6 unicast-routing
R1(config)#ipv6 router ospf 1          //启动 OSPFv3 路由进程
R1(config-rtr)#router-id 1.1.1.1       //定义路由器 ID
R1(config-rtr)#default-information originate metric 30 metric-type 2
//向 OSPFv3 网络注入一条默认路由
R1(config)#interface Serial0/0/0
R1(config-if)#ipv6 address 2007:12::1/64
R1(config-if)#ipv6 ospf 1 area 1    //在接口上启用 OSPFv3，并声明接口所在区域
R1(config-if)#no shutdown
R1(config)#ipv6 route ::/0 s0/0/1 //配置默认路由
```

（2）步骤 2：配置路由器 R2

```
R2(config)#ipv6 unicast-routing
R2(config)#ipv6 router ospf 1
R2(config-rtr)#router-id 2.2.2.2
R2(config)#interface Serial0/0/0
R2(config-if)#ipv6 address 2007:12::2/64
R2(config-if)#ipv6 ospf 1 area 1
R2(config-if)#clock rate 128000
R2(config-if)#no shutdown
R2(config)#interface Serial0/0/1
R2(config-if)#ipv6 address 2007:23::2/64
R2(config-if)# ipv6 ospf 1 area 0
R2(config-if)#clock rate 128000
R2(config-if)#no shutdown
```

（3）步骤 3：配置路由器 R3

```
R3(config)#ipv6 unicast-routing
R3(config)#ipv6 router ospf 1
R3(config-rtr)#router-id 3.3.3.3
R3(config)#interface Serial0/0/0
R3(config-if)#ipv6 address 2007:34::3/64
R3(config-if)#ipv6 ospf 1 area 2
R3(config-if)#clockrate 128000
```

R3(config-if)#no shutdown
R3(config)#interface Serial0/0/1
R3(config-if)#ipv6 address 2007:23::3/64
R3(config-if)#ipv6 ospf 1 area 0
R3(config-if)#no shutdown
（4）步骤 4：配置路由器 R4
R4(config)#ipv6 unicast-routing
R4(config)#ipv6 router ospf 1
R4(config-rtr)#router-id 4.4.4.4
R4(config)#interface gigabitEthernet0/0
R4(config-if)#ipv6 address 2008:4444::4/64
R4(config-if)# ipv6 ospf 1 area 2
R4(config-if)#no shutdown
R4(config)#interface Serial0/0/0
R4(config-if)#ipv6 address 2007:34::4/64
R4(config-if)#ipv6 ospf 1 area 2
R4(config-if)#no shutdown

4.实验调试

（1）show ipv6 route

```
R4#show ipv6 route
IPv6 Routing Table - 11 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
OE2  ::/0 [110/30], tag 1
     via FE80::C802:AFF:FE90:0, Serial0/0/0
OI  2007:12::/64 [110/192]
     via FE80::C802:AFF:FE90:0, Serial0/0/0
OI  2007:23::/64 [110/128]
     via FE80::C802:AFF:FE90:0, Serial0/0/0
C   2007:34::/64 [0/0]
     via ::, Serial0/0/0
L   2007:34::4/128 [0/0]
     via ::, Serial0/0/0
C   2008:4444::/64 [0/0]
     via ::, GigabitEthernet0/0
L   2008:4444::4/128 [0/0]
     via ::, GigabitEthernet0/0
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
```

via ::, Null0

以上输出表明 OSPFv3 的外部路由代码为“OE2”或“OE1”，区域间路由代码为“OI”，区域内路由代码为“O”。

（2）show ip protocols

```
R2#show ipv6 protocols
IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "static"
IPv6 Routing Protocol is "ospf 1"
  Interfaces (Area 0):
    Serial0/0/1
  Interfaces (Area 1):
    Serial0/0/0
  Redistribution:
    None
```

以上输出表明启动的 OSPFv3 进程 ID 为 1， Serial0/0/1 和 Serial0/0/0 接口上启用 OSPFv3，Serial0/0/1 属于区域 0，Serial0/0/0 属于区域 1。

（3）show ipv6 ospf database

该命令用来查看 OSPFv3 拓扑结构数据库。

```
R4#show ipv6 ospf database

            OSPFv3 Router with ID (4.4.4.4) (Process ID 1)

                Router Link States (Area 2)

ADV Router      Age         Seq#       Fragment ID  Link count  Bits
3.3.3.3         418         0x80000002 0            1           B
4.4.4.4         375         0x80000004 0            1           None

                Inter Area Prefix Link States (Area 2)

ADV Router      Age         Seq#       Prefix
3.3.3.3         450         0x80000001 2007:23::/64
3.3.3.3         440         0x80000001 2007:12::/64

                Inter Area Router Link States (Area 2)

ADV Router      Age         Seq#       Link ID    Dest RtrID
3.3.3.3         440         0x80000001 16843009   1.1.1.1

                Link (Type-8) Link States (Area 2)

ADV Router      Age         Seq#       Link ID    Interface
4.4.4.4         415         0x80000001 3          Fa0/0
3.3.3.3         463         0x80000001 4          Se1/0
```

```
4.4.4.4         437         0x80000001  4          Se1/0

                Intra Area Prefix Link States (Area 2)

ADV Router      Age         Seq#        Link ID    Ref-lstype  Ref-LSID
3.3.3.3         463         0x80000001  0          0x2001      0
4.4.4.4         430         0x80000002  0          0x2001      0

                Type-5 AS External Link States

ADV Router      Age         Seq#        Prefix
1.1.1.1         6           0x80000013  ::/0
```

以上输出显示了路由器 R4 的 OSPFv3 的拓扑结构数据库。

（4）**show ipv6 ospf neighbor**

```
R2#show ipv6 ospf neighbor

Neighbor ID     Pri   State           Dead Time   Interface ID   Interface
3.3.3.3           1   FULL/  -        00:00:30    5              Serial0/0/1
1.1.1.1           1   FULL/  -        00:00:37    4              Serial0/0/0
```

以上输出表明路由器 R2 有两个 OSPFv3 的邻居。

（5）**show ipv6 ospf interface**

```
R2#show ipv6 ospf interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Link Local Address FE80::C801:AFF:FE90:0, Interface ID 4
  Area 1, Process ID 1, Instance ID 0, Router ID 2.2.2.2
  Network Type POINT_TO_POINT, Cost: 64
  Transmit Delay is 1 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:04
  Index 1/1/1, flood queue length 0
  Next 0x0(0)/0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 1.1.1.1
  Suppress hello for 0 neighbor(s)
```

以上输出是 OSPFv3 路由器接口的基本信息，和 OSPFv2 非常的相似，包括路由器 ID、网络类型、计时器的值以及邻居的数量等信息。

22.2.4 实验 2: IPv6 EIGRP

1.实验目的

通过本实验可以掌握

（1）启用 IPv6 流量转发

（2）IPv6 EIGRP 配置和调试

2. 拓扑结构

实验拓扑如图 22-4 所示。

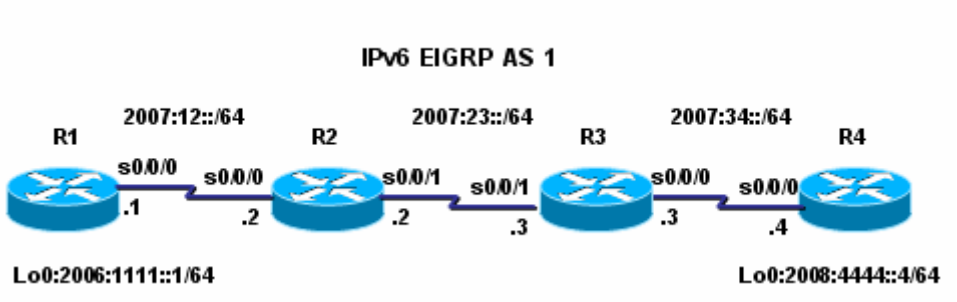

图 22-4 IPv6 EIGRP 配置

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#ipv6 unicast-routing
R1(config)#ipv6 router eigrp 1          //配置 IPv6 EIGRP
R1(config-rtr)#router-id 1.1.1.1        //配置路由器 ID
R1(config-rtr)#no shutdown              //启动 IPv6 EIGRP 进程
R1(config-rtr)#redistribute connected metric 10000 100 255 1 1500
//将直连重分布到 IPv6 EIGRP 中
R1(config)#interface Loopback0
R1(config-if)#ipv6 address 2006:1111::1/64
R1(config)#interface Serial0/0/0
R1(config-if)#ipv6 address 2007:12::1/64
R1(config-if)#ipv6 eigrp 1                          //在接口上启用 IPv6 EIGRP
R1(config-if)#no shutdown
```

（2）步骤 2：配置路由器 R2

```
R2(config)#ipv6 unicast-routing
R2(config)#ipv6 router eigrp 1
R2(config-rtr)#router-id 2.2.2.2
R2(config-rtr)#no shutdown
R2(config)#interface Serial0/0/0
R2(config-if)#ipv6 address 2007:12::2/64
R2(config-if)#clock rate 128000
R2(config-if)#ipv6 eigrp 1
R2(config-if)#no shutdown
R2(config)#interface Serial0/0/1
R2(config-if)#ipv6 address 2007:23::2/64
R2(config-if)#clock rate 128000
R2(config-if)#ipv6 eigrp 1
R2(config-if)#no shutdown
```

（3）步骤 3：配置路由器 R3

```
R3(config)#ipv6 unicast-routing
R3(config)#ipv6 router eigrp 1
```

```
R3(config-rtr)# router-id 3.3.3.3
R3(config-rtr)#no shutdown
R3(config)#interface Serial0/0/0
R3(config-if)#ipv6 address 2007:34::3/64
R3(config-if)#clockrate 128000
R3(config-if)#ipv6 eigrp 1
R3(config-if)#no shutdown
R3(config)#interface Serial0/0/1
R3(config-if)#ipv6 address 2007:23::3/64
R3(config-if)#ipv6 eigrp 1
R3(config-if)#no shutdown
```

（4）步骤 4：配置路由器 R4

```
R4(config)#ipv6 unicast-routing
R4(config)#ipv6 router eigrp 1
R4(config-rtr)# router-id 4.4.4.4
R4(config-rtr)#no shutdown
R4(config)#interface Loopback0
R4(config-if)#ipv6 address 2008:4444::4/64
R4(config-if)#ipv6 eigrp 1
R4(config)#interface Serial0/0/0
R4(config-if)#ipv6 address 2007:34::4/64
R4(config-if)#ipv6 eigrp 1
R4(config-if)#no shutdown
```

4.实验调试

（1）show ipv6 route eigrp

该命令用来查看 IPv6 EIGRP 的路由。

```
R1#show ipv6 route eigrp
IPv6 Routing Table - 8 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
D   2007:23::/64 [90/21024000]
     via FE80::219:55FF:FE66:6320, Serial0/0/0
D   2007:34::/64 [90/21536000]
     via FE80::219:55FF:FE66:6320, Serial0/0/0
D   2008:4444::/64 [90/21664000]
     via FE80::219:55FF:FE66:6320, Serial0/0/0
R2#show ipv6 route eigrp
IPv6 Routing Table - 8 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
```

```
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
EX  2006:1111::/64 [170/20537600]
     via FE80::219:55FF:FE35:B828, Serial0/0/0
D   2007:34::/64 [90/21024000]
     via FE80::219:55FF:FE35:B548, Serial0/0/1
D   2008:4444::/64 [90/21152000]
     via FE80::219:55FF:FE35:B548, Serial0/0/1
```

以上输出说明路由表中的下一跳是对方的本地链路地址，同时 IPv6 EIGRP 也能够区分内部路由和外部路由，外部路由代码为“EX”。

（2）show ipv6 eigrp neighbors

该命令用来查看 IPv6 EIGRP 的邻居。

```
R2#show ipv6 eigrp neighbors
IPv6-EIGRP neighbors for process 1
H  Address                 Interface  Hold Uptime   SRTT  RTO  Q  Seq
                                      (sec)         (ms)       Cnt Num
1  Link-local address:     Se0/0/1     14  00:33:32   13  1140  0  24
   FE80::219:55FF:FE35:B548
0  Link-local address:     Se0/0/0     14  00:33:32   10  1140  0   9
   FE80::219:55FF:FE35:B828
```

以上输出表明路由器 R2 有两个 IPv6 EIGRP 邻居，邻居的地址用对方的本地链路地址表示。

（3）show ipv6 eigrp topology

该命令用来查看 IPv6 EIGRP 的拓扑结构信息。

```
R2#show ipv6 eigrp topology
IPv6-EIGRP Topology Table for AS(1)/ID(2.2.2.2)

Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status

P 2007:12::/64, 1 successors, FD is 20512000
        via Connected, Serial0/0/0
P 2006:1111::/64, 1 successors, FD is 20537600
        via FE80::219:55FF:FE35:B828 (20537600/281600), Serial0/0/0
P 2007:23::/64, 1 successors, FD is 20512000
        via Connected, Serial0/0/1
P 2007:34::/64, 1 successors, FD is 21024000
        via FE80::219:55FF:FE35:B548 (21024000/20512000), Serial0/0/1
P 2008:4444::/64, 1 successors, FD is 21152000
        via FE80::219:55FF:FE35:B548 (21152000/20640000), Serial0/0/1
```

（4）show ipv6 protocols

```
R2#show ipv6 protocols
```

```
IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "static"
IPv6 Routing Protocol is "eigrp 1"
// IPv6 EIGRP 进程
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
//计算度量之的因子
  EIGRP maximum hopcount 100
//最大跳数
  EIGRP maximum metric variance 1
// variance 值为 1，表示默认只支持等价路径负载均衡
  Interfaces:
    Serial0/0/0
    Serial0/0/1
//以上三行表示启用 IPv6 EIGRP 的接口
  Redistribution:
    None
  Maximum path: 16
//默认最大等价路径为 16 条，最多可以配置 64 条
  Distance: internal 90 external 170
//IPv6 EIGRP 的内部路由管理距离为 90，外部路由管理距离为 170
```

22.3 IPv6 命令汇总

表 22-1 列出了本章涉及到的主要的命令。

表 22-1 本章命令汇总

命令	作用
show ipv6 route	查看 IPv6 路由表
show ipv6 interface	查看 IPv6 接口信息
show ipv6 protocols	查看和 IPv6 路由协议相关的信息
show ipv6 rip next-hops debug ipv6 rip	查看 RIPng 的下一跳地址
show ipv6 rip database	查看 RIPng 的数据库
show ipv6 ospf neighbor	查看 OSPFv3 邻居的基本信息
show ipv6 ospf interface	查看 OSPFv3 路由器接口的信息
show ipv6 ospf database	查看 OSPFv3 拓扑结构数据库
show ipv6 ospf	查看 OSPFv3 进程及其细节
show ipv6 route eigrp	查看 IPv6 EIGRP 的路由
show ipv6 eigrp topology	查看 IPv6 EIGRP 的拓扑结构信息
show ipv6 eigrp neighbors	查看 IPv6 EIGRP 的邻居
debug ipv6 rip	动态查看 RIPng 的更新
ipv6 unicast-routing	启动 IPv6 流量转发
ipv6 address	在接口下配置 IPv6 地址
ipv6 route	配置 IPv6 静态路由
ipv6 router rip	启动 IPv6 RIPng 进程
split-horizon	启用水平分割

poison-reverse	启用毒化反转
ipv6 rip *tag* enable	在接口上启用 RIPng
ipv6 rip *tag* default-information originate	向 IPv6 RIPng 区域注入一条默认路由
ipv6 router ospf	启动 OSPFv3 路由进程
router-id	配置路由器 ID
default-information originate	向 OSPFv3 网络注入一条默认路由
ipv6 ospf *process-id* area *area-id*	接口上启用 OSPFv3，并声明接口所在区域
ipv6 router eigrp	配置 IPv6 EIGRP 路由协议
ipv6 eigrp	接口下启用 IPv6 EIGRP
maximum-paths	配置能支持的等价路径的条数
variance	配置 IPv6 EIGRP 非等价负载均衡

第 23 章 组播

随着网络中的视频等应用越来越多，组播也越来越时髦。路由器转发组播流的方式和转发单播流有很大的差别，发送数据的源不知道接收者在何处。组播也有很多路由协议，它们为组播流建立一棵无环路、无重复流量的转发树。本章将介绍常用的、简单的两个协议：PIM dense 和 PIM sparse。

23.1 组播简介

当有 10000 个用户通过网络看电视时，如果分别为每个用户传输一路流量，不仅服务器受不了，网络也承载不了。组播（也称多播）则像电视一样，传输一份数据，需要接收数据的计算机加入到这个组就行了。虽然组播配置并不复杂，但理论知识则相当复杂。组播采用 224.0.0.0--239.255.255 的地址，不同地址就是不同的组，一个组可能有多个源，而需要接收数据的设备是这个组的成员。

路由器转发组播流的方式和转发单播流有很大的差别，发送数据的组播源不知道接收者在何处。保证接收者能收到数据，并且数据不会在不必要的网络上存在是很重要的事情。路由器必须为组播确定出一条转发路径。路由器采用反向路径转发（RPF），即对每个接收到的组播进行源地址测试，如果数据是从到达源的接口上接收到的，就往下游路由器转发。

为了确定是否应该往某个网络转发组播流，路由器使用 IGMP（Internet Group Manage Protocol）和主机之间通信，确定这个网络是否有某个组的成员。IGMP 有 V1、V2、V3，目前 IOS 采用 V2。IGMP 有各种消息，例如：Membership Query 消息、Membership Report 消息、Leave Group 消息、General Query 消息、Group-Specific Query 消息等。

PIM（Protocol Independent Multicast）是一个组播路由协议，独立于协议的意思是该组播协议不关心单播路由是通过 RIP 还是 OSPF 或者其他方式学习到的。PIM 有两种模式：PIM Dense 和 PIM Sparse，后者通常和 Dense 结合使用，成为 PIM Sparse-Dense。

Dense 通常用于组成员比较密集的网络中。在 Dense 模式中，当有组播源出现时，路由器假设所有的网络都有组成员，构建了一棵从源开始的转发树，全部网络就都有了组播流量。然而各个路由器会紧接着查询自己的接口上是否有这个组的成员存在，如果没有成员，将停止往这个接口转发组播流。如果路由器上一个成员都没有，它将向上游路由器发送消息，把它从转发树上修剪掉。一级一级地往上发送消息，最终多播路由协议将构建一棵以源为根、不会有多余组播流量存在的转发树。如果有新的成员加入，路由器将一级一级往上发送消息，建立转发路径。在 Dense 模式中，会为不同的源建立不同的树，这样树的数量可能会很多。

Sparse 则通常用于组成员比较稀疏的网络中。在 Sparse 模式中，路由器假设所有的网络都没有组成员，除非有主机明确表示加入该组。转发树的建立从终端的叶节点组成员开始，向后扩展到中心的根节点上。和 Dense 模式不同，Sparse 是基于共享树的。也就是说某个组的流量是先发送到中心节点上（称为 RP），然后在从 RP 转发到各个组成员上的。组成员加入到这个组时，本地路由器向 RP 发送成员报告，沿途的路由器将树枝加入到共享树中。组成员从组中退出时，才执行修剪。这样好处是树的数量少，然而可能造成一些组播数据绕了一圈才到达主机。因此，实际上默认时，当路由器发现不是从到达源的最佳路径的接口上收到组播流，会自动切换到基于源的树。Sparse 模式中，我们可以手工为某个组指定 RP，也可以让路由器自动选举。要注意的是，路由器自动选举 RP 时发送的是组播流量，由于 RP 还没出现，所以只能使用 Dense 模式传输这些组播流量，所以 Sparse 通常和 Dense 结合使用。

而对于交换机，也不能说从一个接口收到组播，就防洪到全部接口。然而交换机并没有IGMP 协议和主机通信，交换机采用 2 种方案，一种是 IGMP Snopping，另一种是 CGMP。IGMP Snopping 中，交换机监听主机和路由器之间的 IGMP 消息，从而确定出哪个接口上有什么组的成员存在，组播流则从这些特定的接口发送出去。在 2 层交换机或者低端的三层交换机上，IGMP Snopping 基本是默认采用的。而 CGMP 协议则是交换机用来和路由器进行通信，从路由器获得组的成员名单，从而确定哪些接口应该转发哪些组的流量。

23.2 实验 1：PIM Dense

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 PIM Dense 的工作原理

（2） 掌握 PIM Dense 的配置

2. 实验拓扑

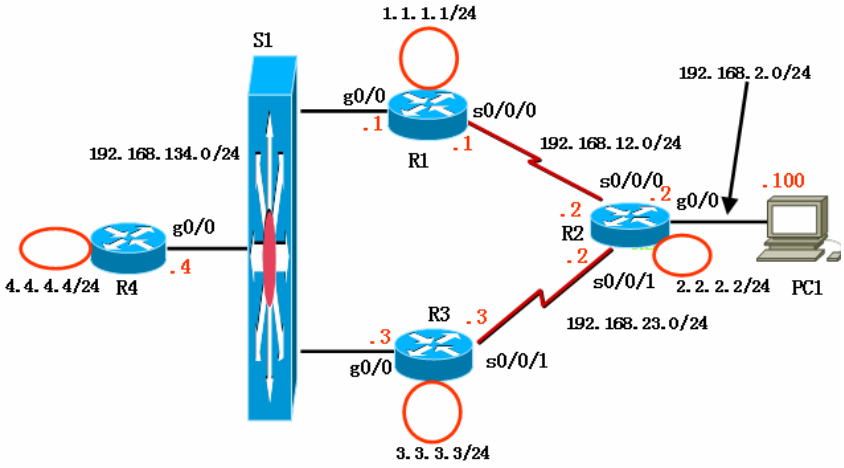

图 23-1 实验 1、实验 2 拓扑图

拓扑中 R2 的 g0/0 接口以及 PC1 也是接在交换机 S1 上的，我们把这两个接口单独划分在一个 VLAN 中。图 23-1 没有画出它们和 S1 的连接情况。

3. 实验步骤

（1） 步骤 1：配置 IP 地址、把 R2 的 g0/0 和 PC1 所接的交换机接口单独划在一个 VLAN

各路由器的 IP 如图 23-1 所示，在每个路由器上创建环回口 Loopback0，配置 IP 地址。把 R2 的 g0/0 所接的 f0/2 和 PC1 所接的 f0/5 接口单独划分一个 VLAN，如下：

```
S1(config)#vlan 2
S1(config)#int f0/2
S1(config-if)#switch mode access
S1(config-if)#switch access vlan 2
S1(config)#int f0/5
S1(config-if)#switch mode access
S1(config-if)#switch access vlan 2
```

（2） 步骤 2：配置路由协议

```
R1(config)#router rip
R1(config-router)#network 1.0.0.0
R1(config-router)#network 192.168.12.0
R1(config-router)#network 192.168.134.0

R2(config)#router rip
R2(config-router)#network 192.168.12.0
R2(config-router)#network 192.168.23.0
R2(config-router)#network 192.168.2.0
R2(config-router)#network 2.0.0.0

R3(config)#router rip
R3(config-router)#network 192.168.23.0
R3(config-router)#network 192.168.134.0
R3(config-router)#network 3.0.0.0

R4(config)#router rip
R4(config-router)#network 192.168.134.0
R4(config-router)#network 4.0.0.0
```

（3） 步骤 3：配置 PIM Dense

```
R1(config)#ip multicast-routing
//以上是启用组播路由功能
R1(config)#int loopback0
R1(config-if)#ip pim dense
//组播的配置相当简单，在接口上运行 pim dense 协议即可
R1(config)#int s0/0/0
R1(config-if)#ip pim dense
R1(config)#int g0/0
R1(config-if)#ip pim dense

R2(config)#ip multicast-routing
R2(config)#int loopback0
R2(config-if)#ip pim dense
R2(config)#int s0/0/0
R2(config-if)#ip pim dense
R2(config)#int s0/0/1
R2(config-if)#ip pim dense
R2(config)#int g0/0
R2(config-if)#ip pim dense

R3(config)#ip multicast-routing
R3(config)#int loopback0
R3(config-if)#ip pim dense
```

R3(config)#int s0/0/1
R3(config-if)#ip pim dense
R3(config)#int g0/0
R3(config-if)#ip pim dense

R4(config)#ip multicast-routing
R4(config)#int loopback0
R4(config-if)#ip pim dense
R4(config-if)#ip igmp join-group 237.0.0.1
//该接口加入到 237.0.0.1 组中，我们要利用 237.0.0.1 组做测试
R4(config)#int g0/0
R4(config-if)#ip pim dense

（4） 步骤 4：检查 pim 邻居、测试组播路由

R1#show ip pim neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      S - State Refresh Capable

Neighbor Address	Interface	Uptime/Expires	Ver	DR Prio/Mode
192.168.12.2	Serial0/0/0	06:57:59/00:01:27	v2	1 / S
192.168.134.3	GigabitEthernet0/0	06:25:46/00:01:29	v2	1 / S
192.168.134.4	GigabitEthernet0/0	06:25:46/00:01:24	v2	1 / DR S

//以上显示了 R1 上 pim 邻居

在 PC1 上配置 IP 地址，执行“ping -t 237.0.0.1”，这时 PC1 实际上就是组 237.0.0.1 的组播源了。在各个路由器上检查多播路由表：

R2#show ip mroute
（此处省略）
(*, 237.0.0.1), 00:03:07/stopped, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0/0/1, Forward/Dense, 00:03:07/00:00:00
    Serial0/0/0, Forward/Dense, 00:03:07/00:00:00

(192.168.2.100, 237.0.0.1), 00:03:07/00:02:52, flags: T
  Incoming interface: GigabitEthernet0/0, RPF nbr 0.0.0.0
//以上表明 R2 是从 g0/0 接口接收到多播流的
  Outgoing interface list:
    Serial0/0/1, Forward/Dense, 00:03:08/00:00:00
    Serial0/0/0, Prune/Dense, 00:01:09/00:01:50
//可以看到 R2 上只往 s0/0/1 接口转发多播流量；s0/0/0 接口不转发，处于被修剪状态。

(*, 224.0.1.40), 06:49:33/stopped, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0

```
  Outgoing interface list:
    Serial0/0/1, Forward/Dense, 06:49:33/00:00:00
    Serial0/0/0, Forward/Dense, 06:49:33/00:00:00
```

【技术要点】需要仔细观察组播树的情况。在各个路由器上使用“**show ip mroute**”命令，应该可以看到：组播数据从 R2 的 g0/0 接口接收到，从 s0/0/1 接口转发出去到达 R3 的 s0/0/1，s0/0/0 接口则不转发；R3 从 s0/0/1 接口收到组播，从 g0/0 接口转发出去到达 R4 的 g0/0；R4 从 g0/0 接口收到组播，从 loopback0 接口转发出去。

把 R1 上的 loopback0 接口也加入到组 237.0.0.1 中，重新在 R2 上检查多播路由表。如下：

```
R1(config)#int loopback0
R1(config-if)#ip igmp join-group 237.0.0.1

R2#show ip mroute
（此处省略）
(192.168.2.100, 237.0.0.1), 00:10:21/00:02:58, flags: T
  Incoming interface: GigabitEthernet0/0, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0/0/1, Forward/Dense, 00:10:22/00:00:00
    Serial0/0/0, Forward/Dense, 00:00:03/00:00:00
```

//可以看到R2上的s0/0/0接口开始转发数据了,原因在于R1上有组成员了。可见PIM Dense会根据组成员的加入或者退出，动态地维护转发树。

```
（此处省略）

R1#show ip mroute
（此处省略）
(192.168.2.100, 237.0.0.1), 00:10:44/00:02:59, flags: LT
  Incoming interface: Serial0/0/0, RPF nbr 192.168.12.2
  Outgoing interface list:
    Loopback0, Forward/Dense, 00:02:25/00:00:00
    GigabitEthernet0/0, Prune/Dense, 00:01:43/00:01:16
```

//可以看到R1上的g0/0处于修剪状态,因为R3已经往192.168.134.0/24网段转发数据了，这样该网段才不会有2份组播流量。

```
（此处省略）

R3#show ip mroute
（此处省略）
(192.168.2.100, 237.0.0.1), 06:42:14/00:02:58, flags: T
  Incoming interface: Serial0/0/1, RPF nbr 192.168.23.2
  Outgoing interface list:
  GigabitEthernet0/0, Forward/Dense, 00:13:21/00:00:00, A
```

//可以是看到R3在往192.168.134.0/24网段转发数据。

```
（此处省略）
```

在 PC1 上停止 ping，在各个路由器上查看多播路由表，各个路由器的多播路由表在大约 3 分钟后会消失。这是因为没有了源，路由器无需再维护这些路由表了。

（5） 步骤 5：查看 IGMP 组成员

```
R1#show ip igmp groups
IGMP Connected Group Membership
Group Address    Interface                Uptime    Expires   Last Reporter   Group Accounted
237.0.0.1        Loopback0                00:15:14  00:02:48  1.1.1.1
//这是我们用命令加入的成员
224.0.1.40       GigabitEthernet0/0       00:28:07  00:02:50  192.168.134.4
224.0.1.40       Serial0/0/0              07:30:07  stopped   192.168.12.1
```

//以上是查看 R1 上各个接口上各组有什么成员存在。

（6） 步骤 6：配置、检查 IGMP Snooping

IGMP Snooping 是在交换机上工作的。默认时就已经启用，也可以用以下命令打开：

```
S1(config)#ip igmp snooping
S1#show ip igmp snooping
Global IGMP Snooping configuration:
-----------------------------------
IGMP snooping              : Enabled
IGMPv3 snooping (minimal)  : Enabled
Report suppression         : Enabled
TCN solicit query          : Disabled
TCN flood query count      : 2
Last Member Query Interval : 1000

Vlan 1:
--------
IGMP snooping                       : Enabled
IGMPv2 immediate leave              : Disabled
Explicit host tracking              : Enabled
Multicast router learning mode      : pim-dvmrp
Last Member Query Interval          : 1000
CGMP interoperability mode          : IGMP_ONLY
```

//以上是显示交换机上的 IGMP Snooping 运行情况

```
S1#show ip igmp snooping groups
Vlan      Group          Type        Version     Port List
------------------------------------------------------------
1         224.0.1.40     igmp        v2          Fa0/1, Fa0/3, Fa0/4
```

// f0/1、f0/3、f0/4 接口上有 224.0.1.40 组的成员存在，则应该往这些接口转发 224.0.1.40 的多播流，其他接口不转发，节约了带宽。

23.2 实验 2: PIM Sparse-Dense

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 PIM Sparse-Dense 的工作原理

（2） 掌握 PIM Sparse-Dense 的配置

2. 实验拓扑

如图 23-1。

3. 实验步骤

在实验 1 基础上继续本实验。

（1） 步骤 1：配置 PIM Sparse，采用静态 RP

```
R1(config)#int loopback0
R1(config-if)#ip pim sparse-dense
//在接口上运行 pim sparse-dense 协议即可
R1(config)#int s0/0/0
R1(config-if)#ip pim sparse-dense
R1(config)#int g0/0
R1(config-if)#ip pim sparse-dense
R1(config)#ip pim rp-address 3.3.3.3
//以上静态配置 R3 为 RP。
R1(config)#ip pim spt-threshold infinity
//以上是防止从基于 RP 的树切换到基于源的树，默认是切换的，稍后介绍树的切换问题。

R2(config)#int loopback0
R2(config-if)#ip pim sparse-dense
R2(config)#int s0/0/0
R2(config-if)#ip pim sparse-dense
R2(config)#int s0/0/1
R2(config-if)#ip pim sparse-dense
R2(config)#int g0/0
R2(config-if)#ip pim sparse-dense
R2(config)#ip pim rp-address 3.3.3.3

R3(config)#int loopback0
R3(config-if)#ip pim sparse-dense
R3(config)#int s0/0/1
R3(config-if)#ip pim sparse-dense
R3(config)#int g0/0
R3(config-if)#ip pim sparse-dense
R3(config)#ip pim rp-address 3.3.3.3

R4(config)#int loopback0
R4(config-if)#ip pim sparse-dense
```

R4(config)#int g0/0
R4(config-if)#ip pim sparse-dense
R4(config)#ip pim rp-address 3.3.3.3

（2） 步骤 2：测试组播路由

在 PC1 上执行“ping -t 237.0.0.1”，在各个路由器上检查多播路由表：

R1#show ip mroute

（此处省略）

(*, 237.0.0.1), 07:31:41/00:02:59, RP 3.3.3.3, flags: SCL

Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.134.3

//可以看到 R1 是从 g0/0 接口接收到组播流的，原因在于 RP 是 R3，而从 R1 到达 R3，从 g0/0 是最近的。这时，组播流实际上是从 PC1 到 R2，再到 R3（RP）上，然后从 R3 的 g0/0 再到 R1 的 g0/0 接口，很显然绕了一圈，这是不合理的。

【技术要点】在基于 RP 的树中，组播数据是先从源到达 RP 的，然后从 RP 到达各个组成员。从 RP 到达各个组成员的数据转发是根据共享树的构建情况来转发;而从组播源到达 RP 的组播流的转发实际上也需要一棵树，这时采用是基于源的树。

（3） 步骤 3：从基于 RP 的树切换到基于源的树

R1(config)#no ip pim spt-threshold infinity

//以上是允许从基于 RP 的树切换到基于源的树，实际上这是默认值，在步骤 1 中被我们修改了。

R1#show ip mroute

（此处省略）

(*, 237.0.0.1), 07:36:48/00:02:58, RP 3.3.3.3, flags: SJCL

Incoming interface: GigabitEthernet0/0, RPF nbr 192.168.134.3

Outgoing interface list:

Loopback0, Forward/Sparse-Dense, 00:57:35/00:02:26

(192.168.2.100, 237.0.0.1), 00:00:01/00:02:59, flags: LJT

Incoming interface: Serial0/0/0, RPF nbr 192.168.12.2

Outgoing interface list:

Loopback0, Forward/Sparse-Dense, 00:00:01/00:02:58

//可以看到这时多了一条多播路由，这是基于源的路由，这时 R1 实际上是从 s0/0/0 接口接收到数据，显然这才是合理的。

【技术要点】树的切换是很重要、很复杂的事情。树的切换是避免组播流绕了一圈才到达接受者。默认时是会切换的，因此在 Sparse 模式中，也是存在基于源的树的。

（4） 步骤 4：配置 Auto RP

在各个路由器上，去掉配置静态 RP 的命令，如下：

R1(config)#no p pim rp-address 3.3.3.3
R2(config)#no p pim rp-address 3.3.3.3
R3(config)#no p pim rp-address 3.3.3.3
R4(config)#no p pim rp-address 3.3.3.3

R2(config)# ip pim send-rp-discovery loopback 0 scope 255

//把 R2 配置为映射代理，Loopback0 为代理地址。

R3(config)# **ip pim send-rp-announce Loopback0 scope 255**

//把 R3 配置为候选 RP，Loopback0 为 RP 地址。

在各个路由器上检查 RP 的地址：

R1#**show ip pim rp mapping**

PIM Group-to-RP Mappings

Group(s) 224.0.0.0/4

RP 3.3.3.3 (?), v2v1

Info source: 2.2.2.2 (?), elected via Auto-RP

Uptime: 00:00:36, expires: 00:02:19

//在各个路由器上都可以看到，所有的组播地址的 RP 都已经映射到 R3 了。

再按照步骤 2 测试组播路由。

23.4 本章小结

本章介绍了组播的目的和基本工作原理，组播的原理甚是复杂。组播路由协议目的是建立一棵合理的转发树，并能动态地根据组成员的加入、退出调整转发树。本章介绍了 PIM Dense 和 PIM Sparse 这两个组播路由协议的配置。在 Dense 中，是先向所有网络转发数据，然后修剪不必要的树枝；而 Sparse 中，组播流先从源发到 RP 上，当有组成员加入时，才沿途打开接口转发组播流。在交换机上是采用 IGMP Snopping 和 CGMP 来决定应该往什么接口转发什么样的组流量。表 23-1 是本章出现的命令。

表 23-1 本章命令汇总

命令	作用
ip multicast-routing	启用组播路由功能
ip pim dense	在接口上运行 pim dense 协议
ip igmp join-group 237.0.0.1	该接口加入到 237.0.0.1 组中
show ip pim neighbor	显示了 pim 邻居
show ip mroute	显示组播路由表
show ip igmp groups	显示组成员
ip igmp snooping	在交换机上配置 IGMP Snopping 功能
show ip igmp snooping	显示交换机上的 IGMP Snooping 运行情况
show ip igmp snooping groups	交换机上显示各组的成员在什么接口上
ip pim sparse-dense	在接口上运行 pim sparse-dense 协议
ip pim spt-threshold infinity	静止从基于 RP 的树切换到基于源的树
ip pim send-rp-discovery loopback 0 scope 255	把路由器设为映射代理，其 Loopback0 为代理地址
ip pim send-rp-announce Loopback0 scope 255	把路由器设为候选 RP，其 Loopback0 为 RP 地址

第 24 章　BGP

通常可以将路由协议分为 IGP（内部网关协议）和 EGP（外部网关协议）。EGP 主要用于 ISP 之间交换路由信息。目前使用最为广泛的 EGP 是 BGP 版本 4，它是第一个支持 CIDR 和路由汇总的 BGP 版本。RFC1772 对 BGP 有详细的定义。

24.1　BGP 概述

24.1.1 BGP 特征

BGP 被称为是路径向量路由协议,它的任务是在自治系统之间交换路由信息，同时确保没有路由环路，其特征如下：

1. 用属性(attribute)描述路径，而不是用度量值；
2. 使用 TCP（端口 179）作为传输协议，继承了 TCP 的可靠性和面向连接的特性；
3. 通过 keepalive 信息来检验 TCP 的连接；
4. 具有丰富的属性特征，方便实现基于策略的路由；
5. 拥有自己的 BGP 表；
6. 支持 VLSM 和 CIDR；
7. 适合在大型网路中使用。

在详细讨论 BGP 之前，首先应该掌握如下 BGP 术语：

1. 对等体（peer）：当两台 BGP 路由器之间建立了一条基于 TCP 的连接后，就称它们为邻居或对等体；

2. AS：是一组处于统一管理控制和策略下的路由器或主机。AS 号由因特网注册机构分配,范围为 1-65535，其中 64512-65535 是私有使用的；

3. IBGP：当 BGP 在一个 AS 内运行时，被称为内部 BGP（IBGP）；

4. EBGP：当 BGP 运行在 AS 之间时，被称为外部 BGP（EBGP）；

5. NLRI（网络层可达性信息）：BGP 通过 NLRI 支持 CIDR 的。NLRI 是 BGP 更新报文的一部分，用于列出可到达的目的地的集合；

6. 同步：在 BGP 能够通告路由之前，该路由必须存在于当前的 IP 路由表中。也就是说，BGP 和 IGP 必须在网络能被通告前同步。Cisco 允许通过命令“no synchronization”来关闭同步；

7. IBGP 水平分割:通过 IBGP 学到的路由不能通告给其它的 IBGP 邻居。

24.1.2 BGP 属性

BGP 具有丰富的属性，为网络管理员进行路由控制带来很大的方便，BGP 路径属性分为 4 类：

1. **公认必遵（Well-Known Mandatory）**：BGP 更新报文中必须包含的，且必须被所有 BGP 厂商实现所能识别的,包括 ORIGIN，AS_PATH 和 Next_Hop 三个属性。

（1）**ORIGIN（起源）**：这个属性说明了源路由是怎样放到 BGP 表中的。有三个可能的源：IGP，EGP 以及 INCOMPLETE。路由器在多个路由选择的处理中使用这个信息。路由器选择具有最低 ORIGIN 类型的路径。ORIGIN 类型从低到高的顺序为：IGP<EGP<INCOMPLETE；

（2）**AS_PATH（AS 路径）**：指出包含在 UPDATE 报文中的路由信息所经过的自治系统的序列；

（3）**Next_HOP（下一跳）**：声明路由器所获得的 BGP 路由的下一跳。对 EBGP 会话来说，

下一跳就是通告该路由的邻居路由器的源地址。对于 IBGP 会话，有两种情况，一是起源 AS 内部的路由的下一跳就是通告该路由的邻居路由器的源地址；二是由 EBGP 注入 AS 的路由，它的下一跳会不变的带入 IBGP 中；

2. **公认自决（Well-Known Discretionary）**：指必须被所有 BGP 实现所识别，但是在 BGP 更新报文中可以发送，也可以不发送的属性，包括 LOCAL_PREF 和 ATOMIC_ AGGREGATE 两个属性。

（1）**LOCAL_PREF（本地优先级）**：本地优先级属性是用于告诉自治系统内的路由器在有多条路径的时候，怎样离开自治系统。本地优先级越高，路由优先级越高。这个属性仅仅在 IBGP 邻居之间传递；

（2）**ATOMIC_ AGGREGATE（原子聚合）**：原子聚合属性指出已被丢失了的信息。当路由聚合时将会导致信息的丢失，因为聚合来自具有不同属性的不同源。如果一个路由器发送了导致信息丢失的聚合，路由器被要求将原子聚合属性附加到该路由上。

3. **可选过渡（Optional Transitive）**：可选过渡属性并不要求所有的 BGP 实现都支持。如果该属性不能被 BGP 进程识别，它就会去看过渡标志。如果过渡标志被设置了， BGP 进程会接受这个属性并将它不加改变的传送，包括 AGGREGATOR 和 COMMUNITY。

（1）**AGGREGATOR（聚合者）**：此属性标明了实施路由聚合的 BGP 路由器 ID 和聚合路由的路由器的 AS 号；

（2）**COMMUNITY（团体）**：此属性指共享一个公共属性的一组路由器。

4. **可选非过渡（Optional Nontransitive）**：可选非过渡属性并不要求所有的 BGP 实现都支持。如果这些属性被发送到不能对其识别的路由器，这些属性将会被丢弃，不能传送给 BGP 邻居，包括 MED、ORIGINATOR_ID 和 CLUSTER_LIST。

（1）**MED（多出口区分）**：该属性通知 AS 外的路由器采用哪一条路径到达 AS。它也被认为是路由的外部度量，低的 MED 值表示高的优先级。MED 属性在自治系统间交换，但 MED 属性不能传递到第三方 AS；

（2）**ORIGINATOR_ID（起源 ID）**：路由反射器会附加到这个属性上，它携带本 AS 源路由器的路由器 ID,用以防止环路；

（3）**CLUSTER_LIST（簇列表）**：此属性显示了采用的反射路径。

24.1.3 BGP 路由判定

BGP 使用了描述路由特性的很多属性。这些属性和每一个路由一起在 BGP 更新报文中被发送。路由器使用这些属性去选择到目的地的最佳路由。理解 BGP 路由判定的过程很重要的，下面按优先顺序给出了路由器在 BGP 路径选择中的判定过程：

1. 如果下一跳不可达，则不考虑该路由；
2. 优先选取具有最大权重（weight）值的路径,权重是 Cisco 专有属性；
3. 如果权重值相同，优先选取具有最高本地优先级的路由；
4. 如果本地优先级相同，优先选取源自于本路由器（即下一跳为“0.0.0.0”）上 BGP 的路由；
5. 如果本地优先级相同，并且没有源自本路由器的路由，优先选取具有最短 AS 路径的路由；
6. 如果具有相同的 AS 路径长度，优先选取有最低起源代码（IGP<EGP<INCOMPLETE）的路由；
7. 如果起源代码相同，优先选取具有最低 MED 值的路径；
8. 如果 MED 都相同，在 EBGP 路由和联盟 EBGP 路由中，首选 EBGP 路由，在联盟 EBGP 路由和 IBGP 路由中，首选联盟 EBGP 路由；

9. 如果前面所有属性都相同，优先选取离 IGP 邻居最近的路径；
10. 如果内部路径也相同，优先选取具有最低 BGP 路由器 ID 的路径。

24.2 实验 1:IBGP 和 EBGP 基本配置

1. 实验目的

通过本实验可以掌握

（1）启动 BGP 路由进程
（2）BGP 进程中通告网络
（3）IBGP 邻居配置
（4）EBGP 邻居配置
（5）BGP 路由更新源配置
（6）next-hop-self 配置
（7）BGP 路由汇总配置
（8）BGP 路由调试

2. 拓扑结构

实验拓扑如图 24-1 所示。

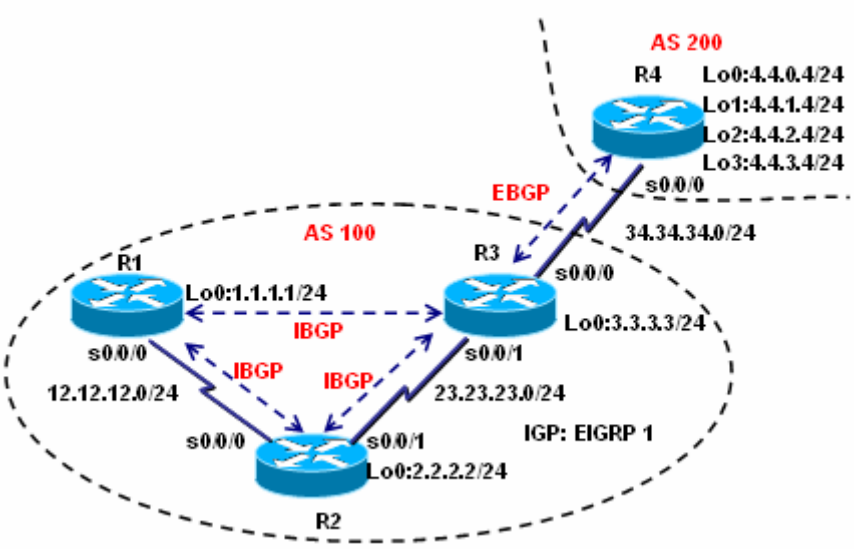

图 24-1 IBGP 和 EBGP 基本配置

3. 实验步骤

因为本实验中 IBGP 的路由器（R1,R2 和 R3）形成全互联（FULL MESH）的邻居关系，所以路由器 R1、R2 和 R3 均关闭同步。IBGP 路由器之间运行的 IGP 是 EIGRP,为了提供 BGP 建立邻居关系的 TCP 连接和 BGP 下一跳可达。

（1）步骤 1：配置路由器 R1

```
R1(config)#router eigrp 1
R1(config-router)#network 1.1.1.0 255.255.255.0
R1(config-router)#network 12.12.12.0 255.255.255.0
R1(config-router)#no auto-summary
R1(config)#router bgp 100                    //启动 BGP 进程
R1(config-router)#no synchronization         //关闭同步
R1(config-router)#bgp router-id 1.1.1.1      //配置 BGP 路由器 ID
```

R1(config-router)#neighbor 2.2.2.2 remote-as 100 //指定邻居路由器及所在的 AS
R1(config-router)#neighbor 2.2.2.2 update-source Loopback0 //指定更新源
R1(config-router)#neighbor 3.3.3.3 remote-as 100
R1(config-router)#neighbor 3.3.3.3 update-source Loopback0
R1(config-router)#network 1.1.1.0 mask 255.255.255.0 //通告网络
R1(config-router)#no auto-summary //关闭自动汇总

（2）步骤 2：配置路由器 R2

R2(config)#router eigrp 1
R2(config-router)#network 2.2.2.0 255.255.255.0
R2(config-router)#network 12.12.12.0 255.255.255.0
R2(config-router)#network 23.23.23.0 255.255.255.0
R2(config-router)#no auto-summary
R2(config)#router bgp 100
R2(config-router)#no synchronization
R2(config-router)#bgp router-id 2.2.2.2
R2(config-router)#neighbor 1.1.1.1 remote-as 100
R2(config-router)#neighbor 1.1.1.1 update-source Loopback0
R2(config-router)#neighbor 3.3.3.3 remote-as 100
R2(config-router)#neighbor 3.3.3.3 update-source Loopback0
R2(config-router)#no auto-summary

（3）步骤 3：配置路由器 R3

R3(config)#router eigrp 1
R3(config-router)#network 3.3.3.0 255.255.255.0
R3(config-router)#network 23.23.23.0 255.255.255.0
R3(config-router)#no auto-summary
R3(config)#router bgp 100
R3(config-router)#no synchronization
R3(config-router)#bgp router-id 3.3.3.3
R3(config-router)#neighbor 1.1.1.1 remote-as 100
R3(config-router)#neighbor 1.1.1.1 update-source Loopback0
R3(config-router)#neighbor 1.1.1.1 next-hop-self

//配置下一跳自我，即对从 EBGP 邻居传入的路由，在通告给 IBGP 邻居时，强迫路由器通告自己是发送 BGP 更新的下一跳，而不是 EBGP 邻居

R3(config-router)#neighbor 2.2.2.2 remote-as 100
R3(config-router)#neighbor 2.2.2.2 update-source Loopback0
R3(config-router)#neighbor 2.2.2.2 next-hop-self
R3(config-router)#neighbor 34.34.34.4 remote-as 200
R3(config-router)#no auto-summary

（4）步骤 4：配置路由器 R4

R4(config)#router bgp 200
R4(config-router)#no synchronization
R4(config-router)#bgp router-id 4.4.4.4
R4(config-router)#neighbor 34.34.34.3 remote-as 100
R4(config-router)#no auto-summary

```
R4(config-router)#network 4.4.0.0 mask 255.255.255.0
R4(config-router)#network 4.4.1.0 mask 255.255.255.0
R4(config-router)#network 4.4.2.0 mask 255.255.255.0
R4(config-router)#network 4.4.3.0 mask 255.255.255.0
R4(config-router)#network 4.4.0.0 mask 255.255.252.0
//用 network 做路由汇总通告
R4(config)#ip route 4.4.0.0 255.255.252.0 null0
//在 IGP 表中构造该汇总路由，否则不能用 network 通告
```

【技术要点】

（1）一台路由器只能启动一个 BGP 进程；

（2）命令“neighbor”后边跟的是邻居路由器 BGP 路由更新源的地址；

（3）BGP 中的“network”命令与 IGP 不同，它只是将 IGP 中存在的路由条目（可以是直连、静态路由或动态路由）在 BGP 中通告。同时“network”命令使用参数“mask”来通告单独的子网。如果 BGP 的自动汇总功能没有关闭，如果在 IGP 路由表中存在子网路由，在 BGP 中可以用“network”命令通告主类网络的。如果 BGP 的自动汇总功能关闭，则通告必须严格匹配掩码长度；

（4）在命令“neighbor” 后边跟“update-source”参数，是用来指定更新源的。如果网络中有多条路径，那么用环回接口建立 TCP 连接，并作为 BGP 路由的更新源，会增加 BGP 的稳健性；

（5）在命令“neighbor” 后边跟“next-hop-self”参数是为了解决下一跳可达的问题，因为当路由通过 EBGP 注入到 AS 时，从 EBGP 获得的下一跳会被不变的在 IBGP 中传递，“next-hop-self”参数使得路由器会把自己作为发送BGP更新的下一跳来通告给IBGP邻居；

（6）BGP 的下一跳是指 BGP 路由表中路由条目的下一跳，也就是相应“neighbor”命令所指的地址。

4.实验调试

（1）show tcp brief

该命令用来查看 TCP 连接信息摘要。

```
R3#show tcp brief
TCB        Local Address          Foreign Address        (state)
64752BAC   3.3.3.3.11002          1.1.1.1.179            ESTAB
64753B5C   3.3.3.3.11000          2.2.2.2.179            ESTAB
6472708    34.34.34.3.11001       34.34.34.4.179         ESTAB
```

以上输出标明路由器 R3 和路由器 R1、R2 和 R4 的 179 端口建立了 TCP 连接。建立 TCP 连接的双方使用 BGP 路由更新源的地址。只要两台路由器之间建立了一条 TCP 连接，就可以形成 BGP 邻居关系。

（2）show ip bgp neighbors

该命令用来查看邻居的 TCP 和 BGP 连接的详细信息。

```
R3#show ip bgp neighbors 34.34.34.4
BGP neighbor is 34.34.34.4,  remote AS 200, external link
  BGP version 4, remote router ID 4.4.4.4
  BGP state = Established, up for 00:50:29
  Last read 00:00:21, hold time is 180, keepalive interval is 60 seconds
```

```
  Neighbor capabilities:
    Route refresh: advertised and received(old & new)
    Address family IPv4 Unicast: advertised and received
......
```

以上输出表明路由器有一个外部 BGP 邻居路由器 R4（**34.34.34.4**）在 AS **200**。此邻居的路由器 ID 号是 **4.4.4.4**。命令“**show ip bgp neighbors**”显示出的信息最重要的一部分是“**BGP state=**”那一行。此行给出了 BGP 连接的状态。“**Established**”状态表示 BGP 对等体间的会话是打开的并正在运行。如果显示的是其它状态，如 Idle、Connect、Active、OpenSent 或 OpenConfirm，那就存在问题。

（3）**show ip bgp summary**

该命令用来查看 BGP 连接的摘要信息。

```
R3#show ip bgp summary
BGP router identifier 3.3.3.3, local AS number 100
//路由器 ID 及本地 AS
BGP table version is 11, main routing table version 11
//BGP 表的内部版本号（BGP 表变化时号码会逐次加 1）和注入到主路由表的最后版本号
5 network entries using 505 bytes of memory
//网络条目和使用的 memory
5 path entries using 240 bytes of memory
//路径条目和使用的 memory
2 BGP path attribute entries using 120 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 889 total bytes of memory
BGP activity 5/0 prefixes, 6/1 paths, scan interval 60 secs

Neighbor    V AS  MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
1.1.1.1     4 100      80      81       11    0    0 00:38:29        1
2.2.2.2     4 100      74      77       11    0    0 01:12:46        0
34.34.34.4  4 200      71      74       11    0    0 01:07:47        4
```

以上输出的邻居表的各个字段的含义如下：

① Neighbor：BGP 邻居的 ID；

② V：BGP 的版本为 4；

③ AS：邻居所在的 AS 号码；

④ MsgRcvd：接收的信息；

⑤ MsgSent：发送的信息；

⑥ TblVer：BGP 表的内部版本号；

⑦ Up/Down：邻居关系建立的时间；

⑧ State/PfxRcd：BGP 连接的状态或者通告的路由前缀。

（4）**show ip bgp**

该命令用来查看 BGP 表的信息。

```
R3#show ip bgp
BGP table version is 11, local router ID is 3.3.3.3
```

```
//BGP 表的内部版本号和本路由器的 BGP 路由器 ID
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
r>i1.1.1.0/24       1.1.1.1                  0    100      0 i
*> 4.4.0.0/24       34.34.34.4               0             0 200 i
*> 4.4.0.0/22       34.34.34.4               0             0 200 i
*> 4.4.1.0/24       34.34.34.4               0             0 200 i
*> 4.4.2.0/24       34.34.34.4               0             0 200 i
*> 4.4.3.0/24       34.34.34.4               0             0 200 i
```

以上输出中，路由条目表项的状态代码（Status codes）的含义解释如下：

① s：表示路由条目被抑制；

② d：表示路由条目由于被惩罚而受到抑制，从而阻止了不稳定路由的发布；

③ h：表示该路由该路由正在被惩罚，但还未达到抑制阀值而使它被抑制；

④ *：表示该路由条目有效；

⑤ >：表示该路由条目最优，可以被传递，达到最优的重要前提是下一跳可达；

⑥ i：表示该路由条目是从 IBGP 邻居学到的；

⑦ r：表示将 BGP 表中的路由条目放入到 IP 路由表中失败。

以上输出中，起源代码（Origin codes）的含义解释如下：

① i：表示路由条目来源为 IGP；

② e：表示路由条目来源为 EGP；

③ ?：表示路由条目来源不清楚，通常是从 IGP 重分布到 BGP 的路由条目。

下面具体地解释 BGP 路由条目

“r>i1.1.1.0/24　　1.1.1.1　　0　　100　　0 i”的含义：

① r：因为路由器 R3 通过 EIGRP 学到“1.1.1.0/24”路由条目，其管理距离为 90，而通过 IBGP 学到“1.1.1.0/24”路由条目的管理距离是 200，而且关闭了同步，BGP 表中的路由条目放入到 IP 路由表中失败，所以出现代码“r”；

② >：表示该路由条目最优，可以继续传递；

③ i：表示该路由条目是从 IBGP 邻居学到的；

④ 1.1.1.1：表示该 BGP 路由的下一跳；

⑤ 0（标题栏对应 Metric）：表示该路由外部度量值即 MED 值为 0；

⑥ 100：表示该路由本地优先级为 100；

⑦ 0（标题栏对应 Weight）：表示该路由的权重值为 0，如果是本地产生的，默认权重值是 32768；如果是从邻居学来的，默认权重值为 0；

⑧ 由于该路由是通过相同 AS 的 IBGP 邻居传递来，所以 PATH 字段为空；

⑨ i：表示路由条目来源为 IGP，它是路由器 R1 用“network”命令通告的。

（5）show ip route

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     1.0.0.0/24 is subnetted, 1 subnets
C       1.1.1.0 is directly connected, Loopback0
     2.0.0.0/24 is subnetted, 1 subnets
D       2.2.2.0 [90/2297856] via 12.12.12.2, 03:03:44, Serial0/0/0
     3.0.0.0/24 is subnetted, 1 subnets
D       3.3.3.0 [90/2809856] via 12.12.12.2, 03:03:44, Serial0/0/0
     4.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
B       4.4.0.0/24 [200/0] via 3.3.3.3, 03:02:52
B       4.4.0.0/22 [200/0] via 3.3.3.3, 01:37:48
B       4.4.1.0/24 [200/0] via 3.3.3.3, 03:02:52
B       4.4.2.0/24 [200/0] via 3.3.3.3, 03:02:52
B       4.4.3.0/24 [200/0] via 3.3.3.3, 03:02:52
     23.0.0.0/24 is subnetted, 1 subnets
D       23.23.23.0 [90/2681856] via 12.12.12.2, 03:03:45, Serial0/0/0
     12.0.0.0/24 is subnetted, 1 subnets
C       12.12.12.0 is directly connected, Serial0/0/0
R3#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, 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
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     34.0.0.0/24 is subnetted, 1 subnets
C       34.34.34.0 is directly connected, Serial0/0/0
     1.0.0.0/24 is subnetted, 1 subnets
D       1.1.1.0 [90/2809856] via 23.23.23.2, 02:17:48, Serial0/0/1
     2.0.0.0/24 is subnetted, 1 subnets
D       2.2.2.0 [90/2297856] via 23.23.23.2, 02:51:36, Serial0/0/1
     3.0.0.0/24 is subnetted, 1 subnets
C       3.3.3.0 is directly connected, Loopback0
     4.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
B       4.4.0.0/24 [20/0] via 34.34.34.4, 02:45:33
B       4.4.0.0/22 [20/0] via 34.34.34.4, 00:51:20
```

```
B       4.4.1.0/24 [20/0] via 34.34.34.4, 02:45:33
B       4.4.2.0/24 [20/0] via 34.34.34.4, 02:45:33
B       4.4.3.0/24 [20/0] via 34.34.34.4, 02:45:34
     23.0.0.0/24 is subnetted, 1 subnets
C       23.23.23.0 is directly connected, Serial0/0/1
     12.0.0.0/24 is subnetted, 1 subnets
D       12.12.12.0 [90/2681856] via 23.23.23.2, 02:17:53, Serial0/0/1
```

以上输出表明 IBGP 的管理距离是 200,EBGP 的管理距离是 20。

（6）PING

在路由器 R1 上 ping 4.4.0.4，结果是不通的，原因很简单，就是路由器 R1 和 R2 的路由表中没有 34.34.34.0 的路由，此时如果执行扩展 ping，就是通的：

```
R1#ping
Protocol [ip]:
Target IP address: 4.4.0.4
Repeat count [5]: 2
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 1.1.1.1
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 2, 100-byte ICMP Echos to 4.4.0.4, timeout is 2 seconds:
Packet sent with a source address of 1.1.1.1
!!
```

如果一定要标准 ping 的话，无非就是让路由器 R1 和 R2 学到“34.34.34.0”的路由，方法很多，比如在路由器 R3 上重分布直连。

（7）在 R1 上打开 BGP 同步，然后查看 BGP 表：

```
R1(config)#router bgp 100
R1(config-router)#synchronization     //打开同步
R1#clear ip bgp *  //重置 BGP 连接
R1#show ip bgp
BGP table version is 1, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i4.4.0.0/24       3.3.3.3                  0    100      0 200 i
* i4.4.0.0/22       3.3.3.3                  0    100      0 200 i
```

```
* i4.4.1.0/24        3.3.3.3                  0    100      0 200 i
* i4.4.2.0/24        3.3.3.3                  0    100      0 200 i
* i4.4.3.0/24        3.3.3.3                  0    100      0 200 i
```

以上输出表明 BGP 路由不是被优化的，因为 IGP 的路由表中并没有这些路由条目。

（8）删除路由器 R1 和 R3 之间的邻居关系，保持路由器 R1 和 R2 建立邻居关系，路由器 R2 和 R3 建立邻居关系，操作如下：

```
R1(config)#router bgp 100
R1(config-router)#no synchronization
R1(config-router)#no neighbor 3.3.3.3
R3(config)#router bgp 100
R3(config-router)#no neighbor 1.1.1.1
```

在路由器 R1 和 R2 上查看 BGP 表：

```
R1#show ip bgp
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       0.0.0.0                  0         32768 i
R2#show ip bgp
BGP table version is 8, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
r>i1.1.1.0/24       1.1.1.1                  0    100      0 i
*>i4.4.0.0/24       3.3.3.3                  0    100      0 200 i
*>i4.4.0.0/22       3.3.3.3                  0    100      0 200 i
*>i4.4.1.0/24       3.3.3.3                  0    100      0 200 i
*>i4.4.2.0/24       3.3.3.3                  0    100      0 200 i
*>i4.4.3.0/24       3.3.3.3                  0    100      0 200 i
```

以上输出表明路由器 R2 并没有将路由器 R3 通告的路由通告给路由器 R1，这也进一步验证了 IBGP 水平分割的基本原理：通过 IBGP 学到的路由不能通告给相同 AS 内的其它的 IBGP 邻居。通常的解决办法有两个：IBGP 形成全互联邻居关系或使用路由反射器。

24.3 实验 2:BGP 地址聚合

1.实验目的

通过本实验可以掌握

（1）启动 BGP 路由进程

（2）BGP 中通告网络

（3）EBGP 邻居配置

（4）BGP 地址聚合配置和调试

（5）地址聚合中参数“as-set”含义

（6）地址聚合中参数“summary-only”含义

（7）地址聚合中参数“suppress-map”含义

2. 拓扑结构

实验拓扑如图 24-2 所示。

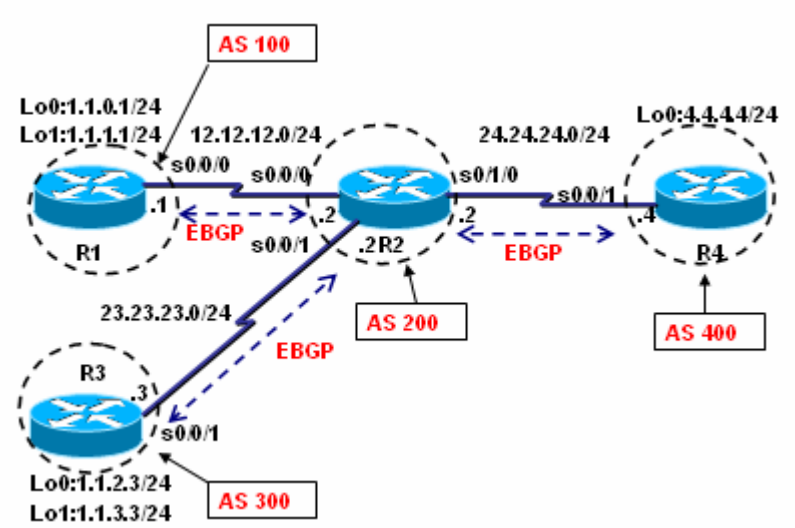

图 24-2 BGP 地址聚合配置

3. 实验步骤

本实验实现在路由器 R2 上将路由器 R1 和路由器 R3 通告的环回接口的路由进行地址聚合，并通告给路由器 R4。在路由器 R1、R3、R4 配置静态路由实现网络互通。

（1）步骤 1：配置路由器 R1

```
R1(config)#router bgp 100
R1(config-router)#no synchronization
R1(config-router)#no auto-summary
R1(config-router)#bgp router-id 1.1.1.1
R1(config-router)#neighbor 12.12.12.2 remote-as 200
R1(config-router)#network 1.1.0.0 mask 255.255.255.0
R1(config-router)#network 1.1.1.0 mask 255.255.255.0
R1(config)#ip route 24.24.24.0 255.255.255.0 12.12.12.2
R1(config)#ip route 23.23.23.0 255.255.255.0 12.12.12.2
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router bgp 200
R2(config-router)#no synchronization
R2(config-router)#no auto-summary
R2(config-router)#bgp router-id 2.2.2.2
R2(config-router)#neighbor 12.12.12.1 remote-as 100
R2(config-router)#neighbor 23.23.23.3 remote-as 300
R2(config-router)#neighbor 24.24.24.4 remote-as 400
R2(config-router)#aggregate-address 1.1.0.0 255.255.252.0 //配置地址聚合
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router bgp 300
```

```
R3(config-router)#no synchronization
R3(config-router)#no auto-summary
R3(config-router)#bgp router-id 3.3.3.3
R3(config-router)#neighbor 23.23.23.2 remote-as 200
R3(config-router)#network 1.1.2.0 mask 255.255.255.0
R3(config-router)#network 1.1.3.0 mask 255.255.255.0
R3(config)#ip route 12.12.12.0 255.255.255.0 23.23.23.2
R3(config)#ip route 24.24.24.0 255.255.255.0 23.23.23.2
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router bgp 400
R4(config-router)#no synchronization
R4(config-router)#no auto-summary
R4(config-router)#bgp router-id 4.4.4.4
R4(config-router)#neighbor 24.24.24.2 remote-as 200
R4(config-router)#network 4.4.4.0 mask 255.255.255.0
R4(config)#ip route 12.12.12.0 255.255.255.0 24.24.24.2
R4(config)#ip route 23.23.23.0 255.255.255.0 24.24.24.2
```

4.实验调试

（1）在路由器 R1、R4 上查看 BGP 表：

```
R1#show ip bgp
BGP table version is 21, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.0.0/24       0.0.0.0                  0         32768 i
*> 1.1.0.0/22       12.12.12.2               0             0 200 i
*> 1.1.1.0/24       0.0.0.0                  0         32768 i
*> 1.1.2.0/24       12.12.12.2                             0 200 300 i
*> 1.1.3.0/24       12.12.12.2                             0 200 300 i
*> 4.4.4.0/24       12.12.12.2                             0 200 400 i
R4#show ip bgp
BGP table version is 45, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.0.0/24       24.24.24.2                             0 200 100 i
*> 1.1.0.0/22       24.24.24.2               0             0 200 i
*> 1.1.1.0/24       24.24.24.2                             0 200 100 i
*> 1.1.2.0/24       24.24.24.2                             0 200 300 i
*> 1.1.3.0/24       24.24.24.2                             0 200 300 i
```

*> 4.4.4.0/24 **0.0.0.0** 0 32768 i

以上输出表明：

① 路由器 R1、R4 收到“**1.1.0.0/22**”聚合路由，通过 AS-PATH 属性可以看出，执行地址聚合的路由器 R2 成为新路由的创造者，原来 AS-PATH 属性丢失；

② 路由器 R4 同时也收到 4 条明细路由，在显示的 AS-PATH 序列中，路由的始发 AS 在列表的末端（右侧），每个收到该路由，并把它传递给其它 AS 的 BGP 对等体会把它自己的 AS 追加在列表的开头（左侧）；

③ BGP 路由器下一跳为“**0.0.0.0**”，表示该 BGP 路由起源本地，Weight 值为“32768”；

④ 因为所有 BGP 路由条目的代码为“*>”，所以所有 BGP 路由条目都为最优。

（2）“**as-set**”参数可以使 BGP 聚合路由不丢失原来的 AS-PATH 属性，从而避免路由环路，在路由器 R2 上操作如下：

R2(config-router)**#aggregate-address 1.1.0.0 255.255.252.0 as-set**

在路由器 R1、R4 上再次查看 BGP 表：

R1**#show ip bgp**

```
BGP table version is 22, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.0.0/24       0.0.0.0                  0         32768 i
*> 1.1.1.0/24       0.0.0.0                  0         32768 i
*> 1.1.2.0/24       12.12.12.2                             0 200 300 i
*> 1.1.3.0/24       12.12.12.2                             0 200 300 i
*> 4.4.4.0/24       12.12.12.2                             0 200 400 i
```

R4**#show ip bgp**

```
BGP table version is 56, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.0.0/24       24.24.24.2                             0 200 100 i
*> 1.1.0.0/22       24.24.24.2               0             0 200 {100,300} i
*> 1.1.1.0/24       24.24.24.2                             0 200 100 i
*> 1.1.2.0/24       24.24.24.2                             0 200 300 i
*> 1.1.3.0/24       24.24.24.2                             0 200 300 i
*> 4.4.4.0/24       0.0.0.0                  0         32768 i
```

以上输出表明：

① 路由器 R4 上收到的汇总路由“**1.1.0.0/22**”中，AS-PATH 包含了被聚合路由中所有的 AS 号码的集合“**{100,300}**”；

② 聚合路由正是由于携带了所有的 AS，所以在路由器 R1 的 BGP 表中没有出现，当然在路由器 R3 的 BGP 表中也不会出现。

【技术要点】

BGP 使用 AS-PATH 属性作为路由更新的一部分来确保没有路由环路。因为在 BGP 对等体之间传递的每条路由都携带它所经过的 AS 号码序列表，如果该路由被通告给它始发的 AS，该 AS 路由器将在 AS 序列表中看到自己的 AS，它将不接受该路由。以下的输出充分的说明了这一点；

```
R2#show ip bgp neighbor 12.12.12.1 advertised-routes
BGP table version is 8, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.0.0/22       0.0.0.0                  100  32768 {100,300} i
*> 1.1.2.0/24       23.23.23.3               0             0 300 i
*> 1.1.3.0/24       23.23.23.3               0             0 300 i
*> 4.4.4.0/24       24.24.24.4               0             0 400 i
```

以上输出表明路由器 R2 仍然向邻居 12.12.12.1 发送聚合路由“1.1.0.0/22”。

```
R1#show ip bgp neighbors 12.12.12.2 received-routes
BGP table version is 22, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.2.0/24       12.12.12.2                             0 200 300 i
*> 1.1.3.0/24       12.12.12.2                             0 200 300 i
*> 4.4.4.0/24       12.12.12.2                             0 200 400 i
```

以上输出表明路由器 R1 没有接收聚合路由“1.1.0.0/22”，因为它发现聚合路由条目中的 AS-PATH 属性列表“{100,300}”中包含自己的 AS 号码 100，所以不接收。同理，路由器 R3 也不会接收该聚合路由条目。

【提示】

要执行“show ip bgp neighbors 12.12.12.2 received-routes”命令，必须完成下面这条命令：

```
R1(config-router)#neighbor 12.12.12.2 soft-reconfiguration inbound
```

（3）如果在路由器 R4 上只想看到汇总路由，没有明细路由，“summary-only”参数可以实现，在路由器 R2 上的配置如下；

```
R2(config-router)#aggregate-address 1.1.0.0 255.255.252.0 as-set summary-only
```

在路由器 R2、R4 上查看 BGP 表：

```
R2#show ip bgp
BGP table version is 14, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
s> 1.1.0.0/24       12.12.12.1               0             0 100 i
*> 1.1.0.0/22       0.0.0.0                        100  32768 {100,300} i
s> 1.1.1.0/24       12.12.12.1               0             0 100 i
s> 1.1.2.0/24       23.23.23.3               0             0 300 i
s> 1.1.3.0/24       23.23.23.3               0             0 300 i
*> 4.4.4.0/24       24.24.24.4               0             0 400 i
R4#show ip bgp
BGP table version is 62, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.0.0/22       24.24.24.2               0             0 200 {100,300} i
*> 4.4.4.0/24       0.0.0.0                  0         32768 i
```

以上输出表明：

① 路由器 R2 上所有被聚合的明细路由被标记为“s”，表示被抑制，不被发送；

② 路由器 R4 只收到一跳聚合路由“1.1.0.0/22 ”。如果不加“as-set”参数，则路由器 R1、R3 也会收到该聚合路由。

（4）如果有特殊的需求，在聚合后只抑制部分明细路由条目，参数“suppress-map”可以完成。本实验要求路由器 R2 地址聚合后，要求路由器 R1 的两条明细路由被抑制，而路由器 R3 的明细路由要求传递给路由器 R4，路由器 R2 配置步骤如下：

```
R2(config)#ip prefix-list 1 permit 1.1.0.0/24 //匹配路由条目，以便进行控制
R2(config)#ip prefix-list 1 permit 1.1.1.0/24
R2(config)#route-map sup permit 10
R2(config-route-map)#match ip address prefix-list 1
R2(config)#router bgp 200
R2(config-router)#aggregate-address     1.1.0.0     255.255.252.0     as-set
suppress-map sup
```

分别在四台路由器查看 BGP 表：

```
R1#show ip bgp
BGP table version is 28, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.0.0/24       0.0.0.0                  0         32768 i
*> 1.1.1.0/24       0.0.0.0                  0         32768 i
*> 1.1.2.0/24       12.12.12.2                             0 200 300 i
*> 1.1.3.0/24       12.12.12.2                             0 200 300 i
```

```
*> 4.4.4.0/24        12.12.12.2                             0 200 400 i
R2#show ip bgp
BGP table version is 20, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
s> 1.1.0.0/24       12.12.12.1               0             0 100 i
*> 1.1.0.0/22       0.0.0.0                        100  32768 {100,300} i
s> 1.1.1.0/24       12.12.12.1               0             0 100 i
*> 1.1.2.0/24       23.23.23.3               0             0 300 i
*> 1.1.3.0/24       23.23.23.3               0             0 300 i
*> 4.4.4.0/24       24.24.24.4               0             0 400 i
R3#show ip bgp
BGP table version is 58, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.2.0/24       0.0.0.0                  0         32768 i
*> 1.1.3.0/24       0.0.0.0                  0         32768 i
*> 4.4.4.0/24       23.23.23.2                             0 200 400 i
R4#show ip bgp
BGP table version is 64, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.0.0/22       24.24.24.2               0             0 200 {100,300} i
*> 1.1.2.0/24       24.24.24.2                             0 200 300 i
*> 1.1.3.0/24       24.24.24.2                             0 200 300 i
*> 4.4.4.0/24       0.0.0.0                  0         32768 i
```

以上输出表明：

① 由于在路由器 R2 上将路由器 R1 的明细路由“1.1.0.0/24”和“1.1.1.0/24”抑制，所以路由器 R3 和 R4 不能收到；

② 由于在路由器 R2 上没有将明细路由“1.1.2.0/24”和“1.1.3.0/24”抑制，又没有配置“summary-only”参数，所以四台路由器全部收到“1.1.2.0/24”和“1.1.3.0/24”的路由条目；

③ 由于配置了“as-set”参数，所以只有 R4 收到汇总路由“1.1.0.0/22”。

24.4 用 BGP 属性控制选路

BGP 具有丰富的属性，但本节只研究 ORIGIN、AS-PATH、LOCAL_PREF、WEIGHT 和 MED 属性。本节的实验是一个有机的整体，根据 BGP 路由判定的顺序（优先级别从低到高）设计实验，每个分解实验都是以较高的优先级别影响前面分解实验的 BGP 路由选路。实验的拓扑如图 24-3 所示。通过修改 ORIGIN、AS-PATH、LOCAL_PREF、WEIGHT 属性来控制在 AS 100 内路由器 R1、R2 和 R3 对路由器 R4 上通告的 4.4.4.0/24 路由的选路。最后通过在路由器 R2 和 R3 上发布环回接口来控制从路由器 R4 进入 AS 100 的选路。本实验中 IBGP 的路由器（R1、R2 和 R3）形成全互联（FULL MESH）的邻居关系。IGP 运行 EIGRP。在完成每个分解实验后，最好用“clear ip bgp *”清除一下 BGP 路由表，然后再查看结果。

24.4.1 实验 3：用 BGP ORIGIN 属性控制选路

1. 实验目的

通过本实验可以掌握

（1）BGP 路由传递的条件

（2）ORIGIN 代码的优先级

（3）用 ORIGIN 属性选路的原则

2. 拓扑结构

实验拓扑如图 24-3 所示。

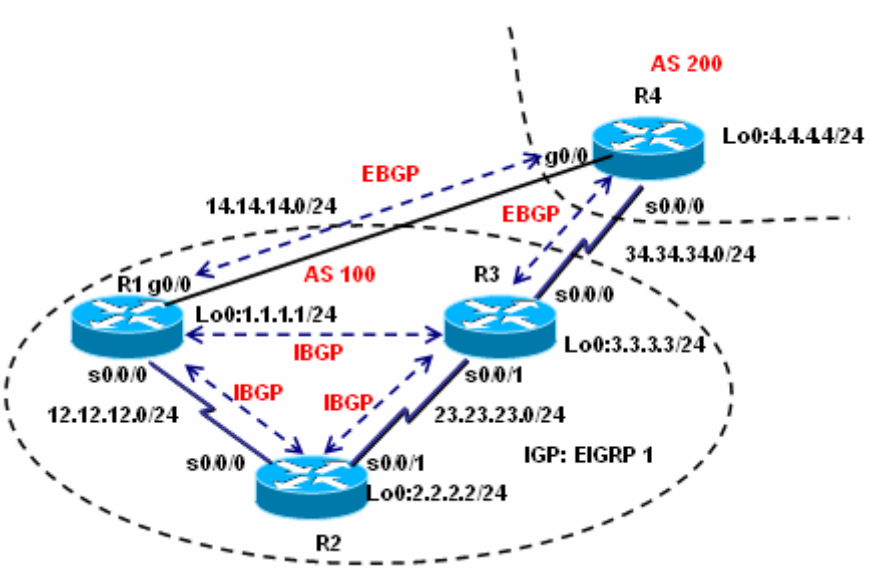

图 24-3 用 BGP 属性控制选路

3. 实验步骤

本实验是在路由器 R4 上配置“4.4.4.0/24”的起源代码属性为 EGP，并通过 EBGP 邻居“14.14.14.1”传入 AS 100 内，然后观察路由器 R1、R2 和 R3 对路由器 R4 上通告的“4.4.4.0/24”路由的选路。

（1）步骤 1：配置路由器 R1

```
R1(config)#router eigrp 1
R1(config-router)#network 1.1.1.0 255.255.255.0
R1(config-router)#network 12.12.12.0 255.255.255.0
R1(config-router)#no auto-summary
```

```
R1(config)#router bgp 100
R1(config-router)#no synchronization
R1(config-router)#no auto-summary
R1(config-router)#bgp router-id 1.1.1.1
R1(config-router)#neighbor 2.2.2.2 remote-as 100
R1(config-router)#neighbor 2.2.2.2 update-source Loopback0
R1(config-router)#neighbor 2.2.2.2 next-hop-self
R1(config-router)#neighbor 3.3.3.3 remote-as 100
R1(config-router)#neighbor 3.3.3.3 update-source Loopback0
R1(config-router)#neighbor 3.3.3.3 next-hop-self
R1(config-router)#neighbor 14.14.14.4 remote-as 200
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router eigrp 1
R2(config-router)#network 2.2.2.0 255.255.255.0
R2(config-router)#network 12.12.12.0 255.255.255.0
R2(config-router)#network 23.23.23.0 255.255.255.0
R2(config-router)#no auto-summary
R2(config)#router bgp 100
R2(config-router)#no synchronization
R2(config-router)#no auto-summary
R2(config-router)#bgp router-id 2.2.2.2
R2(config-router)#neighbor 1.1.1.1 remote-as 100
R2(config-router)#neighbor 1.1.1.1 update-source Loopback0
R2(config-router)#neighbor 3.3.3.3 remote-as 100
R2(config-router)#neighbor 3.3.3.3 update-source Loopback0
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router eigrp 1
R3(config-router)#network 3.3.3.0 255.255.255.0
R3(config-router)#network 23.23.23.0 255.255.255.0
R3(config-router)#no auto-summary
R3(config)#router bgp 100
R3(config-router)#no synchronization
R3(config-router)#no auto-summary
R3(config-router)#bgp router-id 3.3.3.3
R3(config-router)#neighbor 1.1.1.1 remote-as 100
R3(config-router)#neighbor 1.1.1.1 update-source Loopback0
R3(config-router)#neighbor 1.1.1.1 next-hop-self
R3(config-router)#neighbor 2.2.2.2 remote-as 100
R3(config-router)#neighbor 2.2.2.2 update-source Loopback0
R3(config-router)#neighbor 2.2.2.2 next-hop-self
R3(config-router)#neighbor 34.34.34.4 remote-as 200
```

（4）步骤 4：配置路由器 R4

```
R4(config-router)#ip prefix-list 1 permit 4.4.4.0/24
R4(config)#route-map egp permit 10
```

```
R4(config-route-map)#match ip address prefix-list 1
R4(config-route-map)#set origin egp 900    //设置起源代码
R4(config)#router bgp 200
R4(config-router)#no synchronization
R4(config-router)#no auto-summary
R4(config-router)#bgp router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 mask 255.255.255.0
R4(config-router)#neighbor 14.14.14.1 remote-as 100
R4(config-router)#neighbor 14.14.14.1 route-map egp out
//在出方向为去往邻居 14.14.14.1 的路由设置策略
R4(config-router)#neighbor 34.34.34.3 remote-as 100
```

4.实验调试

在路由器 R1、R2 和 R3 上查看 BGP 表：

```
R1#show ip bgp
BGP table version is 6, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i4.4.4.0/24       3.3.3.3                  0    100      0 200 i
*                   14.14.14.4               0             0 200 e
R2#show ip bgp
BGP table version is 5, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i4.4.4.0/24       3.3.3.3                  0    100      0 200 i
R3#show ip bgp
BGP table version is 4, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 4.4.4.0/24       34.34.34.4               0             0 200 i
```

以上输出表明路由器 R1 学到两条关于“4.4.4.0/24”的路由，但是由于起源代码“i”优先于“e”，所以从路由器 R3 学到的路由被优化，而从邻居路由器 R4 学到的路由不能被优化（路由代码只为“*”，没有“>”），不能继续通告给路由器 R2 和 R3,所以路由器 R2 和 R3 只有一条关于“4.4.4.0/24”的路由。

24.4.2 实验 4：用 BGP AS-PATH 属性控制选路

1.实验目的

通过本实验可以掌握

（1）AS-PATH 控制路由环路的原理

（2）配置 AS-PATH 属性

（3）用 AS-PATH 属性选路原则

2.拓扑结构

实验拓扑如图 24-3 所示。

3.实验步骤

路由器 R1、R2 和 R3 上的配置和前面 **24.4.1 实验 3** 相同，路由器 R4 改动配置如下：

```
R4(config)#ip prefix-list 1 permit 4.4.4.0/24
R4(config)#route-map aspath permit 10
R4(config-route-map)#match ip address prefix-list 1
R4(config-route-map)#set as-path prepend 600 700 //为匹配的路由条目追加 AS
R4(config)#router bgp 200
R4(config-router)#neighbor 34.34.34.3 route-map aspath out
```

4.实验调试

在路由器 R1、R2 和 R3 上查看 BGP 表：

```
R1#show ip bgp
BGP table version is 11, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 4.4.4.0/24       14.14.14.4               0             0 200 e
R2#show ip bgp
BGP table version is 9, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i4.4.4.0/24       1.1.1.1                  0    100      0 200 e
R3#show ip bgp
BGP table version is 9, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i4.4.4.0/24       1.1.1.1                  0    100      0 200 e
*                   34.34.34.4               0             0 200 600 700 i
```

以上输出表明路由器 R3 学到两条关于“4.4.4.0/24”的路由，但是由于下一跳为“1.1.1.1”的路由的 AS-PATH 比下一跳为“34.34.34.4”的路由的 AS-PATH 短，所以优选

下一跳为“1.1.1.1”的路由，而下一跳为“34.34.34.4”的路由不能被优化（路由代码为“*”），不能继续通告给路由器 R1 和 R2,所以路由器 R1 和 R2 只有一条关于“4.4.4.0/24”的路由。同时也说明 BGP 在路由判定时 AS-PATH 属性是优于 ORIGIN 属性的。

24.4.3 实验 5：用 BGP LOCAL_PREF 属性控制选路

1.实验目的

通过本实验可以掌握

（1）配置 LOCAL_PREF 属性

（2）用 LOCAL_PREF 属性选路原则

2.拓扑结构

实验拓扑如图 24-3 所示。

3.实验步骤

路由器 R1、R2 和 R4 上的配置和 **24.4.2 实验 4** 相同，路由器 R3 改动配置如下：

```
R3(config)#ip prefix-list 1 permit 4.4.4.0/24
R3(config)#route-map local permit 10
R3(config-route-map)#match ip address prefix-list 1
R3(config-route-map)#set local-preference 2000 //修改 LOCAL_PREF 值
R3(config-router)#neighbor 34.34.34.4 route-map local in
//对从邻居 34.34.34.4 进入的路由条目设置策略
```

4.实验调试

在路由器 R1、R2 和 R3 上查看 BGP 表：

```
R1#show ip bgp
BGP table version is 2, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  4.4.4.0/24       14.14.14.4               0             0 200 e
*>i                 3.3.3.3                  0    2000     0 200 600 700 i
R2#show ip bgp
BGP table version is 2, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*>i4.4.4.0/24       3.3.3.3                  0    2000     0 200 600 700 i
R3#show ip bgp

BGP table version is 2, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete
```

```
   Network          Next Hop            Metric LocPrf Weight Path
*> 4.4.4.0/24       34.34.34.4               0    2000        0 200 600 700 i
```

以上输出表明路由器 R1 学到两条关于“4.4.4.0/24”的路由，但是由于下一跳为“3.3.3.3”的路由本地优先级的值比下一跳为“14.14.14.4”的路由的本地优先级的值高，所以优选下一跳为“3.3.3.3”的路由，而下一跳为“14.14.14.4”的路由不能被优化，不能继续通告给路由器 R2 和 R3,所以路由器 R2 和 R3 只有一条关于“4.4.4.0/24”的路由。同时也说明 BGP 在路由判定时本地优先级属性是优于 AS-PATH 属性的。

【提示】

（1）默认情况下，本地优先级的值为 100；

（2）本地优先级属性只在 AS 内部传递，不会通告给 EBGP 邻居；

（3）本地优先级属性值越高，路由的优选程度越高；

（4）命令“**bgp default local-preference**”也可以修改本地优先级属性，只是说用 route-map 设置本地优先级灵活性更大。

24.4.4 实验 6：用 BGP WEIGHT 属性控制选路

1.实验目的

通过本实验可以掌握

（1）配置 Weight 属性

（2）用 Weight 属性选路原则

2.拓扑结构

实验拓扑如图 24-3 所示。

3.实验步骤

路由器 R2、R3 和 R4 上的配置和 **24.4.3 实验 5** 相同，路由器 R1 改动配置如下：

```
R1(config)#router bgp 100
R1(config-router)#neighbor 2.2.2.2 weight 200
//为从 2.2.2.2 学到路由设置权重值
R1(config-router)#neighbor 3.3.3.3 weight 200
//为从 3.3.3.3 学到路由设置权重限值
R1(config-router)#neighbor 14.14.14.4 weight 500
//为从 14.14.14.4 学到路由设置权重值
```

4.实验调试

在路由器 R1、R2 和 R3 上查看 BGP 表：

```
R1#show ip bgp
BGP table version is 3, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 4.4.4.0/24       14.14.14.4               0             500 200 e
* i                 3.3.3.3                  0    2000     200 200 600 700 i
R2#show ip bgp
```

```
BGP table version is 2, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i4.4.4.0/24       1.1.1.1                  0    100      0 200 e
*>i                 3.3.3.3                  0   2000      0 200 600 700 i
R3#show ip bgp
BGP table version is 2, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
* i4.4.4.0/24       1.1.1.1                  0    100      0 200 e
*>                  34.34.34.4               0   2000      0 200 600 700 i
```

以上输出表明路由器 R1 学到两条关于“4.4.4.0/24”的路由，但是由于下一跳为“14.14.14.4”的路由 weight 值比下一跳为“3.3.3.3”的路由的 weight 值高，所以优选下一跳为“14.14.14.4”的路由。因为 weight 属性只影响本地路由器选路，所以对于路由器 R2 和 R3 仍是通过本地优先级选路。从路由器 R1 的选路说明 BGP 在路由判定时 weight 属性是优于本地优先级属性的。

24.4.5 实验 7：用 MED 属性控制选路

1.实验目的

通过本实验可以掌握

（1）配置 MED 属性

（2）用 MED 属性选路原则

2.拓扑结构

实验拓扑如图 24-3 所示。本实验需要在路由器 R2 添加一个环回地址 Lo1:20.1.1.2/24，并在 BGP 中发布；在路由器 R3 添加一个环回地址 Lo1:30.1.1.3/24，并在 BGP 中发布。通过设置 MED 属性，使得在路由器 R4 上访问 30.1.1.3 的时候走 R4→R3→R4 路径；在 R4 上访问 20.1.1.2 的时候走 R4→R1→R2→R1→R4 的路径。

3.实验步骤

路由器 R4 上的配置和 **24.4.4 实验 6** 相同，路由器 R1 R2、R3 改动配置如下：

（1）步骤 1：配置路由器 R1

```
R1(config)#ip prefix-list 20 permit 20.1.1.0/24
R1(config)#ip prefix-list 30 permit 30.1.1.0/24
R1(config)#route-map med permit 10
R1(config-route-map)#match ip address prefix-list 20
R1(config-route-map)#set metric 50   //设置 MED 值
R1(config)#route-map med permit 20
R1(config-route-map)#match ip address prefix-list 30
R1(config-route-map)#set metric 100
```

```
R1(config)#route-map med permit 30
R1(config-route-map)#router bgp 100
R1(config-router)#neighbor 14.14.14.4 route-map med out
```

（2）步骤 2：配置路由器 R2

```
R2(config-if)#router bgp 100
R2(config-router)#network 20.1.1.0 mask 255.255.255.0
```

（3）步骤 3：配置路由器 R3

```
R3(config)#ip prefix-list 20  permit 20.1.1.0/24
R3(config)#ip prefix-list 30  permit 30.1.1.0/24
R3(config)#route-map med permit 10
R3(config-route-map)#match ip address prefix-list 20
R3(config-route-map)#set metric 100
R3(config)#route-map med permit 20
R3(config-route-map)#match ip address prefix-list 30
R3(config-route-map)#set metric 50
R3(config)#route-map med permit 30
R3(config)#router bgp 100
R3(config-router)#network 30.1.1.0 mask 255.255.255.0
R3(config-router)#neighbor 34.34.34.4 route-map med out
```

4.实验调试

（1）在路由器 R4 查看 BGP 表：

```
R4#show ip bgp
BGP table version is 5, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 4.4.4.0/24       0.0.0.0                  0         32768 i
*  20.1.1.0/24      34.34.34.3             100             0 100 i
*>                  14.14.14.1              50             0 100 i
*> 30.1.1.0/24      34.34.34.3              50             0 100 i
*                   14.14.14.1             100             0 100 i
```

以上输出表明路由器 R4 学到的 BGP 路由是携带了 MED 的值，而且优选 MED 值低的路径。

【提示】

① MED 只用来向 EBGP 邻居发送；

② MED 用来影响外部 AS 选路；

③ 进入到一个 AS 中的 MED 属性是不会从这个 AS 中再传递出去；

④ MED 的值越低，路由的优选程度越高。

（2）用扩展 ping 跟踪路径：

```
R4#ping
Protocol [ip]:
```

```
Target IP address: 20.1.1.2
Repeat count [5]: 1
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 4.4.4.4
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]: R
Number of hops [ 9 ]: 6
Loose, Strict, Record, Timestamp, Verbose[RV]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 20.1.1.2, timeout is 2 seconds:
Packet sent with a source address of 4.4.4.4
Packet has IP options:  Total option bytes= 27, padded length=28
 Record route: <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)

Reply to request 0 (148 ms).  Received packet has options
 Total option bytes= 28, padded length=28
 Record route:
   (14.14.14.4)
   (12.12.12.1)
   (20.1.1.2)
   (23.23.23.2)
   (34.34.34.3)
   (4.4.4.4)
   <*>
 End of list
```

以上输出表明在 R4 上访问 20.1.1.2 的时候走 R4→R1→R2→R1→R4 的路径。

```
R4#ping
Protocol [ip]:
Target IP address: 30.1.1.3
Repeat count [5]: 1
Datagram size [100]:
Timeout in seconds [2]:
```

```
Extended commands [n]: y
Source address or interface: 4.4.4.4
Type of service [0]:
Set DF bit in IP header? [no]:
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]: R
Number of hops [ 9 ]: 4
Loose, Strict, Record, Timestamp, Verbose[RV]:
Sweep range of sizes [n]:
Type escape sequence to abort.
Sending 1, 100-byte ICMP Echos to 30.1.1.3, timeout is 2 seconds:
Packet sent with a source address of 4.4.4.4
Packet has IP options:  Total option bytes= 19, padded length=20
 Record route: <*>
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
   (0.0.0.0)
Reply to request 0 (248 ms).  Received packet has options
 Total option bytes= 20, padded length=20
 Record route:
   (34.34.34.4)
   (30.1.1.3)
   (34.34.34.3)
   (4.4.4.4)
   <*>
 End of list
```

以上输出表明在路由器 R4 上访问 30.1.1.3 的时候走 R4→R3→R4 路径。

24.5 实验 8：路由反射器（RR）配置

1.实验目的

通过本实验可以掌握

（1）RR 的反射原理和反射规则

（2）RR 的配置

（3）ORIGINATOR_ID（起源 ID）属性

（4）CLUSTER_LIST（簇列表）属性

2.拓扑结构

实验拓扑如图 24-4 所示。

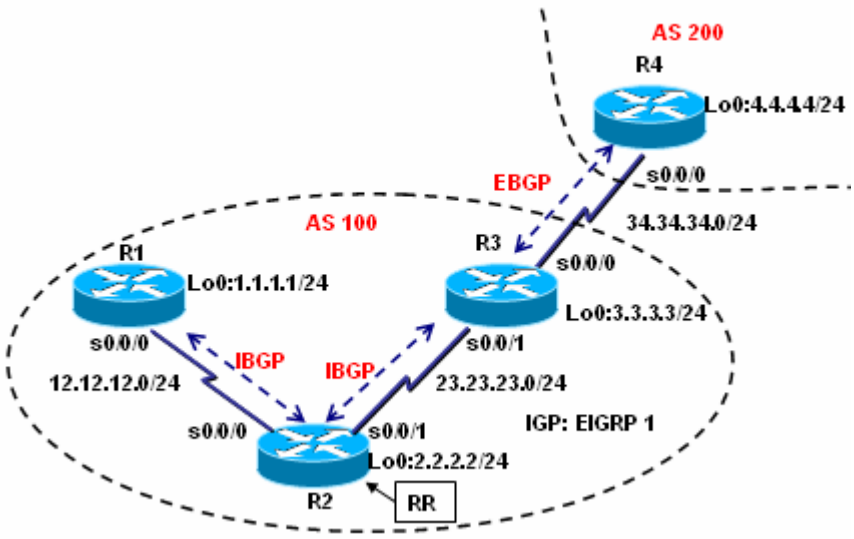

图 24-4 路由反射器（RR）配置

本实验中，路由器 R2 作为路由反射器，路由器 R1 和 R3 作为它的客户端。

3. 实验步骤

（1）步骤 1：配置路由器 R1

```
R1(config)#router eigrp 1
R1(config-router)#network 1.1.1.0 255.255.255.0
R1(config-router)#network 12.12.12.0 255.255.255.0
R1(config-router)#no auto-summary
R1(config)#router bgp 100
R1(config-router)#no synchronization
R1(config-router)#no auto-summary
R1(config-router)#bgp router-id 1.1.1.1
R1(config-router)#neighbor 2.2.2.2 remote-as 100
R1(config-router)#neighbor 2.2.2.2 update-source Loopback0
R1(config-router)#network 1.1.1.0 mask 255.255.255.0
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router eigrp 1
R2(config-router)#network 2.2.2.0 255.255.255.0
R2(config-router)#network 12.12.12.0 255.255.255.0
R2(config-router)#network 23.23.23.0 255.255.255.0
R2(config-router)#no auto-summary
R2(config)#router bgp 100
R2(config-router)#no synchronization
R2(config-router)#no auto-summary
R2(config-router)#bgp router-id 2.2.2.2
R2(config-router)#neighbor 1.1.1.1 remote-as 100
R2(config-router)#neighbor 1.1.1.1 update-source Loopback0
R2(config-router)#neighbor 1.1.1.1 route-reflector-client //配置 RR 客户端
R2(config-router)#neighbor 3.3.3.3 remote-as 100
```

```
R2(config-router)#neighbor 3.3.3.3 update-source Loopback0
R2(config-router)#neighbor 3.3.3.3 route-reflector-client
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router eigrp 1
R3(config-router)#network 3.3.3.0 255.255.255.0
R3(config-router)#network 23.23.23.0 255.255.255.0
R3(config-router)#no auto-summary
R3(config)#router bgp 100
R3(config-router)#no synchronization
R3(config-router)#no auto-summary
R3(config-router)#bgp router-id 3.3.3.3
R3(config-router)#neighbor 2.2.2.2 remote-as 100
R3(config-router)#neighbor 2.2.2.2 update-source Loopback0
R3(config-router)#neighbor 2.2.2.2 next-hop-self
R3(config-router)#neighbor 34.34.34.4 remote-as 200
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router bgp 200
R4(config-router)#no synchronization
R4(config-router)#no auto-summary
R4(config-router)#bgp router-id 4.4.4.4
R4(config-router)#neighbor 34.34.34.3 remote-as 100
R4(config-router)#network 4.4.4.0 mask 255.255.255.0
```

【说明】

当一个 AS 包含多个 IBGP 对等体时，路由反射器非常有用。因为 IBGP 客户只需要和路由反射器建立邻居关系，从而降低了 IBGP 的连接数量。路由反射器和它的客户合称为一个簇。路由反射器是克服 IBGP 水平分割的重要手段。

【技术要点】

RR 的反射规则如下：

（1）如果路由是从非客户的 IBGP 邻居学来的，则 RR 只将它反射给客户；

（2）如果路由是从客户学来的，RR 会将它反射给所有的非客户和客户（除了发起该路由的客户）；

（3）如果路由是从 EBGP 邻居学来的，RR 会将它反射给所有的非客户和客户。

4.实验调试

（1）show ip bgp neighbors

```
R2#show ip bgp neighbors 1.1.1.1
BGP neighbor is 1.1.1.1,  remote AS 100, internal link
  BGP version 4, remote router ID 1.1.1.1
  BGP state = Established, up for 00:31:06
  Last read 00:00:07, hold time is 180, keepalive interval is 60 seconds
......
```

```
 For address family: IPv4 Unicast
  BGP table version 4, neighbor version 4
  Index 1, Offset 0, Mask 0x2
  Route-Reflector Client
......
```

以上输出表明邻居 1.1.1.1 是路由反射器的客户端。

（2）show ip bgp

```
R2#show ip bgp 4.4.4.0
BGP routing table entry for 4.4.4.0/24, version 4
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  1.1.1.1
  200, (Received from a RR-client)
    3.3.3.3 (metric 2297856) from 3.3.3.3 (3.3.3.3)
      Origin IGP, metric 0, localpref 100, valid, internal, best
```

以上输出表明 BGP 路由条目“4.4.4.0/24”是从 RR 的客户端收到的，客户端是 3.3.3.3，并且将它反射给 1.1.1.1。注意上面输出中“(3.3.3.3)”指的是路由器 ID。

```
R1#show ip bgp 4.4.4.0
BGP routing table entry for 4.4.4.0/24, version 3
Paths: (1 available, best #1, table Default-IP-Routing-Table)
  Not advertised to any peer
  200
    3.3.3.3 (metric 2809856) from 2.2.2.2 (2.2.2.2)
      Origin IGP, metric 0, localpref 100, valid, internal, best
      Originator: 3.3.3.3, Cluster list: 2.2.2.2
```

以上输出表明在 AS 100 内 BGP 路由条目“4.4.4.0/24”的创造者是 3.3.3.3，簇 ID 是 2.2.2.2。

【术语】

① ORIGINATOR_ID：由路由反射器生成，是本 AS 内路由创造者的路由器 ID；

② CLUSTER_ID（簇 ID）：一个 AS 内的每个簇必须用一个唯一的 4 个字节的簇 ID 来标识，如果簇内只有一个 RR，那么簇 ID 就是 RR 的路由器 ID。当 RR 收到一个更新消息的时候，它检查 CLUSTER_LIST，如果发现在列表中有自己的簇 ID,就知道出现了路由环路。

24.6 实验 9：BGP 联邦配置

1.实验目的

通过本实验可以掌握

（1）BGP 联邦的含义

（2）BGP 联邦的配置

2.拓扑结构

实验拓扑如图 24-5 所示。

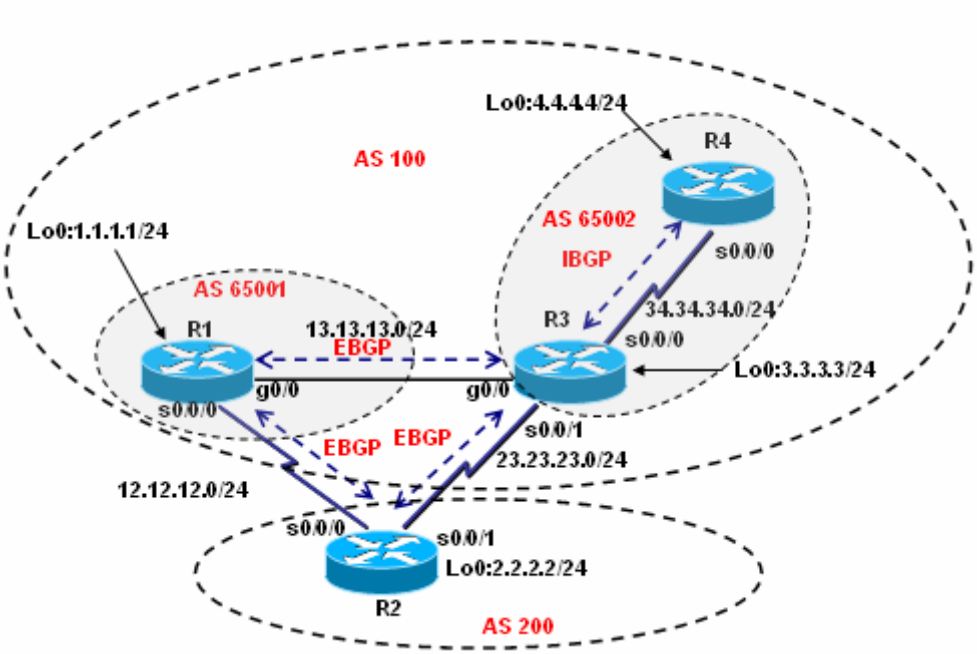

图 24-5 BGP 联邦配置

3. 实验步骤

本实验联邦的成员为 AS 65001 和 AS 65002，联邦对外 AS 为 100。

（1）步骤 1：配置路由器 R1

```
R1(config)#router bgp 65001
R1(config-router)#no synchronization
R1(config-router)#no auto-summary
R1(config-router)#bgp router-id 1.1.1.1
R1(config-router)#bgp confederation identifier 100 //配置联邦 ID
R1(config-router)#bgp confederation peers 65002 //联邦 EBGP 对等的成员
R1(config-router)#network 1.1.1.0 mask 255.255.255.0
R1(config-router)#neighbor 12.12.12.2 remote-as 200
R1(config-router)#neighbor 13.13.13.3 remote-as 65002
R1(config-router)#neighbor 13.13.13.3 next-hop-self
```

（2）步骤 2：配置路由器 R2

```
R2(config)#router bgp 200
R2(config-router)#no synchronization
R2(config-router)#no auto-summary
R2(config-router)#bgp router-id 2.2.2.2
R2(config-router)#network 2.2.2.0 mask 255.255.255.0
R2(config-router)#neighbor 12.12.12.1 remote-as 100
R2(config-router)#neighbor 23.23.23.3 remote-as 100
```

（3）步骤 3：配置路由器 R3

```
R3(config)#router eigrp 1
R3(config-router)#network 3.3.3.0 0.0.0.255
R3(config-router)#network 34.34.34.0 0.0.0.255
R3(config-router)#no auto-summary
R3(config)#router bgp 65002
```

```
R3(config-router)#no synchronization
R3(config-router)#no auto-summary
R3(config-router)#bgp router-id 3.3.3.3
R3(config-router)#bgp confederation identifier 100
R3(config-router)#bgp confederation peers 65001
R3(config-router)#network 3.3.3.0 mask 255.255.255.0
R3(config-router)#neighbor 4.4.4.4 remote-as 65002
R3(config-router)#neighbor 4.4.4.4 update-source Loopback0
R3(config-router)#neighbor 4.4.4.4 next-hop-self
R3(config-router)#neighbor 13.13.13.1 remote-as 65001
R3(config-router)#neighbor 13.13.13.1 next-hop-self
R3(config-router)#neighbor 23.23.23.2 remote-as 200
```

（4）步骤 4：配置路由器 R4

```
R4(config)#router eigrp 1
R4(config-router)#network 4.4.4.0 0.0.0.255
R4(config-router)#network 34.34.34.0 0.0.0.255
R4(config-router)#no auto-summary
R4(config)#router bgp 65002
R4(config-router)#no synchronization
R4(config-router)#no auto-summary
R4(config-router)#bgp router-id 4.4.4.4
R4(config-router)#network 4.4.4.0 mask 255.255.255.0
R4(config-router)#neighbor 3.3.3.3 remote-as 65002
R4(config-router)#neighbor 3.3.3.3 update-source Loopback0
```

【技术要点】

BGP 联邦用于将 AS 分割成多个子 AS,是控制大型 IBGP 对等的另一条途径。而子 AS 被称为成员自治系统。每个联邦都有被分配一个联邦 ID,对联邦外部来讲，这个联邦 ID 是代表整个联邦的 AS 号码。外部看不到联邦内部结构，联邦看起来就是一个 AS, 成员自治系统信息被隐藏起来。

4.实验调试

（1）在路由器 R2 上查看 BGP 表：

```
R2#show ip bgp
BGP table version is 5, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  1.1.1.0/24       23.23.23.3                             0 100 i
*>                  12.12.12.1               0             0 100 i
*> 2.2.2.0/24       0.0.0.0                  0         32768 i
*  3.3.3.0/24       12.12.12.1                             0 100 i
```

```
*>                  23.23.23.3               0             0 100 i
*  4.4.4.0/24       12.12.12.1                             0 100 i
*>                  23.23.23.3                             0 100 i
```

以上输出表明学到的“1.1.1.0/24”、“3.3.3.0/24”和“4.4.4.0/24 ”网络都有两条路径，而且都是来自 AS 100。由此看出 BGP 联邦内所有成员的信息对外都被隐藏。

（2）在路由器 R3 查看 BGP 表：

```
R3#show ip bgp
BGP table version is 6, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       13.13.13.1               0    100      0 (65001) i
*  2.2.2.0/24       13.13.13.1               0    100      0 (65001) 200 i
*>                  23.23.23.2               0             0 200 i
*> 3.3.3.0/24       0.0.0.0                  0         32768 i
r>i4.4.4.0/24       4.4.4.4                  0    100      0 i
```

以上输出表明，联邦内的 AS-PATH 用“()”表示。

【技术要点】

在联邦范围内，将成员 AS 加入到 AS-PATH 中，并且用括号扩起来，但是并不将它们公布到联邦的范围以外。AS-PATH 中联邦的 AS 号用于避免出现路由环路。

24.7 实验 10：BGP 团体配置

1.实验目的

通过本实验可以掌握

（1）BGP 团体的配置

（2）BGP 团体属性 local-AS

（3）BGP 团体属性 no-export

（4）BGP 团体属性 no-advertise

2.拓扑结构

实验拓扑如图 24-5 所示。

3.实验步骤与实验测试

保留**实验 9** 的所有配置，因本实验完全在**实验 9** 的配置的基础上完成。通过让路由器 R4 上的“4.4.4.0”携带不同的团体属性，来验证团体的各个属性的传递特征。对团体的讨论，我们仅仅讨论熟知的属性“local-AS”、“no-export”和“no-advertise”。本实验只给出在实验 9 基础上增加的配置。

（1）步骤 1：在路由器 R4 上配置团体属性 local-AS

```
R4(config)#ip prefix-list 1 permit 4.4.4.0/24 //定义前缀列表
R4(config)#route-map Local_AS permit 10 //定义 route-map
R4(config-route-map)#match ip address prefix-list 1 //匹配前缀列表
R4(config-route-map)#set community local-AS   //设置团体属性
```

R4(config-route-map)#**router bgp 65002**

R4(config-router)#**neighbor 3.3.3.3 send-community** //开启发送团体属性的能力

R4(config-router)#**neighbor 3.3.3.3 route-map Local_AS out**

//在出方向向邻居发送团体属性

（2）团体属性 local-AS 测试，分别在路由器 R3、R2 和 R1 上查看 BGP 表：

R3#**show ip bgp**

```
BGP table version is 6, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       13.13.13.1               0    100      0 (65001) i
*  2.2.2.0/24       13.13.13.1               0    100      0 (65001) 200 i
*>                  23.23.23.2               0             0 200 i
*> 3.3.3.0/24       0.0.0.0                  0         32768 i
r>i4.4.4.0/24       4.4.4.4                  0    100      0 i
```

R2#**show ip bgp**

```
BGP table version is 4, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  1.1.1.0/24       23.23.23.3                             0 100 i
*>                  12.12.12.1               0             0 100 i
*> 2.2.2.0/24       0.0.0.0                  0         32768 i
*> 3.3.3.0/24       23.23.23.3               0             0 100 i
```

R1#**show ip bgp**

```
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       0.0.0.0                  0         32768 i
*  2.2.2.0/24       13.13.13.3               0    100      0 (65002) 200 i
*>                  12.12.12.2               0             0 200 i
*> 3.3.3.0/24       13.13.13.3               0    100      0 (65002) i
```

以上输出表明携带团体“local-AS”属性的条目“4.4.4.0/24”只传递给路由器 R3，因为路由器 R3 和 R4 都在 AS 65002 内，并没有传递给路由器 R2 和 R1。由此可见“local-AS”团体属性只能在本 AS 内传递。

（3）步骤 2：在路由器 R4 上配置团体属性 no-export

R4(config)#**route-map NO-EXPORT permit 10**

```
R4(config-route-map)#match ip address prefix-list 1
R4(config-route-map)#set community no-export //设置团体属性
R4(config)#router bgp 65002
R4(config-router)#neighbor 3.3.3.3 send-community
R4(config-router)#neighbor 3.3.3.3 route-map NO-EXPORT out
//出方向向邻居发送团体属性
```

（4）团体属性 no-export 测试，分别在路由器 R3、R2 和 R1 上查看 BGP 表：

```
R3#show ip bgp
BGP table version is 6, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       13.13.13.1               0    100      0 (65001) i
*> 2.2.2.0/24       23.23.23.2               0             0 200 i
*> 3.3.3.0/24       0.0.0.0                  0         32768 i
r>i4.4.4.0/24       4.4.4.4                  0    100      0 i
R2#show ip bgp
BGP table version is 4, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  1.1.1.0/24       23.23.23.3                             0 100 i
*>                  12.12.12.1               0             0 100 i
*> 2.2.2.0/24       0.0.0.0                  0         32768 i
*  3.3.3.0/24       12.12.12.1                             0 100 i
*>                  23.23.23.3               0             0 100 i
R1#show ip bgp
BGP table version is 5, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       0.0.0.0                  0         32768 i
*  2.2.2.0/24       13.13.13.3               0    100      0 (65002) 200 i
*>                  12.12.12.2               0             0 200 i
*> 3.3.3.0/24       13.13.13.3               0    100      0 (65002) i
*> 4.4.4.0/24       13.13.13.3               0    100      0 (65002) i
```

以上输出表明携带团体“no-export”属性的条目“4.4.4.0/24”传递给路由器 R3 和 R1，因为路由器 R1、R3 和 R4 都在联邦 AS 100 内，并没有传递给路由器 R2。由此可见

“no-export”团体属性能在联邦的大 AS 内传递，如果没有联邦，只能在本 AS 内传递。

【提示】

通过命令“show ip bgp community no-export”来查看 BGP 表哪些条目携带相应的属性，如下所示：

```
R1#show ip bgp community no-export
BGP table version is 5, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 4.4.4.0/24       13.13.13.3               0    100      0 (65002) i
R3#show ip bgp community no-export
BGP table version is 6, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
r>i4.4.4.0/24       4.4.4.4                  0    100      0 i
```

以上输出表明路由器 R1 和 R3 的 BGP 表中的“4.4.4.0/24”携带了“no-export”团体属性。

（5）步骤 3：在路由器 R4 上配置团体属性 no-advertise

```
R4(config)#route-map NO_ADV permit 10
R4(config-route-map)#match ip address prefix-list 1
R4(config-route-map)#set community no-advertise
R4(config)#router bgp 65002
R4(config-router)#neighbor 3.3.3.3 send-community
R4(config-router)#neighbor 3.3.3.3 route-map NO_ADV out
```

（6）团体属性 no-advertise 测试，分别在路由器 R3、R2 和 R1 上查看 BGP 表：

```
R3#show ip bgp
BGP table version is 6, local router ID is 3.3.3.3
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       13.13.13.1               0    100      0 (65001) i
*  2.2.2.0/24       13.13.13.1               0    100      0 (65001) 200 i
*>                  23.23.23.2               0             0 200 i
*> 3.3.3.0/24       0.0.0.0                  0         32768 i
r>i4.4.4.0/24       4.4.4.4                  0    100      0 i
R2#show ip bgp
```

```
BGP table version is 4, local router ID is 2.2.2.2
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*  1.1.1.0/24       23.23.23.3                             0 100 i
*>                  12.12.12.1               0             0 100 i
*> 2.2.2.0/24       0.0.0.0                  0         32768 i
*  3.3.3.0/24       12.12.12.1                             0 100 i
*>                  23.23.23.3               0             0 100 i
R1#show ip bgp
BGP table version is 4, local router ID is 1.1.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 1.1.1.0/24       0.0.0.0                  0         32768 i
*  2.2.2.0/24       13.13.13.3               0    100      0 (65002) 200 i
*>                  12.12.12.2               0             0 200 i
*> 3.3.3.0/24       13.13.13.3               0    100      0 (65002) i
```

以上输出表明携带团体“no-advertise”属性的条目“4.4.4.0/24”只传递给路由器R3，并没有继续传递给路由器R2和R1。由此可见携带“no-advertise”团体属性的条目被收到后，将不通告给任何BGP对等体。

【提示】

可以为一条BGP路由设置多个团体属性。

24.8 BGP命令汇总

表24-1列出了本章涉及到的主要的命令。

表24-1 本章命令汇总

命令	作用
show tcp brief	查看TCP连接信息摘要
show ip bgp neighbors	查看邻居的TCP和BGP连接的详细信息
show ip bgp summary	查看BGP连接的摘要信息
show ip bgp	查看BGP表的信息
show ip bgp community	查看BGP团体属性
clear ip bgp *	重置BGP连接
router bgp	启动BGP进程
no synchronization	关闭同步
synchronization	打开同步

bgp router-id	配置 BGP 路由器 ID
neighbor *ip-address* remote-as	配置邻居路由器及所在的 AS
neighbor *ip-address* update-source	指定更新源
neighbor *ip-address* next-hop-self	配置下一跳自我
neighbor *ip-address* route-reflector-client	配置 RR 客户端
network	通告网络
aggregate-address	配置地址聚合
ip prefix-list	配置前缀列表
set origin egp	设置起源代码为 EGP
set as-path prepend	配置追加 AS-PATH
set local-preference	设置本地优先级属性值
bgp default local-preference	设置默认本地优先级属性值
bgp confederation identifier	配置联邦 ID
bgp confederation peers	配置联邦 EBGP 对等的成员
set community local-AS	设置团体属性
neighbor *ip-address* send-community	开启发送团体属性的能力

第 25 章 QOS

网络带宽的发展永远跟不上需求，因此当网络出现堵塞时如何保证网络的正常工作呢？QOS（服务质量）是一个解决方法，QOS 的基本思想就是把数据分类，放在不同的队列中。根据不同类数据的要求保证它的优先传输或者为它保证一定的带宽。QOS 是在网络发生堵塞才起作用的措施，因此 QOS 并不能代替带宽的升级。本章将介绍简单的 QOS 配置，实际上 Cisco 路由器现在推荐是模块化的 QOS 配置（MQC，Module QOS Config）。大量的 QOS 知识无法在本书中一一介绍。

25.1 QOS 简介

25.1.1 QOS

QOS 有三种模型：尽最大努力服务、综合服务、区分服务。尽最大努力服务实际上就是没有服务，先到的数据先转发。综合服务的典型就是预留资源，在通信之前所有的路由器先协商好，为该数据流预先保留带宽出来。区分服务是比较现实的模型，该服务包含了一系列分类工具和排队机制，为某些数据流提供比其他数据流优先级更高的服务。下面我们介绍典型的区分服务。

25.1.2 优先级队列

优先级队列（PQ，Priority Queue）中，有高、中、普通、低优先级四个队列。数据包根据事先的定义放在不同的队列中，路由器按照高、中、普通、低顺序服务，只有高优先级的队列为空后才为中优先级的队列服务，依次类推。这样能保证高优先级数据包一定是优先服务，然而如果高优先级队列长期不空，则低优先级的队列永远不会被服务。我们可以为每个队列设置一个长度，队列满后，数据包将被丢弃。

25.1.3 自定义队列

自定义队列（CQ，Custom Queue）和 PQ 不一样，在 CQ 中有 16 个队列。数据包根据事先的定义放在不同的队列中，路由器将为第一个队列服务一定包数量或者字节数的数据包后，就转为为第二个队列服务。我们可以定义不同队列中的深度，这样可以保证某个队列被服务的数据包数量较多，但不至于使得某个队列永远不会被服务。CQ 中的队列 0 比较特殊，只有队列 0 为空了，才能为其他队列服务。

25.1.4 加权公平队列

加权公平队列（WFQ，Weight Fair Queue）是低速链路（2.048M 以下）上的默认设置。WFQ 将数据包区分为不同的流，例如在 IP 中利用 IP 地址和端口号可以区分不同的 TCP 流或者 UDP 流。WFQ 为不同的流根据权重分配不同的带宽，权因子是 IP 数据包中的优先级字段。例如有 3 个流，两个流的优先级为 0，第三个为 5，总权为（1＋1＋6）=8，则前两个流每个得到带宽的 1/8，第三个流得到 6/8。

25.1.5 基于类的加权公平队列

基于类的加权公平队列（CBWFQ，Class Based Weight Fair Queue）允许用户自定义类别，并对这些类别的带宽进行控制。这在实际中很有用，例如我们可以控制我们的网络访问 Internet 时的 web 流量的带宽。可以根据数据包的协议类型、ACL、IP 优先级或者输入接口

等条件事先定义好流量的类型，为不同类别的流量配置最大带宽、占用接口带宽的百分比等。CBWFQ 可以和 NBAR、WRED 等一起使用。

25.1.6 低延迟队列

低延迟队列（LLQ，Low Latency Queue）的配置和 CBWFQ 很类似。有的数据包，例如 VOIP 的数据包，对数据的延迟非常敏感。LLQ 允许用户自定义数据类别，并优先让这些类别的数据传输，在这些数据没有传输完之前不会传输其他类别的数据。

25.1.7 加权随机早期检测

加权随机早期检测（WRED，Weight Random Early Detect）是 RED 的 Cisco 实现。当多个 TCP 连接在传输数据时，全部连接都按照最大能力传输数据，很快造成队列满，队列满后的全部数据被丢失；这时所有的发送者立即同时以最小能力传输数据，带宽开始空闲。接着全部发送者开始慢慢加大速度，于是又同时达到最大速率，又出现堵塞，如此反复。这样网络时空时堵，带宽的利用率不高。RED 则随机地丢弃 TCP 的数据包，保证链路的整体利用率。WRED 是对 RED 的改进，数据包根据 IP 优先级分成不同队列，每个队列有最小阀值、最大阀值，当平均长度小于最小阀值时，数据包不会被丢弃；随着平均队列的长度增加，丢弃的概率也增加；当平均长度大于最大阀值时，数据包按照设定的比例丢弃数据包。

25.1.7 CAR

承诺访问速率（CAR，Commited Access Rate）是一种流量策略的分类和标记的方法，它基于 IP 优先级、DSCP 值、MAC 地址或者访问控制列表来限制 IP 流量的速率。标记则可以改变 IP 优先级或者 DSCP。

CAR 使用令牌桶的机制，检查令牌桶中是否有足够的令牌。如果一个接口有可用的令牌，令牌可以从令牌桶中挪走，数据包被转发，当这个时间间隔过去后，令牌会重新添加到令牌桶中。如果接口没有可用的令牌，那么 CAR 可以定义对数据包采取的行为。CAR 使用 3 种速率定义来定义流量的速率：

- Normal rate（正常的速率）：令牌被添加到令牌桶中的平均速率，就是数据包的平均传输速率。
- Normal burst（正常的突发）：正常的突发时在时间间隔内允许正常流量速率的流量。
- Excess burst（过量突发）：超过正常突发的流量。当配置过量突发时，会借令牌并且将它添加到令牌桶中来允许某种程度的流量突发。当被借的令牌已经使用后在这个接口上收到的任何超出的流量会被扔掉。流量突发只会发生在短时间内，直到令牌桶中没有令牌存在才停止传输。

通常建议正常的流量速率配置为等于在一段时间内的平均流量速率。正常的突发速率应当等于正常速率的 1.5 倍。过量速率是正常突发速率的 2 倍。

25.1.8 基于网络的应用识别

基于网络的应用识别（NBAR，Network Based Application Recognition）实际上一个分类引擎，它查看数据包，对数据包包含的信息进行分析。NBAR 使得路由器不仅要做转发数据的工作，还要对数据包进行检查，这样会大大增加负载。NBAR 可以检查应用层的内容，例如可以检查 URL 是否有“.java”字样。NBAR 可以和许多 QOS 配合使用。

25.2 实验 1：PQ

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 PQ 的工作原理

（2） 掌握 PQ 的配置

2. 实验拓扑

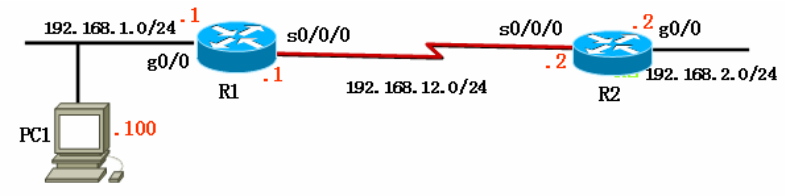

图 25-1 实验 1--实验 8 拓扑图

3. 实验步骤

（1） 步骤 1：配置 IP 地址、配置路由协议

（2） 步骤 2：配置 PQ

```
R1(config)#priority-list 1 protocol ip high tcp telnet
//创建 1 个优先级队列，标号为 1。把 telnet 流量放在高优先级队列中
R1(config)#priority-list 1 protocol ip high list 101
//以上把 ACL 101 定义的流量也放在高优先级队列中
R1(config)#priority-list 1 protocol ip medium gt 1000
//以上把数据包大小大于 1000 字节的流量放在中优先级队列中
R1(config)#priority-list 1 interface GigabitEthernet0/0 normal
//以上把从 g0/0 接口接收到流量放在普通优先级队列中
R1(config)#priority-list 1 default low
//以上把其他的流量放在低优先级队列中
R1(config)#access-list 101 permit ip host 10.1.1.1 any
//以上定义 ACL 101
R1(config)#priority-list 1 queue-limit 20 30 40 50
//以上定义优先级队列高、中、普通、低队列中的长度，如果队列超过这些长度，数据包将被丢弃。

R1(config)#int s0/0/0
R1(config-if)#priority-group 1
//以上把定义好的优先级队列应用在 s0/0/0 接口上
```

4. 实验调试

（1） 检查接口上的队列

```
R1#show interfaces s0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 192.168.12.1/24
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
```

```
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input 00:00:04, output 00:00:03, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: priority-list 1
```

//接口上的队列策略是优先级队列，标号为 1

（此处省略）

（2） 查看队列的配置

```
R1#show queueing priority
Current DLCI priority queue configuration:
Current priority queue configuration:
List   Queue  Args
1      low    default
1      high   protocol ip          tcp port telnet
1      high   protocol ip          list 101
1      medium protocol ip          gt 1000
1      normal interface GigabitEthernet0/0
```

（3） 测试队列是否生效

先从 PC1 ping R2 上的 192.168.2.2，然后：

```
R1#debug priority
```

从 PC1 ping R2 的 g0/0 接口，R1 上有信息，如下：

```
*Feb 28 02:59:57.299: PQ: Serial0/0/0 output (Pk size/Q 24/0)
*Feb 28 03:00:07.299: PQ: Serial0/0/0 output (Pk size/Q 24/0)
*Feb 28 03:00:08.679: PQ: Serial0/0/0: ip (defaulting) -> low
*Feb 28 03:00:08.679: PQ: Serial0/0/0 output (Pk size/Q 56/3)
*Feb 28 03:00:14.755: PQ: Serial0/0/0: cdp (defaulting) -> low
*Feb 28 03:00:14.755: PQ: Serial0/0/0 output (Pk size/Q 326/3)
*Feb 28 03:00:17.299: PQ: Serial0/0/0 output (Pk size/Q 24/0)
```

25.3 实验 2：CQ

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 CQ 的工作原理

（2） 掌握 CQ 的配置

2. 实验拓扑

如图 25-1。

3. 实验步骤

（1） 步骤 1：配置 IP 地址、配置路由协议

（2） 步骤 2：配置 CQ

R1(config)#queue-list 1 protocol ip 1 tcp telnet

//创建 1 个自定义队列，标号为 1。把 telnet 流量放在队列 1 中

R1(config)#queue-list 1 protocol ip 2 list 101

//以上把 ACL 101 定义的流量放在队列 2 中

R1(config)#queue-list 1 protocol ip 3 gt 1000

//以上把数据包大小大于 1000 字节的流量放在队列 3 中

R1(config)#queue-list 1 interface GigabitEthernet0/0 5

//以上把从 g0/0 接口接收到流量放在普通优先级队列 5 中

R1(config)#queue-list 1 default 6

//以上把其他的流量放在队列 6 中

R1(config)#access-list 101 permit ip host 10.1.1.1 any

//以上定义 ACL 101

R1(config)#queue-list 1 queue 1 limit 40

//以上定义队列 1 的深度为 40，也就是说路由器将为队列 1 服务 40 个数据包后，转向队列 2 的服务

R1(config)#queue-list 1 queue 2 limit 35

R1(config)#queue-list 1 queue 3 limit 30

R1(config)#queue-list 1 queue 5 limit 25

R1(config)#int s0/0/0

R1(config-if)#custom-queue-list 1

//以上把定义好的自定义队列应用在 s0/0/0 接口上

4. 实验调试

（1） 检查接口上的队列

R1#show interfaces s0/0/0

```
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 192.168.12.1/24
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set
  Keepalive set (10 sec)
  Last input 00:00:05, output 00:00:04, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: custom-list 1
  Output queues: (queue #: size/max/drops)
     0: 0/20/0 1: 0/40/0 2: 0/35/0 3: 0/30/0 4: 0/20/0
     5: 0/25/0 6: 0/20/0 7: 0/20/0 8: 0/20/0 9: 0/20/0
     10: 0/20/0 11: 0/20/0 12: 0/20/0 13: 0/20/0 14: 0/20/0
     15: 0/20/0 16: 0/20/0
```

//接口上的队列策略是自定义队列，标号为 1 ，可以看到每个队列的深度

（此处省略）

（2） 查看队列配置情况

```
R1#show queueing priority
Current custom queue configuration:
List   Queue  Args
1      6      default
1      1      protocol ip          tcp port telnet
1      2      protocol ip          list 101
1      3      protocol ip          gt 1000
1      5      interface GigabitEthernet0/0
1      1      limit 40
1      2      limit 35
1      3      limit 30
1      5      limit 25
```

（3） 测试队列是否生效

```
R1#debug custom-queue
```

25.4 实验 3：WFQ

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 WFQ 的工作原理

（2） 掌握 WFQ 的配置

2. 实验拓扑

如图 25-1。

3. 实验步骤

（1） 步骤 1：配置 IP 地址、配置路由协议

（2） 步骤 2：配置 WFQ

```
R1(config)#int s0/0/0
R1(config-if)# fair-queue 512 1024 10
```

//以上是在接口上启用 WFQ，实际上在 E1 速（2.048M）或者更低速率的链路上，WFQ 是默认启用的。512 是丢弃值，当队列达到 512 数据包时，数据将被丢弃；1024 是最大的会话数；10 是 RSVP 可预留队列。

4. 实验调试

```
R1#show interfaces s0/0/0
Serial0/0/0 is up, line protocol is up
  Hardware is GT96K Serial
  Internet address is 192.168.12.1/24
  MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
```

```
Encapsulation HDLC, loopback not set
Keepalive set (10 sec)
Last input 00:00:09, output 00:00:08, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/512/0 (size/max total/threshold/drops)
   Conversations  0/0/1024 (active/max active/max total)
   Reserved Conversations 0/0 (allocated/max allocated)
   Available Bandwidth 96 kilobits/sec
```

25.5 实验 4：CBWFQ

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 CBWFQ 的工作原理

（2） 掌握 CBWFQ 的配置

2. 实验拓扑

如图 25-1。

3. 实验步骤

（1） 步骤 1：配置 IP 地址、配置路由协议

（2） 步骤 2：定义 class-map

```
R1(config)#class-map match-any CLASS-MAP1
```

//以上定义了一个 class-map，名为 CLASS-MAP1，class-map 命令参见下文解释。

```
R1(config-cmap)#match protocol http
R1(config-cmap)#match protocol ftp
```

//以上定义只要是 http 或者 ftp 流量就属于 CLASS-MAP1

```
R1(config)#class-map match-all CLASS-MAP2
R1(config-cmap)#match protocol telnet
```

//以上定义只要是 telnet 流量就属于 CLASS-MAP1。系统有一个默认的 class-map，名为 class-default，凡是没有定义的流量就属于这个 class-map。

【技术要点】class-map 命令格式为：“**class-map [match-all | match-any] name**”：

- match-all：指明下面的条件必须全部满足，才可以执行，此为默认值；
- match-any：表示匹配任何一个条件就可以执行。

在 class-map 模式下，可以设置各种匹配条件，例如：

- 匹配一种协议类型：**match protocol** ***protocol-name*** 。协议类型包括 EGP，ICMP，EIGRP，DNS，HTTP，Telnet 等上百种具体协议。
- 匹配访问列表：**match access-group** { ***number*** | **name** ***acl_name*** } 。可以匹配基于号码的 list 和基于 Name 的 Access list。
- 匹配 CoS (class of Servie)：**match cos** ***cos-value*** 。匹配 IP 包中的 CoS 值。

- 匹配 IP 优先级（IP Precedence）：**match ip precedence** *precedence-value* 。匹配 IP 包中的 IP 优先级值。
- 匹配 DSCP 值（Differentiated Services Code Point）：**match ip dscp** *dscp_value* 。匹配 IP 包中的 DSCP 值。
- 匹配入接口：**match input-interface** *type number*。匹配 IP 包的进入接口。

（3） 步骤 3：定义 Policy-map

R1(config)#**policy-map MY-POLICY**

//以上是定义 policy-map。

（4） 步骤 4：配置带宽

R1(config-pmap)#**class CLASS-MAP1**

R1(config-pmap-c)#**bandwidth 60**

R1(config-pmap)#**class CLASS-MAP2**

R1(config-pmap-c)#**bandwidth 10**

//以上配置 CLASS-MAP1 流量的带宽为 60K，CLASS-MAP2 流量的带宽为 10K。该接口的总带宽为 128K。该格式为：“**bandwidth** { *bandwidth_value* | **percent** *percent_value* } 。

- 可以指定具体带宽：单位为 K。
- 也可以指明百分比：percent 关键字指定接口可用带宽百分比，可以 0--100 取值，默认情况下接口可用最大带宽为物理带宽的 75％（其余 25％留给系统自己用），所以 percent 值是这 75%的 percent，而不是物理带宽的 percent，我们可以在接口下使用“**max-reserved-bandwidth precent**”命令更改最大可用带宽。

（5） 步骤 5：将 policy-map 应用到接口上

R1(config)#**int s0/0/0**

R1(config-if)#**service-policy output MY-POLICY**

//以上把我们定义的策略应用在接口的 output 方向上，CBWFQ 只能在 output 方向。这样我们就在接口上限制了 http、ftp 和 telnet 流量的带宽。

4. 实验调试

（1） 检查 class-map 和 policy-map

R1#**show interfaces s0/0/0**

R1#**show class-map**

```
 Class Map match-all CLASS-MAP2 (id 2)
   Match protocol telnet
 Class Map match-any CLASS-MAP1 (id 1)
   Match protocol http
   Match protocol telnet
 Class Map match-any class-default (id 0)
   Match any
```

R1#**show policy-map**

```
  Policy Map MY-POLICY
    Class CLASS-MAP1
      Bandwidth 60 (kbps) Max Threshold 64 (packets)
    Class CLASS-MAP2
      Bandwidth 10 (kbps) Max Threshold 64 (packets)
```

（2） 检查策略在接口上的运用情况

R1#show policy-map interface s0/0/0

25.6 实验 5：LLQ

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 LLQ 的工作原理

（2） 掌握 LLQ 的配置

2. 实验拓扑

如图 25-1。

3. 实验步骤

在实验 4 的基础上继续本实验。

（1） 步骤 1：定义 class-map3，把 IP 优先级为 critical 的 IP 流量包含进来

R1(config)#class-map match-any CLASS-MAP3

R1(config-cmap)#match ip precedence critical

（2） 步骤 2：配置 LLQ

R1(config)#policy-map MY-POLICY

R1(config-pmap)#class CLASS-MAP3

R1(config-pmap-c)#priority 15

//LLQ 的配置和 CQWFQ 配置很类似，不过使用了 priority 命令，我们这里限制它的带宽为 15k，超过这个带宽的数据包将被丢弃。这样 CLASS-MAP3 的流量将优先被发送，然后才发送 CLASS-MAP1 和 CLASS-MAP2 等流量。

4. 实验调试

（1） 检查 policy-map

R1#show policy-map

```
  Policy Map MY-POLICY1
    Class CLASS-MAP1
  Policy Map MY-POLICY
    Class CLASS-MAP1
      Bandwidth 60 (kbps) Max Threshold 64 (packets)
    Class CLASS-MAP2
      Bandwidth 10 (kbps) Max Threshold 64 (packets)
    Class CLASS-MAP3
      Strict Priority
      Bandwidth 15 (kbps) Burst 375 (Bytes)
```

（2） 检查策略在接口上的应用情况

R1#show policy-map interface s0/0/0

```
R1#show policy-map interface s0/0/0
 Serial0/0/0
```

（此处省略）

```
  Class-map: CLASS-MAP3 (match-all)
    0 packets, 0 bytes
    5 minute offered rate 0 bps, drop rate 0 bps
    Match: ip precedence 5
    Queueing
      Strict Priority
      Output Queue: Conversation 40
      Bandwidth 15 (kbps) Burst 375 (Bytes)
      (pkts matched/bytes matched) 0/0
      (total drops/bytes drops) 0/0
```

（此处省略）

25.7 实验 6：WRED

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 WRED 的工作原理

（2） 掌握 WRED 的配置

2. 实验拓扑

如图 25-1。

3. 实验步骤

（1） 步骤 1：配置 IP 地址、路由协议

（2） 步骤 2：配置 WRED

```
R1(config)#int s0/0/0
R1(config-if)#random-detect
//以上在接口上启用 WRED
R1(config-if)#random-detect precedence 0 18 42 12
```

//以上配置 IP 优先级为 0 的队列，最低阀值为 18，平均队列长度小于 18 时，数据包不会被丢弃；当平均队列长度大于 18 时，开始丢弃数据包，平均队列长度越大，丢弃的数据包越多；最大阀值为 42，平均队列长度小于 42 时，数据包按照 1/12 的比例丢弃。

4. 实验调试

```
R1#show queueing random-detect
Current random-detect configuration:
  Serial0/0/0
    Queueing strategy: random early detection (WRED)
    Random-detect not active on the dialer
    Exp-weight-constant: 9 (1/512)
    Mean queue depth: 0
  class          Random drop      Tail drop    Minimum Maximum  Mark
```

	pkts/bytes	pkts/bytes	thresh	thresh	prob
0	**0/0**	**0/0**	**18**	**42**	**1/12**
1	0/0	0/0	22	40	1/10
2	0/0	0/0	24	40	1/10
3	0/0	0/0	26	40	1/10
4	0/0	0/0	28	40	1/10
5	0/0	0/0	31	40	1/10
6	0/0	0/0	33	40	1/10
7	0/0	0/0	35	40	1/10
rsvp	0/0	0/0	37	40	1/10

//以上显示 WRED 的配置情况，默认时不同 IP 优先级的队列的最低有所不同，我们更改了 IP 优先级为 0 的队列。

25.8 实验 7: CAR

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 CAR 的工作原理

（2） 掌握 CAR 的配置

2. 实验拓扑

如图 25-1。

3. 实验步骤

（1） 步骤 1：配置 IP 地址、路由协议

（2） 步骤 2：配置 WRED

```
R1(config)#int s0/0/0
R1(config-if)#rate-limit output access-group 101 64000 12000 16000 conform-action set-prec-transmit 3 exceed-action set-prec-transmit 0
```

//以上在接口上启用 CAR，对于符合 ACL 101 的流量，平均速率为 64000 位/秒，正常突发量为 12000 字节/秒，过量突发量为 12000 字节/秒。

```
R1(config-if)#rate-limit output access-group 102 16000 8000 9000 conform-action set-prec-transmit 2 exceed-action drop
R1(config-if)#rate-limit output 48000 8000 10000 conform-action set-prec-transmit 0 exceed-action drop
R1(config)#access-list 101 permit tcp any any eq www
R1(config)#access-list 102 permit tcp any any eq smtp
```

【技术要点】rate-limit 的命令格式为：

rate-limit { output | input } { CIR BC BE } conform-action { action } exceed-action { action }

- CIR 单位是 bit/s；而 BC 和 BE 的单位是 byte/s。
- conform-action 的条件是指当要发的数据小于正常突发(bc)的时候

- exceed-action 是指要发的数据大于普通突发，小于最大突发(be)的时候。
- action 的选项共有如下这些：

  continue：继续执行下一条 CAR 语句

  drop：丢弃数据包

  tranmsit：转发数据包

  set-prec-continue { precedence }：设置 IP 优先级并继续执行下一条 CAR 语句

  set-prec-transmit { precedence }：设置 IP 优先级并转发数据包

  set-dscp-continue { dscp }：设置 dscp 值并继续执行下一条 CAR 语句

  set-dscp-transmit { dscp }：设置 dscp 值并转发数据包

4. 实验调试

```
R1#show interfaces rate-limit
Serial1/1
  Output
    matches: access-group 101
      params:  64000 bps, 12000 limit, 16000 extended limit
      conformed 0 packets, 0 bytes; action: set-prec-transmit 3
      exceeded 0 packets, 0 bytes; action: set-prec-transmit 0
      last packet: 9703244ms ago, current burst: 0 bytes
      last cleared 00:03:49 ago, conformed 0 bps, exceeded 0 bps
    matches: access-group 102
      params:  16000 bps, 8000 limit, 9000 extended limit
      conformed 0 packets, 0 bytes; action: set-prec-transmit 2
      exceeded 0 packets, 0 bytes; action: drop
      last packet: 9703256ms ago, current burst: 0 bytes
      last cleared 00:03:41 ago, conformed 0 bps, exceeded 0 bps
    matches: all traffic
      params:  48000 bps, 8000 limit, 10000 extended limit
      conformed 0 packets, 0 bytes; action: set-prec-transmit 0
      exceeded 0 packets, 0 bytes; action: drop
      last packet: 9703272ms ago, current burst: 0 bytes
      last cleared 00:03:33 ago, conformed 0 bps, exceeded 0 bps
```

25.9 实验 8：NBAR

1. 实验目的

通过本实验，读者可以掌握如下技能：

（1） 理解 NBAR 的工作原理

（2） 掌握 NBAR 的配置

2. 实验拓扑

如图 25-1。

3. 实验步骤

NABR 的配置和 CBWFQ 没什么差别，因为 NBAR 实际上只是一个分类技术。我们这里将利用 NBAR 来禁止 BT 和 edonkey 下载。如下：

```
R1(config)#class-map match-any BT
R1(config-cmap)#match protocol bittorrent
R1(config-cmap)#match protocol edonkey
//定义流量，匹配 bittorrent 和 edonkey
R1(config)#policy-map DENY-BT
R1(config-pmap)#class BT
R1(config-pmap-c)#drop
//定义策略，匹配 bittorrent 和 edonkey 的流量被丢弃
R1(config)#int s0/0/0
R1(config-if)#service-policy output DNEY-BT
```

【提示】在旧的 IOS 中，class-map 模式下不能使用“**match protocol bittorrent**”等命令，要先从 Cisco 网站下载 **bittorrent.pdlm** 等文件，上传到路由器上的 FLASH 中，并使用命令“**ip nbar pdlm flash: bittorrent.pdlm**”后，才能在 class-map 模式下，使用“**match protocol bittorrent**”命令。

【提示】NBAR 需要路由器启用 CEF,默认时 CEF 是开启的,如果没有开启,可以使用“**ip cef**”命令。

25.10 本章小结

本章介绍了 QOS 的目的和基本工作原理，QOS 的各种概念显得杂乱无章。QOS 有各种拥塞避免技术：FIFO、PQ、CQ、WFQ 和 CBWFQ。它们的共同特点就是把数据流进行分类，放入不同的队列中，不同的队列有不同的处理方式。本章一一介绍以上这些技术的配置。CAR 和 NBAR 是高级的 QOS 应用，可以用来限速，甚至禁止 BT 下载等。表 25-1 是本章的命令汇总。

表 25-1 本章命令汇总

命令	作用
priority-list 1 protocol ip high tcp telnet	创建优先级队列，标号为 1。把 telnet 流量放在高优先级队列中
priority-list 1 queue-limit 20 30 40 50	定义优先级队列高、中、普通、低队列中的长度
priority-group 1	把定义好的优先级队列应用接口上
show queueing priority	查看优先级队列情况
debug priority	调试优先级队列
queue-list 1 protocol ip 1 tcp telnet	创建自定义队列，标号为 1。把 telnet 流量放在队列 1 中
queue-list 1 queue 1 limit 40	定义队列 1 的深度为 40，
custom-queue-list 1	把定义好的自定义队列应用接口上
fair-queue 512 1024 10	在接口上启用 WFQ，512 是丢弃值，1024 是最大的会话数，10 是 RSVP 可预留队列
class-map match-any CLASS-MAP1	定义 class-map，名为 CLASS-MAP1
match protocol http	匹配 http 协议

bandwidth 10	配置 CLASS-MAP 流量的带宽为 60K
service-policy output MY-POLICY	把定义好的策略应用在接口的 output 方向上
show class-map	显示 class-map 信息
show policy-map	显示 policy-map 信息
show policy-map interface s0/0/0	显示接口 s0/0/0 上的 policy-map 配置
priority 15	配置 LLQ，带宽为 15k
random-detect	在接口上启用 WRED
random-detect precedence 0 18 42 12	配置 WRED，对于 IP 优先级为 0 的队列，最低阀值为 18，最大阀值为 42，按照 1/12 的最大比例丢弃数据包
show queueing random-detect	显示 WRED 的配置情况
rate-limit output access-group 101 64000 12000 16000 conform-action set-prec-transmit 3 exceed-action set-prec-transmit 0	在接口上启用 CAR，限制符合 ACL 101 的流量
show interfaces rate-limit	显示各接口上 CAR 的情况
drop	丢弃数据包