

CCNA 实验手册

Version 5.0(1)

SPOTO Global

2011-9-30

密级 ☒开放 ☐内部 ☐机密
类型 ☐讨论版 ☐测试版 ☒正式版

修订记录				
修订日期	修订人	版本号	审核人	修订说明
2011-9-30	Linda	5.0(1)	JackVins.	

分发列表	
发件	SPOTO 标准化小组
收件	SPOTO 全体伙伴
发布日期	
生效日期	

版权声明

思博（全球）2009 年版权所有，保留一切权利
非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文的部分或全部，并不得以任何形式传播。

Copyright©2009 SPOTO(GLOBAL) All Right Reserved.
No part of this document may be reproduced or transmitted in any form or by any means
without prior written consent of SPOTO.

目 录

引 言	1
1 SPOTO 实验环境概述	2
1.1 实验环境拓扑 (RACK)	2
1.2 实验环境操作	4
2 路由基础实验	7
2.1 实验 1-设备基本操作	7
2.2 实验 2-静态路由	12
2.3 实验 3-RIP 协议	18
2.4 实验 4-EIGRP 协议	24
2.5 实验 5-OSPF 协议	28
3 交换基础实验	33
3.1 实验 1-VLAN/TRUNK/VTP/管理	33
3.2 实验 2-STP 配置*	40
4 路由策略实验	43
4.1 实验 1-ACL 实验	43
4.2 实验 2-NAT 实验	48
5 广域网和 IPv6 实验	53
5.1 实验 1-HDLC/Frame-Relay	53
5.2 实验 2-PPP 实验	58
5.3 实验 3-IPv6 实验*	62
6 综合实验	67
6.1 中小型企业组网案例一	67
6.2 中小型企业组网案例二	69

引言

CCNA (Cisco Certified Network Associate) 是 Cisco 职业认证体系中的初级认证，也是任何一名网络初学者进入思科网络殿堂的第一个台阶。良好的开始是成功的一半。充分掌握 CCNA 课程中的理论知识，并在工作环境中熟练的应用，能够使我们 IT 行业技术的理解从朦胧变得清晰。

SPOTO CCNA 640-802 实验手册紧跟 Cisco 最新认证教材，它是由 SPOTO TS 团队为广大学员精心设计的一本网络试验指导书册，它以 CCNA 知识点为基础，经过缜密的思考，大量的实践，多轮讨论，最终撰写完成。

这份手册，旨在为学员在 CCNA 实验过程中提供参考，加深对 CCNA 课程理论的理解，并能更快更熟练的应用网络技术，所以，我们不建议学员按照本手册的内容进行单纯的模仿，而是在充分理解相关知识点的前提下，参考本手册进行实验。

CCNA 实验指导手册目前发布的是 Version5.0，让我们再次感谢 Version1.0 的编写者潘星，Version 2.0 的编写者黄柳荫，Version 3.0 的编写者吴志翔、彭索、黄柳荫，Version4.0 的编写者王晓强、林燕清、林燕为 SPOTO CCNA 试验体系的完善所做的积累。

CCNA 实验指导手册 V5.0 是 SPOTO TS 团队共同努力的结晶，并最终由 TS 林燕、TS 朱仕耿以及 TS 王晓强等人进行后期整理、排版工作。在此特别感谢 SPOTO 创始人胡晓明校长 (CCIE#9748) 和 SPOTO 所有伙伴在我们编写本手册过程中所给予的宝贵的意见和大力的支持。

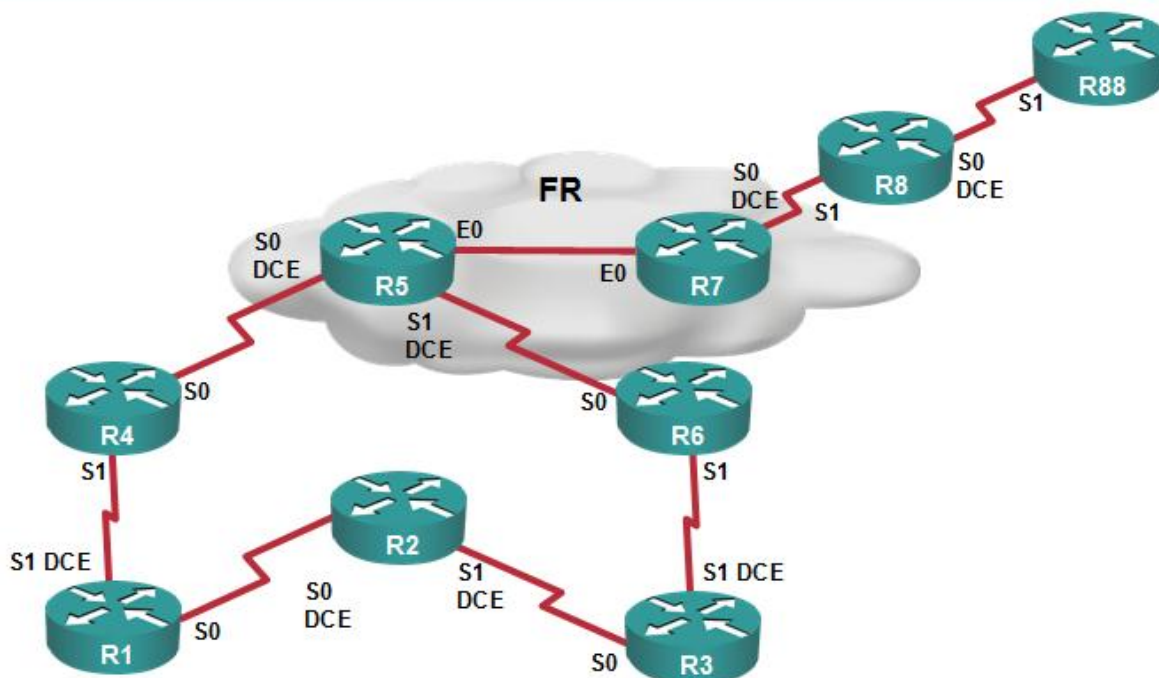
欢迎大家积极反馈意见，以便持续更新。

1 SPOTO 实验环境概述

1.1 实验环境拓扑 (RACK)

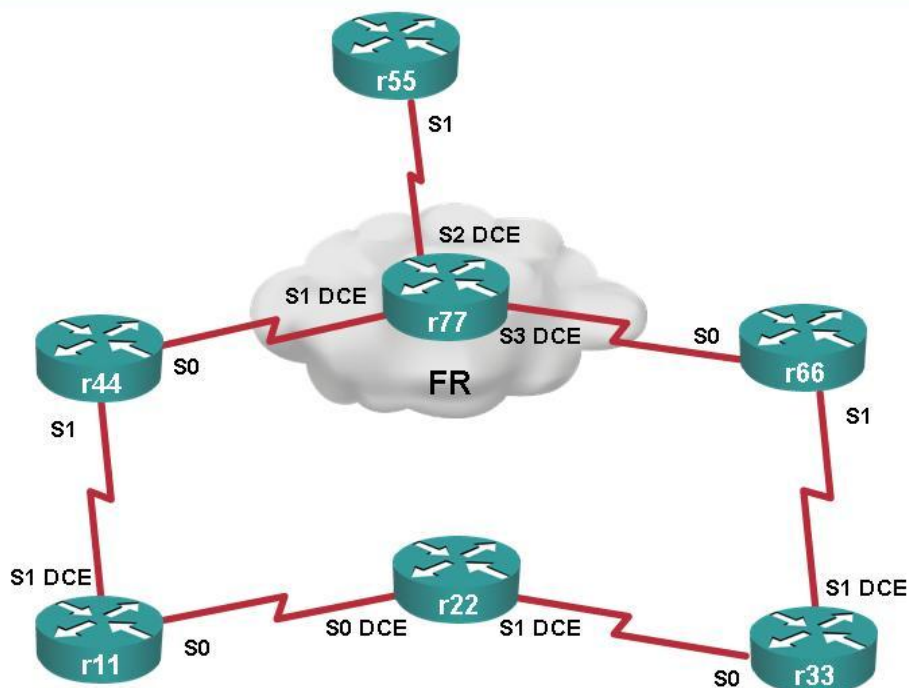
R1~R8 以及 R88 拓扑

路由机架 R1 – R8 及 R88 拓扑

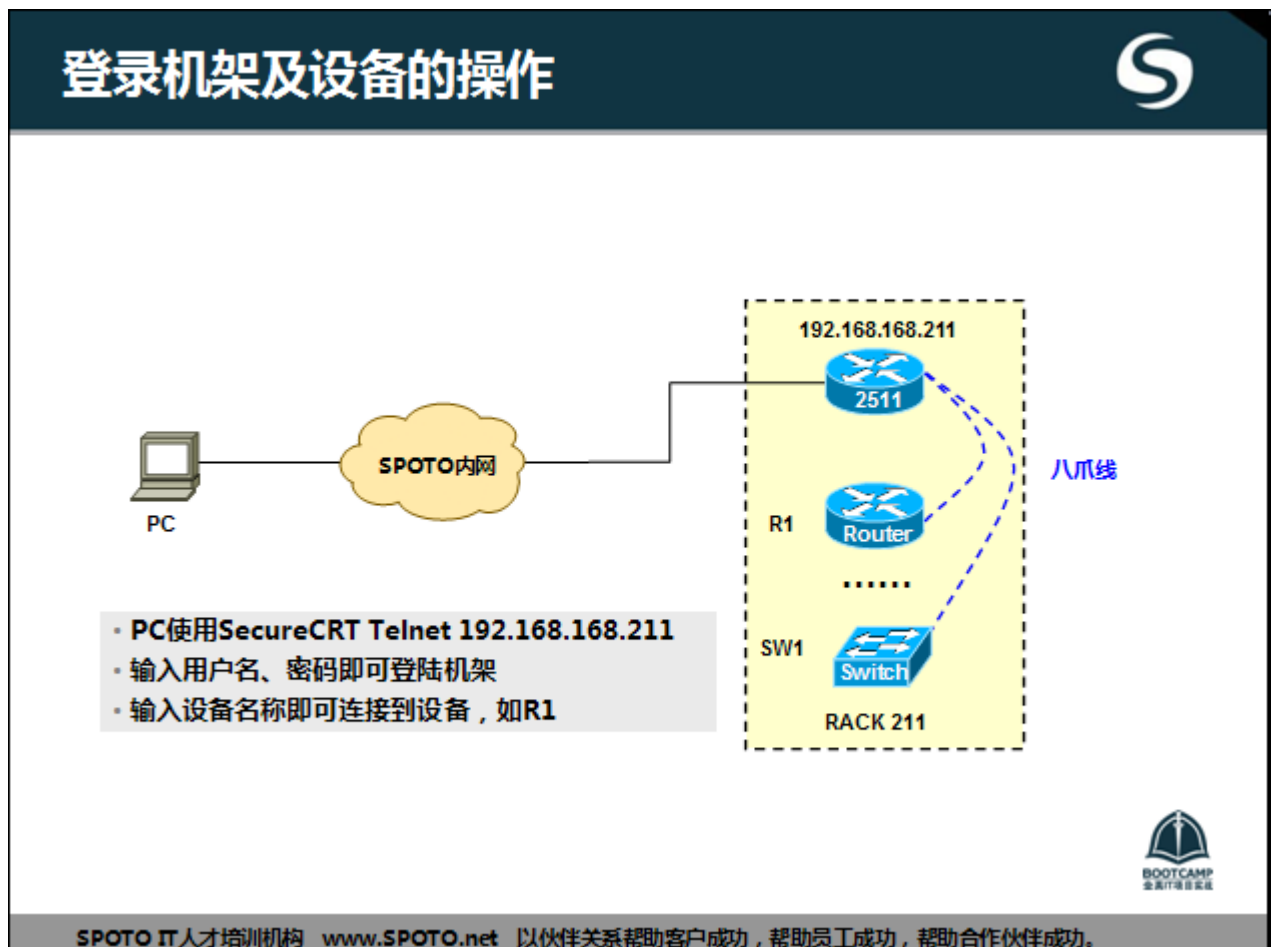


R11~R77 拓扑

路由机架 R11 – R77 拓扑



1.2 实验环境操作



1. 机架地址：

上海机架 1：192.168.168.111

上海机架 2：192.168.168.112

2. 登陆方式：

1) CMD 窗口登录

PC 机 “开始” -> “运行” -> “cmd” 进入 CMD 界面

键入 telnet 192.168.168.111 （ RACK1 的地址 ）即可

2) 使用 SecureCRT

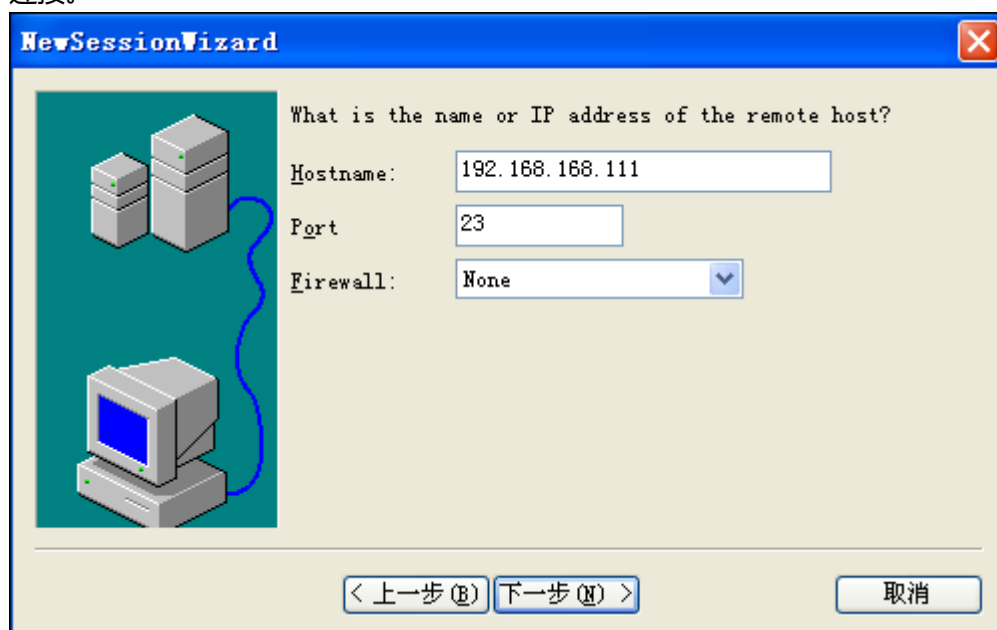
双击桌面上 SecureCRT 的图标



SecureCRT

建立一个会话连接，协议使用 telnet，hostname 为 192.168.168.111，端口号位 23 端口，点击

连接。



3. 机架的相关操作命令

1) show line

看到如图的 16 条 TTY 线路即为我们要登录的 16 台路由器，我们做实验只要看这 16 条 TTY 线路即可根据第一页中的拓扑图来看

TTY : 1 ---- 8 表示 R1 --- R8

TTY : 9 ---- 16 表示 R11 --- R88

```
Rack01_2511>show line
Tty Typ Tx/Rx
0 CTY
* 1 TTY 9600/9600
* 2 TTY 9600/9600
3 TTY 9600/9600
4 TTY 9600/9600
5 TTY 9600/9600
6 TTY 9600/9600
7 TTY 9600/9600
8 TTY 9600/9600
* 9 TTY 9600/9600
10 TTY 9600/9600
11 TTY 9600/9600
12 TTY 9600/9600
13 TTY 9600/9600
14 TTY 9600/9600
15 TTY 9600/9600
16 TTY 9600/9600
17 HW 9600/9600
```

2) 进入操作界面

例如我们要进入到第一条线路，我们就在终端输入 r1，这时候会出现如下提示：是否要进入入一些初始化配置日志的界面？输入 N（如果输入 Y，则会进入初始界面，要求你配置一些由系统要求你配置的东西）。


```
!!!!!!!!!!!!!!!!!!!!!!!!!!!! welcom to SPOTO_LAB !!!!!!!!!!!!!!!!!!!!!!!!!!!!!
% Please answer 'yes' or 'no'.
Would you like to enter the initial configuration dialog? [yes/no]:
% Please answer 'yes' or 'no'.
Would you like to enter the initial configuration dialog? [yes/no]:
```

```
Rack01_2511>r1
Trying r1 (1.1.1.1, 2001)... Open
!!!!!!!!!!!!!!!!!!!!!!!!!!!! Welcom t
Router>
```

注：有可能此路由器被别人占用了，所以你登录不了，需要查看是否被别人占用使用命令 `show sessions` (或是用 `who` 查看)，比如左图的 TTY 1、2 和 9 前面有一个“*”符号，表示有人登陆这两条线路所在的路由器了，所以我们就不能登录了，我们只能登陆 TTY 3 – 8 和 TTY 9—16 这些线路所在的路由器了。

另：想 T 掉 r1 设备上的用户，从“从机架登陆路由器方式”这一步就可以知道 r1 所在的 TTY 线路编号为 1，所以需要输入“`clear line 1`”将当前登陆 r1 的用户 T 掉。

```
Rack01_2511>show sessions
Conn Host          Address
  1 r1              1.1.1.1
* 2 r2              1.1.1.1
```

```
Rack01_2511>w
Conn Host          Address
  1 r1              1.1.1.1
* 2 r2              1.1.1.1
```

```
Rack01_2511>r1
Trying r1 (1.1.1.1, 2001)...
% Connection refused by remote host

Rack01_2511>l
[Resuming connection 1 to r1 ... ]

Router>
```

```
Rack01_2511>clear line 1
[confirm]
[OK]
Rack01_2511>
```

从路由器退回到从机架登陆路由器方式：(此处以 SecureCRT 为例，CMD 的操作完全一样)
若想从路由器退回到 2511 终端继续登陆其他路由器的话只要在当前路由器进程中同时按住“**Ctrl+Shift+6**”然后松开按再按一下“**x**”键。

2 路由基础实验

2.1 实验 1-设备基本操作

1. 实验目的

- 1) 掌握路由器命令行下的基本操作
- 2) 掌握路由器的基本配置命令
- 3) 路由器的基本控制台输出信息抓取及识别

2. 拓扑及需求



3. 配置及实现

- 1) 主机通过 console 口登录路由器
- 2) 设置路由器主机名，设置系统时间

```
Router>enable
Router#configure terminal
Router(config)#

// 路由器三模式的转换：“Router>”表示用户模式，“Router#”表示特权模式，“Router(config)#”表示全局配置模式

Router(config)#hostname R1
// 设置路由器主机名为 R1

Router#clock set hh:mm:ss day month year
// 修改当前设置的时间，如：clock set 12:15:30 5 jun 2011

Router#config terminal
Router(config)#clock timezone GMT +8
// 修改当前系统的时区为 GMT +8 GMT 全称 Greenwich Mean Time
```

- 3) 路由器常用的三条命令：关闭域名解析，配置 console 永不超时，配置 console 的日志同步

```
Router(config)#line console 0
Router(config-line)#no exec-time
```

```
或
Router(config-line)#exec-time 0 0
// 配置 console 永不超时 ( 常用于实验环境 )
Router(config-line)#logging synchronous
// 配置 console 的日志同步
Router(config-line)#exit
// 退出控制台接口模式
Router(config)#no ip domain lookup
// 关闭域名解析
```

4) 创建 MOTD : Welcome to SPOTO !

```
Router(config)#banner motd #Welcome to SPOTO#
// 配置执行标志区, 即为登录路由器时显示的信息, 需要一个起始字符和相同的结束字符, 如
使用: #, 此时为 Welcome to SPOTO
```

5) 设置和取消密码: console 密码、特权模式密码、VTY 线路密码

```
Router(config)#line console 0
Router(config-line)#password word
// 进入 console 口, 配置 console 口密码, word 为自己指定的密码, 例如: password spoto
Router(config-line)#login
// 使密码在 console 线路生效
Router(config-line)#no password word
// 删除密码
Router(config-line)#no login
// 取消密码, 使不需要密码设置就能进入 console 口
Router(config-line)#exit
Router(config)#enable password word
// 设置 enable 明文密码, 可通过 show run 查看密码
Router(config)#no enable password
// 取消 enable 明文密码
Router(config)#enable secret word
// 设置 enable 密文密码, show run 只能查看到字母、数字等的组合, 安全级别高于明文密码,
与明文同时设置时, 密文密码生效
Router(config)#no enable secret
// 取消 enable 密文密码
Router(config)#line vty 0 4
```

```
// 进入 VTY 0-4 线路
Router(config-line)#password word
// 设置 VTY 线路密码，即用户通过 telnet 登入路由器时输入的密码
Router(config-line)#login
// 使密码在线路上生效
Router(config-line)#no password
Router(config-line)#no login
// 取消 VTY 线路密码，使不需要密码就能远程登录
Router(config-line)#exit
```

6) 配置路由器接口 IP

两台路由器的基础配置

R1 :

```
Router(config)#hostname R1
R1 (config)#no ip domain lookup
R1(config)#line console 0
R1(config-line)#no exec-timeout
R1(config-line)#logging synchronous
R1(config-line)#exit
```

R2 同理，此处不再赘述。

进入接口,配置 IP

```
Route(config)#interface type
// 接口类型: serial/ethernet/fastethernet  solt/num
Route(config-if)#ip address ip mask
// 为接口配上 IP 地址
```

R1 :

```
R1(config)#interface fastethernet 0/0
R1(config-if)#ip address 12.12.12.1 255.255.255.252
R1(config-if)#no shutdown
```

R2 :

```
R2(config)#interface fastethernet 0/0
R2(config-if)#ip address 12.12.12.2 255.255.255.252
R2(config-if)#no shutdown
```

7) 查看当前设备接口状态

```
R1#show ip interface brief
```

Interface	IP-Address	OK? Method Status	Protocol
FastEthernet0/0	12.12.12.1	YES manual up	up
R2#show ip interface brief			
Interface	IP-Address	OK? Method Status	Protocol
FastEthernet0/0	12.12.12.2	YES manual up	up

8) 互 ping 测试

R1#ping 12.12.12.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 12.12.12.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 8/62/120 ms

R2#ping 12.12.12.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 12.12.12.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 8/41/100 ms

9) telnet 到邻居路由器

分别在 R1、R2 上完成 TELNET 密码的设置

```
R1(config)#line vty 0 4
```

```
R1(config-line)#password spoto
```

```
R1(config-line)#login
```

```
R2(config)#line vty 0 4
```

```
R2(config-line)#password spoto
```

```
R2(config-line)#login
```

R1 telnet 到 R2

```
R1#telnet 12.12.12.2
```

```
Trying 12.12.12.2 ... Open
```

```
User Access Verification
```

```
Password:
```

```
R2>
```

// 密码输入时是不会显示字符的

4. 常用命令

1) 查看设备配置

```
Router#show running-config
```

// 显示设备当前配置（配置信息保存于 RAM 中）

```
Router#show start-config
```

// 显示设备开机配置（配置信息保存于 NVRAM 中）

2) 保存你的当前配置到开机配置文件中

```
Router#write
```

或

```
Router#copy running-config start-config
```

// 上述两条命令效果相同

3) 删除开机配置文件

```
Router#erase start-config
```

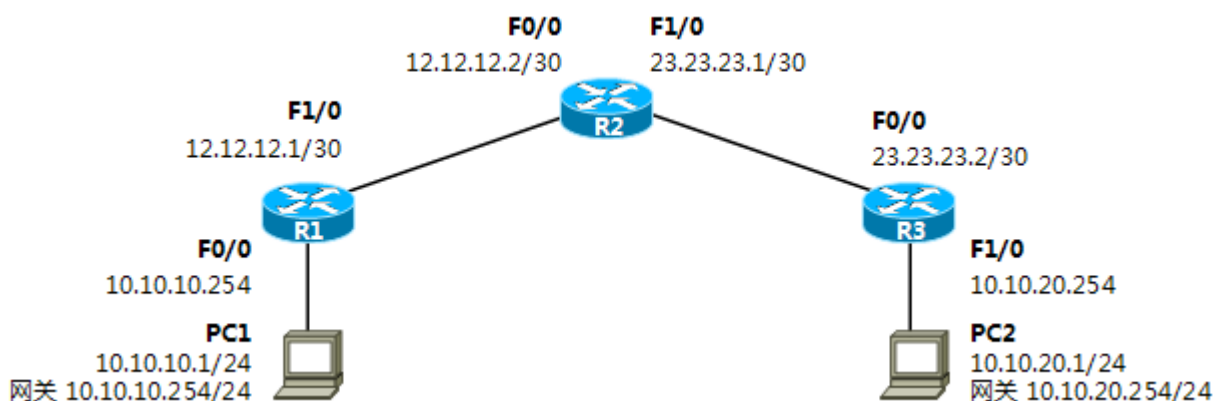
5. 实验总结

2.2 实验 2-静态路由

1. 实验目的

- 1) 掌握静态路由原理和配置命令
- 2) 掌握默认路由原理和配置命令

2. 实验拓扑及需求



需求：PC1 在 LAN1 中，PC2 在 LAN2 中，配置静态路由实现两个 LAN 中的 PC 能够通信。

3. 实验步骤和相关配置

1) 关键命令

静态路由

```
Router(config)#ip route destination-network (目的网络) destination-network-mask (目的网络掩码) ip-address (下一跳地址) /interface X (本地出接口) [distance] [permanent]
```

默认路由

```
router(config)#ip route 0.0.0.0 0.0.0.0 ip-address (下一跳地址) /interface X (本地出接口)
```

2) 实验步骤和相关配置

搭建如图拓扑，完成各设备预配、接口 IP 地址并进行相关测试。

R1：

```
Router>enable
Router#config terminal
Router(config)#hostname R1
R1(config)#interface fastethernet 0/0
```

```
R1(config-if)#ip address 10.10.10.254 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#interface fastethernet 1/0
R1(config-if)#ip address 12.12.12.1 255.255.255.252
R1(config-if)#no shutdown
```

R2 :

```
Router(config)#hostname R2
R2(config)#interface fastethernet 0/0
R2(config-if)#ip address 12.12.12.2 255.255.255.252
R2(config-if)#no shutdown
R2(config-if)#interface fastethernet 1/0
R2(config-if)#ip address 23.23.23.1 255.255.255.252
R2(config-if)#no shutdown
```

R3:

```
Router(config)#hostname R3
R3(config)#interface fastethernet 0/0
R3(config-if)#ip address 23.23.23.2 255.255.255.252
R3(config-if)#no shutdown
R3(config-if)#interface fastethernet 1/0
R3(config-if)#ip address 10.10.20.254 255.255.255.0
R3(config-if)#no shutdown
```

对于 PC1，我们是采用路由器模拟成 PC，需要对其做以下配置

PC1:

```
Router(config)#hostname PC1
PC1(config)#no ip routing
// 将路由器模拟成 PC 机，关闭路由功能。
PC1(config)#ip default-gateway 10.10.10.254
// 为 PC 机指定网关，注意：一旦使用 no ip routing 后，路由器就失去了路由功能了，因此必须
使用 ip default-gateway 的方式为其设置默认网关，而不能使用默认路由的方式。
PC1(config)#interface fastethernet 0/0
PC1(config-if)#ip address 10.10.10.1 255.255.255.0
// 为 PC 配置 IP 地址。模拟 PC 网卡地址
PC1(config-if)#no shutdown
```

PC2 同理，完成相关配置。

测试 1：查看各设备直连接口状态

PC1#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	10.10.10.1	YES	manual	up	up

PC2#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	10.10.20.1	YES	manual	up	up

R1#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	10.10.10.254	YES	manual	up	up
FastEthernet1/0	12.12.12.1	YES	manual	up	up

R2#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	12.12.12.2	YES	manual	up	up
FastEthernet1/0	23.23.23.1	YES	manual	up	up

R3#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
FastEthernet0/0	23.23.23.2	YES	manual	up	up
FastEthernet1/0	10.10.20.254	YES	manual	up	up

测试 1 结论：各设备通过命令 show ip interface brief 检测到各接口状态和协议双 up.

思考 1：若某接口 Status up，Protocol down 是为什么？

测试 2：查看路由器直连网段是否能 ping 通

R1：

R1#ping 10.10.10.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.10.10.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 8/24/72 ms

R1#ping 12.12.12.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 12.12.12.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 8/29/56 ms

R2 :

R2#ping 12.12.12.1

R2#ping 23.23.23.2

R3#ping 23.23.23.1

R3#ping 10.10.20.1

测试 2 结论：直连网段都能 ping 通。

测试 3：查看 PC1 和 PC2 能否正常通信。

PC1#ping 10.10.20.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:

U.U.U

Success rate is 0 percent (0/5)

思考 1：为什么 LAN1 中的 PC1 和 LAN2 中的 PC2 无法正常通信？

查看相关路由表

PC1#show ip route

Default gateway is 10.10.10.254

Host	Gateway	Last Use	Total Uses	Interface
------	---------	----------	------------	-----------

ICMP redirect cache is empty

R1#show ip route

10.0.0.0/24 is subnetted, 1 subnets

C 10.10.10.0 is directly connected, FastEthernet0/0

12.0.0.0/30 is subnetted, 1 subnets

C 12.12.12.0 is directly connected, FastEthernet1/0

// PC 机可以 ping 通网关，但无法互相通信，因为沿途路由器没有相应的路由条目

4. 配置静态路由、默认路由，使 PC 机可以互相通信

1) 静态路由

R1：配置去往 10.10.20.0/24 网段的路由

R1(config)#ip route 10.10.20.0 255.255.255.0 12.12.12.2

// R1 配置去往 PC2 网段的静态路由,用下一跳的方式

查看路由表(show ip route)，增加静态路由条目

S 10.10.20.0 [1/0] via 12.12.12.2

或：

```
R1(config)#ip route 10.10.20.0 255.255.255.0 fastethernet 1/0
```

// R1 配置去往 PC2 网段的静态路由,用出接口的方式

R2：配置去往 10.10.20.0/24 网段的路由，同时为返回的去往 10.10.10.0/24 的数据包配置路由

```
R2(config)#ip route 10.10.20.0 255.255.255.0 23.23.23.2
```

// R2 配置去往 PC2 网段的静态路由

```
R2(config)#ip route 10.10.10.0 255.255.255.0 12.12.12.1
```

// R2 配置去往 PC1 网段，使得回包能正确转发

R3：配置去往 10.10.10.0/24 网段的路由

```
R3(config)#ip route 10.10.10.0 255.255.255.0 23.23.23.1
```

// R3 配置去往 PC1 网段，用下一跳的方式

测试 4：查看两台 PC 机能否通信。

```
PC1#ping 10.10.20.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 56/91/124 ms

测试 5：查看 PC1 能否 ping 通 23.23.23.0/30 网段。

```
PC1#ping 23.23.23.2
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 23.23.23.2, timeout is 2 seconds:

U.U.U

Success rate is 0 percent (0/5)

思考 2：为什么两台 PC 可以正常通信，但 PC1 无法 ping 通 23.23.23.0/30 网段？

提示 1：通信是双向的过程，路由器会根据路由表中的路由条目做转发决定。

2) 默认路由

R1：

```
R1(config)#no ip route 10.10.20.0 255.255.255.0 12.12.12.2
```

```
R1(config)#ip route 0.0.0.0 0.0.0.0 12.12.12.2
```

// 去掉之前静态路由的配置，配置默认路由

查看路由表：

```
R1#show ip route
```

```
S* 0.0.0.0/0 [1/0] via 12.12.12.2
```

R3：

```
R3(config)#no ip route 10.10.10.0 255.255.255.0 23.23.23.1
```

```
R3(config)#ip route 0.0.0.0 0.0.0.0 23.23.23.1
```

// 去掉之前静态路由的配置，配置默认路由

测试 6：查看两台 PC 机能否通信

```
PC1#ping 10.10.20.1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/100/276 ms
```

提示 2：注意默认路由使用的场合以及在什么样的情况下才会选择默认路由转发数据。

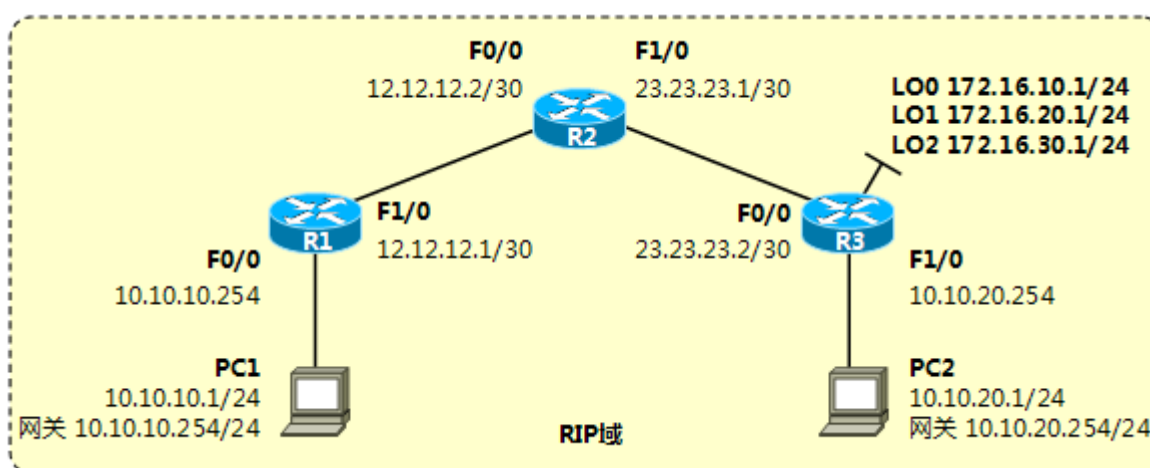
5. 实验总结

2.3 实验 3-RIP 协议

1. 实验目的

- 1) 掌握 RIPv1 的基本配置
- 2) 掌握 RIPv2 的基本配置
- 3) *了解 RIP 手动汇总，接口认证，发布默认路由

2. 实验拓扑及需求



需求：PC1 在 LAN1 中，PC2 在 LAN2 中，配置 RIP 协议实现两个 LAN 中的 PC 之间的通信。

提示：LOOPBACK 口模拟 LAN 内某一网段。

3. 实验步骤和相关配置

1) 关键命令

```
Router(config)#router rip
// 启动进程
Router(config-router)#network X.X.X.X
// RIP 主类宣告
Router(config-router)#version 2
// 将 RIP 版本切换成 version2。此条命令需根据实际需求配置，非必须配置命令
Router(config-router)#no auto-summary
// 关闭自动汇总（仅在 RIP version2 下有效）
*Router(config-router)#default-information originate
```

```
// 下发默认路由 ( 根据需求配置 )
```

```
*Router(config-if)#ip summary-address rip ip mask
```

```
// 接口手动汇总 ( 根据需求配置 )
```

2) 基本配置 (同实验 2)

R1 :

```
Router>enable
Router#config terminal
Router(config)#hostname R1
R1(config)#int fastethernet 0/0
R1(config-if)#ip address 10.10.10.254 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#interface fastethernet 1/0
R1(config-if)#ip address 12.12.12.1 255.255.255.252
R1(config-if)#no shutdown
```

R2 :

```
Router(config)#hostname R2
R2(config)#interface fastethernet 0/0
R2(config-if)#ip address 12.12.12.2 255.255.255.252
R2(config-if)#no shutdown
R2(config-if)#interface fastethernet 1/0
R2(config-if)#ip address 23.23.23.1 255.255.255.252
R2(config-if)#no shutdown
```

R3:

```
Router(config)#hostname R3
R3(config)#interface fastethernet 0/0
R3(config-if)#ip address 23.23.23.2 255.255.255.252
R3(config-if)#no shutdown
R3(config-if)#interface fastethernet 1/0
R3(config-if)#ip address 10.10.20.254 255.255.255.0
R3(config-if)#no shutdown
R3(config)#interface loopback 0
R3(config-if)#ip address 172.16.10.1 255.255.255.0
R3(config)#interface loopback 1
R3(config-if)#ip address 172.16.20.1 255.255.255.0
```

```
R3(config)#interface loopback 2
R3(config-if)#ip address 172.16.30.1 255.255.255.0
```

3) RIP 配置

R1 :

```
R1(config)#router rip
R1(config-router)#network 10.0.0.0
R1(config-router)#network 12.0.0.0
```

R2 :

```
R2(config)#router rip
R2(config-router)#network 12.0.0.0
R2(config-router)#network 23.0.0.0
```

R3 :

```
R3(config)#router rip
R3(config-router)#network 10.0.0.0
R3(config-router)#network 23.0.0.0
R3(config-router)#network 172.16.0.0
```

测试 1:PC1 是否能和 PC2 正常通信。

```
PC1#ping 10.10.20.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
U.U.U
Success rate is 0 percent (0/5)
PC1#ping 172.16.10.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.10.1, timeout is 2 seconds:
.....
Success rate is 100 percent (0/5), round-trip min/avg/max = 28/56/100 ms
```

查看 R2 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 not set

23.0.0.0/30 is subnetted, 1 subnets

C 23.23.23.0 is directly connected, FastEthernet1/0

R 172.16.0.0/16 [120/1] via 23.23.23.2, 00:00:07, FastEthernet1/0

R 10.0.0.0/8 [120/1] via 23.23.23.2, 00:00:07, FastEthernet1/0
[120/1] via 12.12.12.1, 00:00:06, FastEthernet0/0

12.0.0.0/30 is subnetted, 1 subnets

C 12.12.12.0 is directly connected, FastEthernet0/0

思考 1：为什么？

提示 1：RIP 默认为 version1，RIPv1 为距离矢量路由协议，边界自动汇总。由于 RIPv1 为有类路由协议，无法关闭自动汇总。

4) RIPv2 并且关闭自动汇总的配置

R1：

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

R2：

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

R3：

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

测试 2：两台 PC 机能否通信。

```
PC1#ping 10.10.20.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/73/120 ms
```



```
PC1#ping 172.16.10.1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 172.16.10.1, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/90/156 ms
```

查看 R2 路由表，这时候能够学习到细路由条目

```
R2#show ip route
```

```
23.0.0.0/30 is subnetted, 1 subnets
```

```
C 23.23.23.0 is directly connected, FastEthernet1/0
```

```
172.16.0.0/24 is subnetted, 3 subnets
```

```
R 172.16.30.0 [120/1] via 23.23.23.2, 00:00:01, FastEthernet1/0
```

```
R 172.16.20.0 [120/1] via 23.23.23.2, 00:00:01, FastEthernet1/0
```

```
R 172.16.10.0 [120/1] via 23.23.23.2, 00:00:01, FastEthernet1/0
```

```
10.0.0.0/24 is subnetted, 2 subnets
```

```
R 10.10.10.0 [120/1] via 12.12.12.1, 00:00:01, FastEthernet0/0
```

```
R 10.10.20.0 [120/1] via 23.23.23.2, 00:00:01, FastEthernet1/0
```

```
12.0.0.0/30 is subnetted, 1 subnets
```

```
C 12.12.12.0 is directly connected, FastEthernet0/0
```

思考 2：有没有其他方法解决不连续子网问题？

提示 2：利用 secondary 地址。

5) *手动汇总:R1,R2 关于 R3 的环回口只学习到一条汇总路由

查看 R2 路由表：

```
R2#show ip route rip
```

```
172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
```

```
R 172.16.30.0/24 [120/1] via 23.23.23.2, 00:01:25, FastEthernet1/0
```

```
R 172.16.20.0/24 [120/1] via 23.23.23.2, 00:01:25, FastEthernet1/0
```

```
R 172.16.10.0/24 [120/1] via 23.23.23.2, 00:01:25, FastEthernet1/0
```

```
10.0.0.0/24 is subnetted, 2 subnets
```

```
R 10.10.10.0 [120/1] via 12.12.12.1, 00:00:07, FastEthernet0/0
```

```
R 10.10.20.0 [120/1] via 23.23.23.2, 00:00:11, FastEthernet1/0
```

R3 手动汇总环回口路由：

```
R3(config)#interface fastethernet 0/0
```

```
R3(config-if)#ip summary-address rip 172.16.0.0 255.255.224.0
```

查看 R2 路由表：

```
R2#clear ip route *
R2#show ip route rip
    172.16.0.0/19 is subnetted, 1 subnets
R       172.16.0.0 [120/1] via 23.23.23.2, 00:00:04, FastEthernet1/0
    10.0.0.0/24 is subnetted, 2 subnets
R       10.10.10.0 [120/1] via 12.12.12.1 , 00:00:04, FastEthernet0/0
R       10.10.20.0 [120/1] via 23.23.23.2, 00:00:04, FastEthernet1/0
```

手动汇总后，减少路由表条目。

6) *RIP 协议下发默认路由

R3 不宣告环回口网段，下发默认路由，让 PC1 可以与 R3 环回口通信。

```
R3(config-router)#no network 172.16.0.0
R3(config-router)#default-information originate
R2#clear ip route *
R2#show ip route rip
    10.0.0.0/24 is subnetted, 2 subnets
R       10.10.10.0 [120/1] via 12.12.12.1 , 00:00:02, FastEthernet0/0
R       10.10.20.0 [120/1] via 23.23.23.2, 00:00:02, FastEthernet1/0
R*    0.0.0.0/0 [120/1] via 23.23.23.2, 00:00:02, FastEthernet1/0
```

7) **RIP 认证：R1-R2 的网段间两接口启用认证

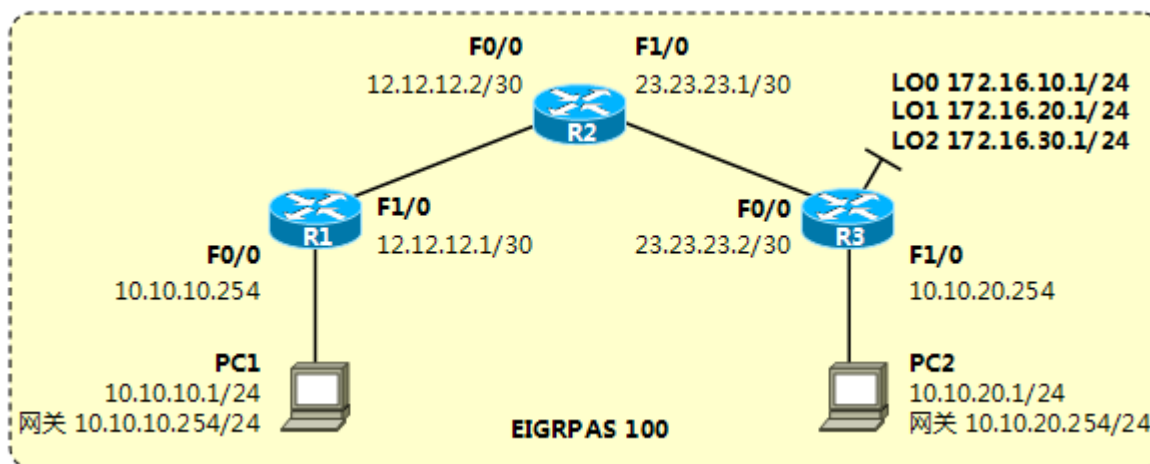
4. 实验总结

2.4 实验 4-EIGRP 协议

1. 实验目的

- 1) 掌握 EIGRP 的基本配置
- 2) 掌握 EIGRP 的手动汇总

2. 实验拓扑及需求



需求：PC1 在 LAN1 中，PC2 在 LAN2 中，配置 EIGRP 协议实现两个 LAN 中的 PC 能够通信。

提示 1：LOOPBACK 为路由器上逻辑接口，模拟 LAN 内的其他网段。

3. 实验步骤和相关配置

1) 关键命令

```
Router(config)#router eigrp as 号
Router(config-router)#network x.x.x.x [x.x.x.x]
// 不使用反掩码为主类宣告，加上反掩码达到精确宣告
```

2) 基本配置，如实验 2

3) R1,R2,R3 启用 EIGRP 进程，且 AS 号统一

R1:

```
R1(config)#router eigrp 100
R1(config-router)#network 10.0.0.0
R1(config-router)#network 12.0.0.0
```

R2 :

```
R2(config)#router eigrp 100
```

```
R2(config-router)#network 12.0.0.0
```

```
R2(config-router)#network 23.0.0.0
```

R3 :

```
R3(config)#router eigrp 100
```

```
R3(config-router)#network 10.0.0.0
```

```
R3(config-router)#network 23.0.0.0
```

```
R3(config-router)#network 172.16.0.0
```

R2 控制台信息 :

```
*Mar  1 00:19:08.579: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 12.12.12.1
(FastEthernet0/0) is up: new adjacency
```

```
*Mar  1 00:19:15.407: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 100: Neighbor 23.23.23.2
(FastEthernet1/0) is up: new adjacency
```

R2 查看 EIGRP 邻居表

```
R2#show ip eigrp neighbors
```

```
IP-EIGRP neighbors for process 100
```

H	Address	Interface	Hold Uptime (sec)	SRTT (ms)	RTO	Q Cnt	Seq Num
1	23.23.23.2	Fa1/0	11 00:00:04	91	546	0	4
0	12.12.12.1	Fa0/0	13 00:00:11	85	510	0	4

R2 与 R1,R3 建立邻居关系。

测试 1 : 两台 PC 间是否正常通信。

```
PC1#ping 10.10.20.1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
```

```
U.U.U
```

```
Success rate is 0 percent (0/5)
```

为什么不能通信呢？

查看 R1 的 EIGRP 路由表

```
R1#show ip route ei
```

```
D    23.0.0.0/8 [90/30720] via 12.12.12.2, 00:09:27, FastEthernet1/0
```

```
D    172.16.0.0/16 [90/158720] via 12.12.12.2, 00:03:32, FastEthernet1/0
```

```
10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
```

```
D    10.0.0.0/8 is a summary, 00:09:44, Null0
```

```
12.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
D      12.0.0.0/8 is a summary, 00:09:44, Null0
```

去往 10.10.20.1 的数据包转发给了 Null0 接口。

提示 2：EIGRP 为高级平衡混合路由协议，默认自动汇总。

4) 关闭自动汇总

```
R1(config-router)#no auto-summary
R2(config-router)#no auto-summary
R3(config-router)#no auto-summary
```

查看 R1 EIGRP 路由表：

```
R1#show ip route eigrp
      23.0.0.0/30 is subnetted, 1 subnets
D      23.23.23.0 [90/30720] via 12.12.12.2, 00:00:08, FastEthernet1/0
      172.16.0.0/24 is subnetted, 3 subnets
D      172.16.30.0 [90/158720] via 12.12.12.2, 00:00:03, FastEthernet1/0
D      172.16.20.0 [90/158720] via 12.12.12.2, 00:00:03, FastEthernet1/0
D      172.16.10.0 [90/158720] via 12.12.12.2, 00:00:03, FastEthernet1/0
      10.0.0.0/24 is subnetted, 2 subnets
D      10.10.20.0 [90/2684416] via 12.12.12.2, 00:00:24, FastEthernet1/0
```

测试 2：测试两台 PC 间是否能正常通信。

5) EIGRP 的手动汇总

查看手动汇总前 R1,R3 的 EIGRP 路由表

```
R1#show ip route eigrp
      23.0.0.0/30 is subnetted, 1 subnets
D      23.23.23.0 [90/30720] via 12.12.12.2, 00:00:08, FastEthernet1/0
      172.16.0.0/24 is subnetted, 3 subnets
D      172.16.30.0 [90/158720] via 12.12.12.2, 00:00:03, FastEthernet1/0
D      172.16.20.0 [90/158720] via 12.12.12.2, 00:00:03, FastEthernet1/0
D      172.16.10.0 [90/158720] via 12.12.12.2, 00:00:03, FastEthernet1/0
      10.0.0.0/24 is subnetted, 2 subnets
D      10.10.20.0 [90/33280] via 12.12.12.2, 00:00:21, FastEthernet1/0
R3#show ip route eigrp
      10.0.0.0/24 is subnetted, 2 subnets
D      10.10.10.0 [90/33280] via 23.23.23.1, 00:01:42, FastEthernet0/0
      12.0.0.0/30 is subnetted, 1 subnets
```

```
D      12.12.12.0 [90/30720] via 23.23.23.1, 00:01:11, FastEthernet0/0
```

R3 接下口手动汇总：

```
R3(config)#interface FastEthernet0/0
R3(config-if)#ip summary-address eigrp 100 172.16.0.0 255.255.224.0
R3(config-if)#do show ip route eigrp
172.16.0.0/16 is variably subnetted, 4 subnets, 2 masks
D      172.16.0.0/19 is a summary, 00:00:29, Null0
      10.0.0.0/24 is subnetted, 2 subnets
D      10.10.10.0 [90/33280] via 23.23.23.1, 00:02:58, FastEthernet0/0
      12.0.0.0/30 is subnetted, 1 subnets
D      12.12.12.0 [90/30720] via 23.23.23.1, 00:02:26, FastEthernet0/0
```

查看 R1 路由表

```
R1#show ip route eigrp
      23.0.0.0/30 is subnetted, 1 subnets
D      23.23.23.0 [90/30720] via 12.12.12.2, 00:03:10, FastEthernet1/0
      172.16.0.0/19 is subnetted, 1 subnets
D      172.16.0.0 [90/158720] via 12.12.12.2, 00:00:49, FastEthernet1/0
      10.0.0.0/24 is subnetted, 2 subnets
D      10.10.20.0 [90/33280] via 12.12.12.2, 00:03:05, FastEthernet1/0
```

查看 R2 的路由表

```
R2#show ip route eigrp
      172.16.0.0/19 is subnetted, 1 subnets
D      172.16.0.0 [90/156160] via 23.23.23.2, 00:01:00, FastEthernet1/0
      10.0.0.0/24 is subnetted, 2 subnets
D      10.10.10.0 [90/30720] via 12.12.12.1, 00:05:46, FastEthernet0/0
D      10.10.20.0 [90/30720] via 23.23.23.2, 00:05:37, FastEthernet1/0
```

结论 1：EIGRP 的汇总路由会出现在本地路由表中，下一跳指向 null0。

思考 1：Null0 接口的作用。

思考 2：EIGRP 和 RIP 的汇总异同点。

4. 实验总结

2.5 实验 5-OSPF 协议

1. 实验目的

1) 掌握 OSPF 的基本配置

OSPF 单区域实验：全网网段属于 area 0（如图 5-1）

OSPF 多区域实验：10.10.10.1/24 网段属于 area 1，172.16.0.0/16 网段属于 area 2，其他网段属于 area 0（如图 5-2）

2. 实验拓扑及需求

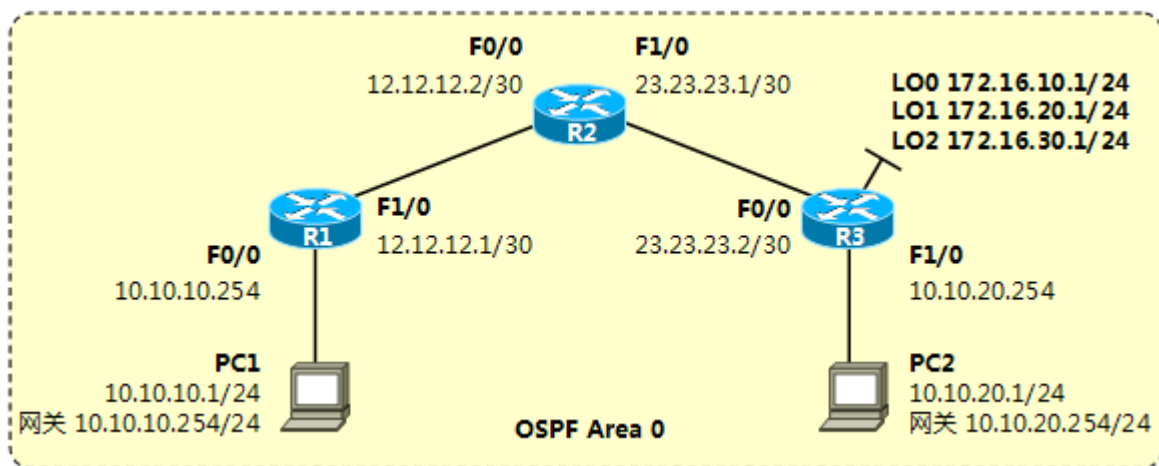


图 5-1

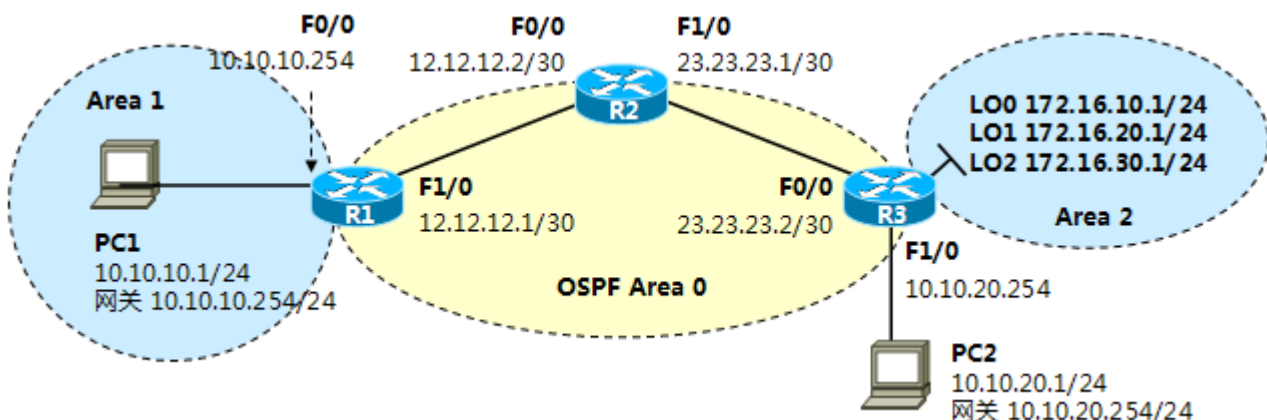


图 5-2

背景：某公司总部和分部网络实现互连，考虑分部数量进行路由域的控制。总部网络存在多个网段，实验中以 LOOPBACK 模拟。要求分部能够访问到总部的所有资源。

3. 实验步骤和相关配置

1) 关键命令

```
Route(config)#router ospf process-id
Router(config-router)#router-id X.X.X.X
// 手动指定 router-id
Router(config-router)#network x.x.x.x x.x.x.x area x
// 进程下精确宣告
*Router(config-router)#area x range x.x.x.x x.x.x.x
// ABR 上区域汇总
```

2) 基本配置(如实验 2)

3) 单区域：启用 OSPF 进程

R1：

```
R1(config)#router ospf 1
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 10.10.10.0 0.0.0.255 area 0
R1(config-router)#network 12.12.12.0 0.0.0.3 area 0
```

R2：

```
R2(config)#router ospf 1
R2(config-router)#router-id 2.2.2.2
R2(config-router)#network 12.12.12.0 0.0.0.3 area 0
R2(config-router)#network 23.23.23.0 0.0.0.3 area 0
```

R3：

```
R3(config)#router ospf 1
R3(config-router)#router-id 3.3.3.3
R3(config-router)#network 23.23.23.0 0.0.0.3 area 0
R3(config-router)#network 10.10.20.0 0.0.0.255 area 0
R3(config-router)#network 172.16.0.0 0.0.255.255 area 0
```

R2 与 R1,R3 形成邻接关系，以下为控制台消息：

```
*Mar  1 00:32:59.167: %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on FastEthernet0/0 from
LOADING to FULL, Loading Done
*Mar  1 00:33:10.571: %OSPF-5-ADJCHG: Process 1, Nbr 3.3.3.3 on FastEthernet1/0 from
LOADING to FULL, Loading Done
```

R2 上查看 OSPF 的邻居表


```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
3.3.3.3	1	FULL/ DR	00:00:39	23.23.23.2	FastEthernet1/0
1.1.1.1	1	FULL/ BDR	00:00:37	12.12.12.1	FastEthernet0/0

查看 R1 路由器的 OSPF 路由表

```
R1#show ip route ospf
```

```

23.0.0.0/30 is subnetted, 1 subnets
O       23.23.23.0 [110/2] via 12.12.12.2, 00:00:15, FastEthernet1/0
172.16.0.0/32 is subnetted, 3 subnets
O       172.16.30.1 [110/129] via 12.12.12.2, 00:01:05, FastEthernet1/0
O       172.16.20.1 [110/129] via 12.12.12.2, 00:01:05, FastEthernet1/0
O       172.16.10.1 [110/129] via 12.12.12.2, 00:01:05, FastEthernet1/0
10.0.0.0/24 is subnetted, 2 subnets
O       10.10.20.0 [110/129] via 12.12.12.2, 00:01:05, FastEthernet1/0

```

思考 1：为什么 172.16.0.0 是/32 的网段。

查看 R2 路由器的 OSPF 路由表

```
R2#show ip route ospf
```

```

172.16.0.0/32 is subnetted, 3 subnets
O       172.16.30.1 [110/65] via 23.23.23.2, 00:01:30, FastEthernet1/0
O       172.16.20.1 [110/65] via 23.23.23.2, 00:01:30, FastEthernet1/0
O       172.16.10.1 [110/65] via 23.23.23.2, 00:01:30, FastEthernet1/0
10.0.0.0/24 is subnetted, 2 subnets
O       10.10.10.0 [110/65] via 12.12.12.1, 00:01:30, FastEthernet0/0
O       10.10.20.0 [110/65] via 23.23.23.2, 00:01:30, FastEthernet1/0

```

查看 R3 路由器的 OSPF 路由表

```
R3#show ip route ospf
```

```

12.0.0.0/30 is subnetted, 1 subnets
O       12.12.12.0 [110/2] via 23.23.23.1, 00:01:15, FastEthernet0/0
10.0.0.0/24 is subnetted, 2 subnets
O       10.10.10.0 [110/129] via 23.23.23.1, 00:01:45, FastEthernet0/0

```

测试 1：两 PC 是否能正常通信。

```
PC1#ping 10.10.20.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 32/73/120 ms

查看 R1 的 F0/0 接口

```
R1#show ip ospf interface f0/0
```

```
FastEthernet0/0 is up, line protocol is up
```

```
Internet Address 10.10.10.254/24, Area 0
```

```
Process ID 1, Router ID 12.12.12.1, Network Type BROADCAST, Cost: 1
```

```
Transmit Delay is 1 sec, State WAITING, Priority 1
```

```
No designated router on this network
```

```
No backup designated router on this network
```

```
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
```

```
oob-resync timeout 40
```

```
Hello due in 00:00:07
```

```
Wait time before Designated router selection 00:00:17
```

```
Supports Link-local Signaling (LLS)
```

```
Index 1/1, flood queue length 0
```

```
Next 0x0(0)/0x0(0)
```

```
Last flood scan length is 0, maximum is 0
```

```
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)
```

4) 多区域：

R1：

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

```
R1(config-router)#network 10.10.10.0 0.0.0.255 area 1
```

R3：

```
R3(config)#router ospf 1
```

```
R3(config-router)#network 172.16.0.0 0.0.255.255 area 2
```

查看 R1 路由的 OSPF 路由表：

```
R1#show ip route ospf
```

```
23.0.0.0/30 is subnetted, 1 subnets
```

```
O 23.23.23.0 [110/2] via 12.12.12.2, 00:00:15, FastEthernet1/0
```

```
172.16.0.0/32 is subnetted, 3 subnets
O IA    172.16.30.1 [110/129] via 12.12.12.2, 00:02:51, FastEthernet1/0
O IA    172.16.20.1 [110/129] via 12.12.12.2, 00:02:51, FastEthernet1/0
O IA    172.16.10.1 [110/129] via 12.12.12.2, 00:02:51, FastEthernet1/0

10.0.0.0/24 is subnetted, 2 subnets
O       10.10.20.0 [110/129] via 12.12.12.2, 00:02:51, FastEthernet1/0
```

查看 R2 路由的 OSPF 路由表：

```
R2#show ip route ospf

172.16.0.0/32 is subnetted, 3 subnets
O IA    172.16.30.1 [110/65] via 23.23.23.2, 00:03:02, FastEthernet1/0
O IA    172.16.20.1 [110/65] via 23.23.23.2, 00:03:02, FastEthernet1/0
O IA    172.16.10.1 [110/65] via 23.23.23.2, 00:03:02, FastEthernet1/0

10.0.0.0/24 is subnetted, 2 subnets
O IA    10.10.10.0 [110/65] via 12.12.12.1 , 00:03:02, FastEthernet0/0
O       10.10.20.0 [110/65] via 23.23.23.2, 00:03:02, FastEthernet1/0
```

查看 R3 路由的 OSPF 路由表：

```
R3#show ip route ospf

12.0.0.0/30 is subnetted, 1 subnets
O       12.12.12.0 [110/2] via 23.23.23.1, 00:01:15, FastEthernet0/0

10.0.0.0/24 is subnetted, 2 subnets
O IA    10.10.10.0 [110/129] via 23.23.23.1, 00:02:23, FastEthernet0/0
```

思考 2：O IA 标志说明什么？

测试 2：两台 PC 是否能正常通信。

```
PC1#ping 10.10.20.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/73/120 ms
```

4. 实验小结

3 交换基础实验

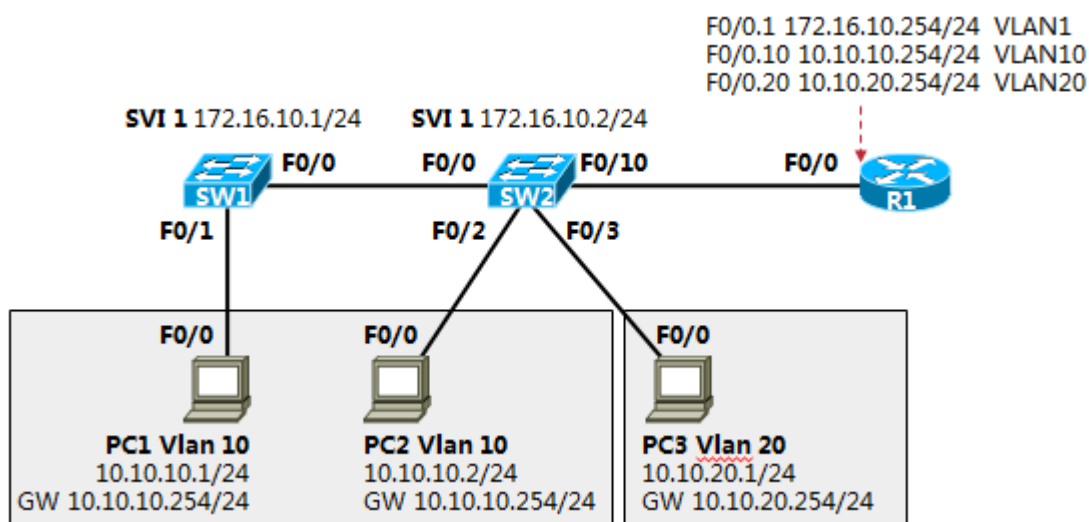
3.1 实验 1-VLAN/TRUNK/VTP/管理

1. 实验目的

掌握二层交换的基本配置

- 1) VLAN
- 2) TRUNK
- 3) VTP
- 4) 交换机远程管理

2. 实验拓扑及需求



背景：某公司出于安全考虑，将内网进行广播域隔离，同时需要实现内网间互访，网管员能够远程管理内网设备。

3. 实验步骤和相关配置

1) 关键命令

划分 VLAN：

```
Switch#vlan database
Switch(vlan-config)#vlan vlan-id name word
或 Switch(config)#vlan vlan-id
Switch(vlan-config)#name word
```

```
// 特权模式下或 vlan 数据库里创建 vlan
Switch(vlan-config)#exit
或
Switch(vlan-config)#apply
// 命令 exit 或 apply 使相关配置生效
Switch(config)#interface x
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan vlan-id
// 将接口加入 vlan
Switch#show vlan-switch
// 模拟器环境下使用 show vlan-switch,在真实环境中使用 show vlan brief
Switch#show vlan vlan-id
// 查看 vlan 和接口
```

配置 Trunk

```
Switch(config-if)#interface x
Switch(config-if)#switchport trunk encapsulation dot1q/isl
Switch(config-if)#switchport mode trunk
// 配置 trunk
Switch#show interface trunk
// 查看 trunk
```

交换机远程管理

```
Switch(config)#interface vlan 1
Switch(config)#ip address x.x.x.x x.x.x.x
Switch(config)#ip default-gateway x.x.x.x
// 为二层交换机设置网关
```

2) 路由器模拟 PC 机的配置.

3) VLAN,Trunk 配置

SW1 :

```
sw1#vlan database
sw1(vlan)#vlan 10 name vlan10
VLAN 10 added:
  Name: vlan10
sw1(vlan)#vlan 20 name vlan20
VLAN 20 added:
```

```

Name: vlan20
sw1(vlan)#exit
APPLY completed.
Exiting....
sw1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
sw1(config-if)#interface fastethernet 0/0
sw1(config-if)#switchport trunk encapsulation dot1q
sw1(config-if)#switchport mode trunk
*Mar  1 00:03:35.019: %DTP-5-TRUNKPORTON: Port Fa0/0 has become dot1q trunk
sw1(config-if)#interface f0/1
sw1(config-if)#switchport mode access
sw1(config-if)#switchport access vlan 10
  
```

查看 vlan 和 trunk 信息

```

sw1#show vlan-switch
VLAN Name                Status    Ports
-----
1    default                active    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, Fa0/15
10   vlan10                  active    Fa0/1
20   vlan20                  active
sw1#show interface trunk
  
```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/0	on	802.1q	trunking	1

SW2 :

```

sw2(config)#interface fastethernet 0/0
sw2(config-if)#switchport trunk encapsulation dot1q
sw2(config-if)#switchport mode trunk
sw2(config-if)#interface f0/10
sw2(config-if)#switchport trunk encapsulation dot1q
sw2(config-if)#switchport mode trunk
  
```

SW2 上查看 trunk 接口

```
sw2#show interface trunk
```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/0	on	802.1q	trunking	1
Fa0/10	on	802.1q	trunking	1

```
sw2#show vlan-switch
```

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, Fa0/11, Fa0/12, Fa0/13 Fa0/14, Fa0/15

4) VTP 配置

需求：SW2 通过 VTP 学习到 VLAN 信息

基本命令

```
Switch#vlan database
```

```
Switch(vlan)#vtp server/client/transparent
```

// 设置 VTP 模式

```
Switch(vlan)#vtp domain name
```

// 设置 VTP 域名

```
Switch(vlan)#vtp password xx
```

// 设置 VTP 密码，可选，若配置密码，每台设备密码必须相同

```
Switch(vlan)#vtp pruning
```

// 设置 VTP 修剪，可选

SW1：

```
sw1#vlan database
```

```
sw1(vlan)#vtp domain SPOTO
```

Changing VTP domain name from NULL to SPOTO

```
sw1(vlan)#exit
```

APPLY completed.

Exiting....

SW2：

```

sw2#vlan database
sw2(vlan)#vtp client
sw2(vlan)#vtp domain SPOTO
Domain name already set to SPOTO .
sw2(vlan)#exit
APPLY completed.
Exiting....
sw2#conf t
sw2(config)#interface f0/2
sw2(config-if)#switchport access vlan 10
sw2(config-if)#interface f0/3
sw2(config-if)#switchport access vlan 20
sw2(config-if)#do show vlan-switch
  
```

VLAN Name	Status	Ports
1 default	active	Fa0/1, Fa0/4, Fa0/5, Fa0/6 Fa0/7, Fa0/8, Fa0/9, Fa0/11 Fa0/12, Fa0/13, Fa0/14, Fa0/15
10 vlan10	active	Fa0/2
20 vlan20	active	Fa0/3

测试 1：PC 间是否能正常通信？

同一 VLAN：

```

PC1#ping 10.10.10.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/73/120 ms
  
```

不同 VLAN：

```

PC1#ping 10.10.20.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
  
```


5) 单臂路由配置

关键命令：

```
Router(config)#interface x
Router(config-if)#no ip address
Router(config-if)#no shutdown
Router(config-if)#interface x.x
Router(config-subif)#encapsulation dot1q vlan-id
Router(config-subif)#ip address x.x.x.x
```

R1：

```
R1(config)#interface fastethernet 0/0
R1(config-if)#no ip address
R1(config-if)#no shut
R1(config-if)#interface fastethernet 0/0.10
R1(config-subif)#encapsulation dot1q 10
R1(config-subif)#ip address 10.10.10.254 255.255.255.0
R1(config-subif)#interface fastethernet 0/0.20
R1(config-subif)#encapsulation dot1q 20
R1(config-subif)#ip address 10.10.20.254 255.255.255.0
```

测试 2：不同 VLAN 间 PC 间能否正常通信？

```
PC1#ping 10.10.20.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5)
```

6) PC1 远程管理交换机

SW1：

```
sw1(config)#no ip routing
sw1(config)#interface vlan 1
sw1(config-if)#ip address 172.16.10.1 255.255.255.0
sw1(config)# ip default-gateway 172.16.10.254
sw1(config)#line vty 0 4
sw1(config-line)#password spoto
sw1(config-line)#login
```

SW2 :

```
sw2(config)#no ip routing
sw2(config)#interface vlan 1
sw2(config-if)#ip address 172.16.10.2 255.255.255.0
sw2(config)# ip default-gateway 172.16.10.254
sw2(config)#line vty 0 4
sw2(config-line)#password spoto
sw2(config-line)#login
```

R1 :

```
R1(config)#int fastethernet 0/0.1
R1(config-subif)#encapsulation dot1q 1
R1(config-subif)#ip address 172.16.10.254 255.255.255.0
```

测试 3 : PC1 远程管理 SW1

```
PC1#telnet 172.16.10.1
Trying 172.16.10.1 ... Open

User Access Verification

Password:
sw1>
```

PC1 远程管理 SW2

```
PC1#telnet 172.16.10.2
Trying 172.16.10.2 ... Open

User Access Verification

Password:
Sw2>
```

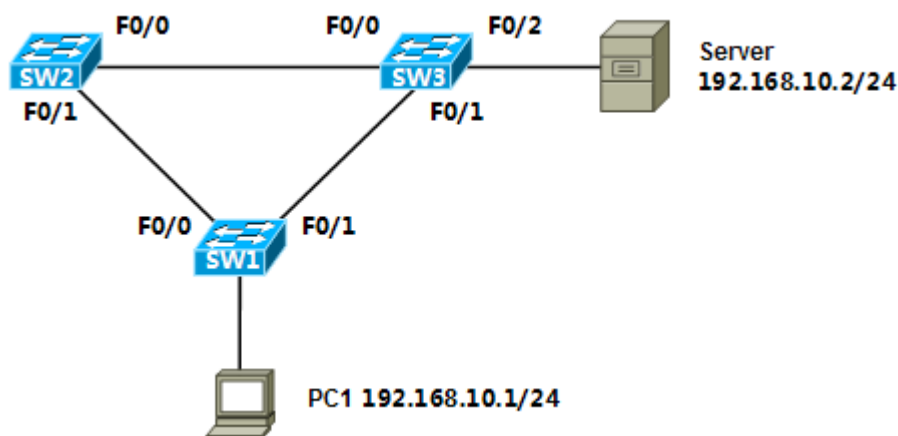
4. 实验总结

3.2 实验 2-STP 配置*

1. 实验目的

- 1) 掌握 STP 的基本原理
- 2) 通过修改网桥优先级选举根桥
- 3) 通过修改 cost 值，选举 RP

2. 实验拓扑及需求



背景：某公司为提供 PC 机访问服务器的冗余，使用多台交换机互连，要求在冗余的基础上防止二层环路。

需求：CISCO 的交换机默认开启 STP，且模式为 PVST+（CISCO 私有），当现网中存在多个厂家的交换机时，一般 CISCO 的交换机开启 STP，并将模式改为 MST（多实例生成树协议）。其他厂家交换机关闭 STP。此拓扑中，SW1 为接入层交换机，而 SW2 和 SW3 为汇聚层交换机。

注：交换机上更改 STP 的模式，及 MST 的相关配置会在后续课程中学习。

3. 主要步骤和相关配置

1) 关键命令：

```
Switch(config)#spanning-tree vlan x priority x
Switch(config-if)#spanning-tree vlan x cost x
```

2) 基本配置，PC1 和 Server 处于同一网段，实现 PC 访问 Server 链路冗余。

测试 1：PC1#192.168.10.2

3) 修改网桥优先级，使得汇聚层 SW2 成为根桥

利用 show spanning-tree brief 查看根桥：(此处由于模拟器各交换机的 MAC 地址的随机性，使

得根桥也具有随机性。为了达到实验目的,通过命令修改优先级使得根桥的位置变化为另一台汇聚层交换机即可)

SW3 :

```
sw3#show spanning-tree brief
VLAN1
  Spanning tree enabled protocol ieee
  Root ID    Priority    32768
              Address     cc00.03cc.0000
              This bridge is the root
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    32768
              Address     cc00.03cc.0000
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time 300
```

SW2 :

```
sw2#show spanning-tree brief
VLAN1
  Spanning tree enabled protocol ieee
  Root ID    Priority    32768
              Address     cc00.03cc.0000
              Cost         19
              Port         1 (FastEthernet0/0)
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    32768
              Address     cc00.0e68.0000
              Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time 300

SW2(config)#spanning-tree vlan 1 priority 4096
SW2#show spanning-tree brief
VLAN1
  Spanning tree enabled protocol ieee
  Root ID    Priority    4096
              Address     cc00.0e68.0000
              This bridge is the root
```

```

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 4096
Address cc00.0e68.0000
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 300
  
```

4) 修改 cost 值，使得 SW1 的 f0/1 为 RP

查看 SW1 的端口状态

```
SW1# show spanning-tree brief
```

Interface Name	Port ID	Prio	Cost	Sts	Cost	Bridge ID	Port ID
FastEthernet0/0	128.1	128	19	FWD	0	4096 cc00.0e68.0000	128.2
FastEthernet0/1	128.2	128	19	BLK	19 32768	cc00.03cc.0000	128.2

```
SW1(config)#interface f0/0
```

```
SW1(config-if)#spanning-tree cost 40
```

```
SW1(config-if)#do show spanning-tree brief
```

Interface Name	Port ID	Prio	Cost	Sts	Cost	Bridge ID	Port ID
FastEthernet0/0	128.1	128	40	BLK	0	4096 cc00.0e68.0000	128.2
FastEthernet0/1	128.2	128	19	FWD	19 32768	cc00.03cc.0000	128.2

测试 2 : PC1#ping 192.168.10.2

关闭 SW1 的 f0/1 接口

```
SW1(config)#interface fastethernet 0/1
```

```
SW1(config-if)#shutdown
```

测试 3 : PC1#ping 192.168.10.2

4. 实验总结

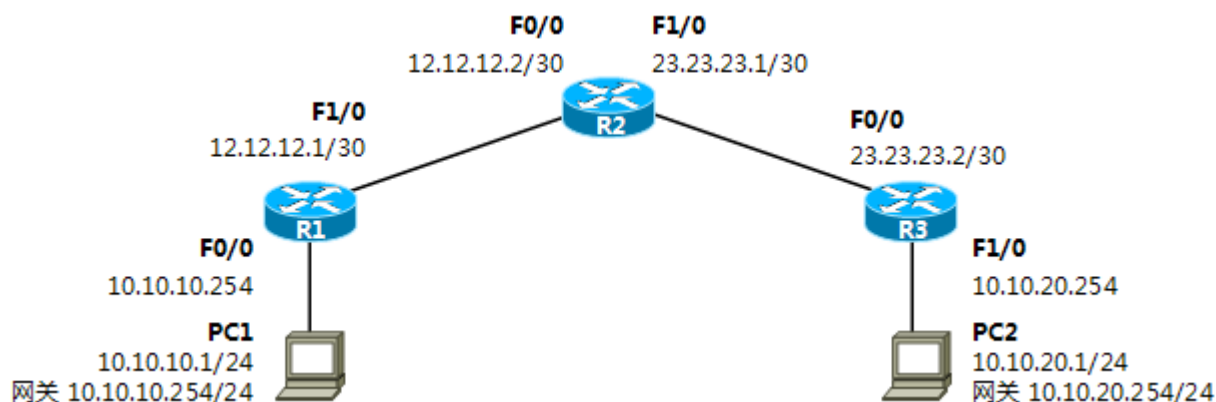
4 路由策略实验

4.1 实验 1-ACL 实验

1. 实验目的：

- 1) 掌握标准和扩展的 ACL 原理和配置
- 2) 掌握命名的 ACL 的原理和配置

2. 拓扑和需求



背景：某公司总部和分部网络互连，内网间的通信有部分应用需要做隔离。

需求：两个 LAN 中的 PC 可以正常通信，在路由器的相应接口应用 ACL，实现如下要求

- 1) PC1、PC2 之间通过路由器实现互连。使用静态路由协议实现。
- 2) 标准 ACL 配置及应用：在 R1 上配置 ACL 实现除 PC2 网段外都能访问到 PC1。
- 3) 标准 ACL 实现 Telnet 登陆控制：在 R2 上配置只允许 PC2 远程管理 R2。
- 4) 扩展 ACL 配置及应用：在 R3 配置 ACL 实现 PC2 只允许被 ping 通，其他访问都不能进行。
- 5) *命名 ACL 实现上述需求。

3. 主要步骤和相关配置

1) 关键命令

标准 ACL

```
Router(config)#access-list num(1-99) permit/deny source-ip wild-mask
```

// 配置标准 ACL

```
Router(config-if)#ip access-group x in/out
```

// 接口下套用 ACL

```
Router(config)#line vty 0 4
Router(line-config)#access-class numbel in
// vty 线路下套用标准 ACL
Router# show access-list
// 查看 ACL
```

扩展的 ACL

```
Router(config)#access-list num(100-199) permit/deny protocol soucre-ip wildmask eq
port-num des-ip wildmask eq port-num
// 配置扩展 ACL
Router(config-if)#ip access-group x in/out
// 接口下套用 ACL
```

*命名的 ACL

```
Router(config)#ip access-list standard word
Router(standar)#permit/deny source-ip wildmask
Router(config)#ip access-list extended word
Router(standar)#permit/deny protocol source-ip wildmask des-ip wildmask
// 配置 ACL
Router(config-if)#ip access-group word in/out
// 套用 ACL
```

2) 配置 ACL

路由器模拟 PC，路由器使用静态路由实现两 PC 正常通信。(略)

A 需求:在 R1 上拒绝 PC2 所在网段的流量，允许其他流量通过。

```
R1(config)#access-list 10 deny 10.10.20.0 0.0.0.255
R1(config)#access-list 10 permit any
R1(config)#interface fastethernet 1/0
R1(config-if)#ip access-group 10 in
```

测试 1：

```
R2#ping 10.10.10.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.10.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5)
PC2#ping 10.10.10.1
```

```
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 10.10.10.1, timeout is 2 seconds:  
u.u.u  
Success rate is 0 percent (0/5)
```

结论 1：R2 能访问 PC1,PC2 网段无法访问 R1。

思考：仅仅拒绝 PC2 主机的流量有几种配置方式？

B 需求：只允许 PC2 所在的网段 telnet R2

```
R2(config)#access-list 1 permit 10.10.20.0 0.0.0.255  
R2(config)#line vty 0 4  
R2(config-line)#access-class 1 in
```

测试 2:

```
PC2# telnet 23.23.23.1  
Trying 23.23.23.1 ... Open  
  
Password required, but none set  
PC1#telnet 23.23.23.1  
Trying 23.23.23.1 ...  
% Connection refused by remote host
```

C 需求：只允许 PC1 能 PING 通 PC2，但是不能 TLENET 到 PC2。

R1：

```
R1(config)#no access-list 10  
R1(config)#interface fastethernet 1/0  
R1(config-if)#no ip access-group 10 in
```

R3：

```
R3(config)#access-list 100 permit icmp any 10.10.20.0 0.0.0.255  
R3(config)#interface fastethernet 0/0  
R3(config-if)#ip access-group 100 in
```

测试 3:

```
PC1#ping 10.10.20.1  
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:  
!!!!  
Success rate is 100 percent (5/5)  
PC1# telnet 10.10.20.1
```



```
Trying 10.10.20.1 ...
```

```
% Connection refused by remote host
```

D 需求：PC1 能 PING 通 PC2，但是 R1 不能 PING 通 PC2

R3：

```
R3(config)#no access-list 100
```

```
R3(config)#interface fastethernet 0/0
```

```
R3(config-if)#no ip access-group 100 in
```

R2：

```
R2(config)#access-list 101 permit ip 10.10.10.0 0.0.0.255 10.10.20.0 0.0.0.255
```

```
R2(config)#access-list 101 deny ip 12.12.12.0 0.0.0.3 10.10.20.0 0.0.0.255
```

```
R2(config)#int f0/0
```

```
R2(config-if)#ip access-group 101 in
```

测试 1：

```
PC1#ping 10.10.20.1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 124/167/248 ms
```

```
R1#ping 10.10.20.1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
```

```
U.U.U
```

```
Success rate is 0 percent (0/5)
```

测试 2：

```
R2#ping 10.10.20.1 source 12.12.12.2
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
```

```
Packet sent with a source address of 12.12.12.2
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/81/112 ms
```

思考：为什么 12.12.12.1 不能与 PC2 通信，而 12.12.12.2 能与 PC2 通信？

E 命名的 ACL

实现拒绝 PC1 telnet 到 PC2，允许 PC1 ping 通 PC2.

```
R2(config)#no access-list 101
R2(config)#ip access-list extended cisco
R2(config-ext-nacl)#10 deny tcp host 10.10.10.1 host 10.10.20.1 eq 23
R2(config-ext-nacl)#permit ip any any
R2(config)#int f0/0
R2(config-if)#ip access-group cisco in
```

测试：

```
PC1#ping 10.10.20.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.10.20.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 156/192/216 ms

PC1#telnet 10.10.20.1
Trying 10.10.20.1 ...
% Destination unreachable; gateway or host down
```

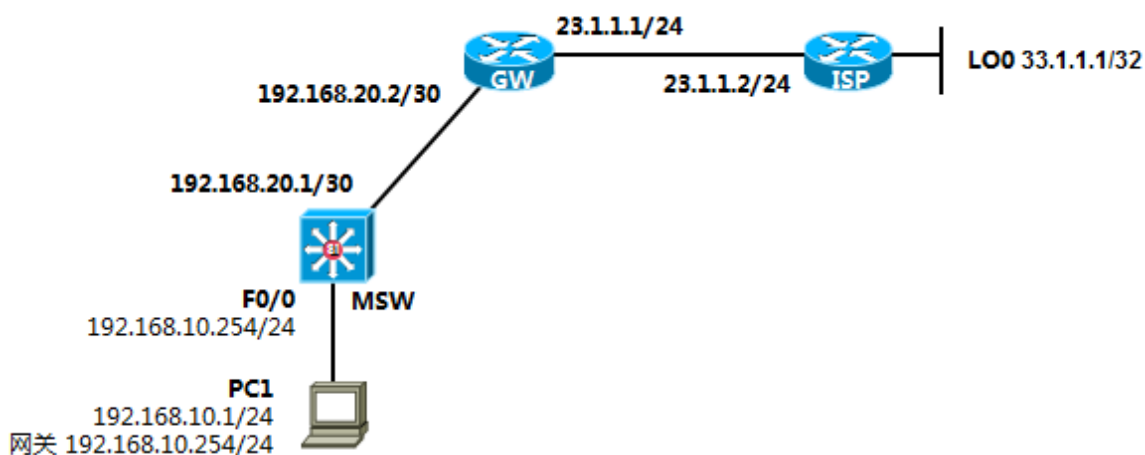
4. 实验总结

4.2 实验 2-NAT 实验

1. 实验目的：

掌握静态 NAT、动态 NAT、PAT 的配置

2. 拓扑和需求



背景：某公司组建网络接入 Internet,使用了多层交换机作为终端电脑的网关，使用路由器作为 Internet 接入设备。（防火墙也可作为 Internet 接入设备。Internet 接入方式包括：光纤接入，网线接入，电话线接入。公网 IP 地址的分配包括：固定 IP，PPPoE。）

需求：PC1 位于 LAN 内，利用 NAT 实现内网可以访问外网。

3. 实验步骤和相关配置

1) 关键命令

```

ip nat inside
// 指明内网接口

ip nat outside
// 指明外网接口

ip nat inside source static x.x.x.x x.x.x.x
// 静态 NAT

access-list x permit any

ip nat pool word first-ip last-ip netmask mask
  
```

```
ip nat inside source list x pool word
// 动态 NAT
access-list x permit any
ip nat inside source list x interface x overload
// 复用出接口 IP 的 PAT
access-list x permit any
ip nat pool word first-ip last-ip netmask mask
ip nat inside source list x pool word overload
// 复用某个公网 IP 的 PAT
```

2) 路由器模拟 PC，路由器预配及接口 IP (略),MSW 两个接口为 RP.

```
MSW(config)#interface fastethernet 0/0
MSW(config-if)#no switchport
// 启用路由接口
MSW(config-if)#ip address 192.168.10.254 255.255.255.0
MSW(config-if)#interface fastethernet 0/1
MSW(config-if)#no switchport
MSW(config-if)#ip address 192.168.20.1 255.255.255.252
```

3) 利用静态路由实现 PC1-MSW-GW 组成的内网正常通信，MSW，网关配置默认路由可访问外网 GW：

```
GW(config)#ip route 192.168.10.0 255.255.255.0 192.168.20.1
GW(config)#ip route 0.0.0.0 0.0.0.0 23.1.1.2
```

MSW：

```
MSW(config)#ip routing
MSW(config)#ip route 0.0.0.0 0.0.0.0 192.168.20.2
```

4) 配置静态 NAT，实现 PC1 与 ISP 的 loopback0 互访

```
GW(config)#ip nat inside source static 192.168.10.1 23.1.1.3
GW(config)#interface fastethernet 0/0
GW(config-if)#ip nat inside
GW(config-if)#interface fastethernet 1/0
GW(config-if)#ip nat outside
GW#show ip nat translations
```

Pro	Inside global	Inside local	Outside local	Outside global
---	23.1.1.3	192.168.10.1	---	---

测试 1：

```
PC1#ping 33.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 33.1.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5)
```

测试 ISP 的 loopback0 的地址与 nat 地址 23.1.1.3 的通信：

```
ISP#ping 23.1.1.3 source loo0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 23.1.1.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5)
```

GW 查看相关 debug 消息

```
GW#debug ip nat
IP NAT debugging is on
GW#
*Mar  1 00:10:28.027: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [30]
*Mar  1 00:10:28.091: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [30]
*Mar  1 00:10:28.131: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [31]
*Mar  1 00:10:28.151: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [31]
*Mar  1 00:10:28.167: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [32]
*Mar  1 00:10:28.179: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [32]
*Mar  1 00:10:28.215: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [33]
*Mar  1 00:10:28.231: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [33]
*Mar  1 00:10:28.271: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [34]
*Mar  1 00:10:28.279: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [34]
*Mar  1 00:10:31.003: NAT: expiring 23.1.1.3 (192.168.10.1) icmp 5 (5)
```

结论 1：创建静态 NAT，即可看到 NAT 转换表形成一对一的对应关系，使得外网也可访问内网。

5) 配置动态 NAT，实现 PC1 与 ISP 的 Ip0 互访

```
GW(config)#no ip nat inside source static 192.168.10.1 23.1.1.3
GW(config)#access-list 10 permit any
GW(config)#ip nat pool SPOTO 23.1.1.3 23.1.1.254 netmask 255.255.255.0
GW(config)#ip nat inside source list 10 pool SPOTO
GW#show ip nat translations
```

结论 2：动态 NAT 在没有流量的情况下 NAT 转换表为空。

测试 2 :

```
PC1#ping 33.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 33.1.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5)
```

测试 ISP 的 loopback0 的地址与 nat 地址 23.1.1.3 的通信 :

```
ISP#ping 23.1.1.3 source loo0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 23.1.1.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5)
```

GW 查看 NAT 表

```
GW#show ip nat translations

Pro Inside global      Inside local       Outside local      Outside global
icmp 23.1.1.3:2        192.168.10.1 :2    23.1.1.1:2        23.1.1.1:2
--- 23.1.1.3          192.168.10.1      ---                ---

GW#
*Mar  1 00:12:59.431: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [35]
*Mar  1 00:12:59.479: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [35]
*Mar  1 00:12:59.531: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [36]
*Mar  1 00:12:59.579: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [36]
*Mar  1 00:12:59.599: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [37]
*Mar  1 00:12:59.619: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [37]
*Mar  1 00:12:59.623: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [38]
*Mar  1 00:12:59.639: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [38]
*Mar  1 00:12:59.647: NAT*: s=192.168.10.1 23.1.1.3, d=33.1.1.1 [39]
GW#
*Mar  1 00:12:59.663: NAT*: s=33.1.1.1, d=23.1.1.3 192.168.10.1 [39]
```

结论 3 : 由于内网有访问外网的流量 , 故 GW 处可查看到一对一的 NAT 的转换表。使得外网也可访问内网。

6) 配置 PAT 实现内网访问外网

```
GW(config)#no ip nat pool SPOTO 23.1.1.3 23.1.1.254 netmask 255.255.255.0
```

```
GW(config)#no ip nat inside source list 10 pool SPOTO
GW(config)#ip nat inside source list 10 interface fastethernet 1/0 overload
GW#show ip nat translations
```

GW#

测试 3 :

```
PC1#ping 33.1.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 33.1.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5)
ISP#ping 23.1.1.3 source loo0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 23.1.1.3, timeout is 2 seconds:
.....
Success rate is 100 percent (0/5)
```

GW 查看 NAT 表

```
GW#show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
icmp 23.1.1.1:12       192.168.10.1:12   33.1.1.1:12       33.1.1.1:12
GW#
*Mar  1 00:17:41.335: NAT*: s=192.168.10.1 23.1.1.1, d=33.1.1.1 [45]
*Mar  1 00:17:41.431: NAT*: s=33.1.1.1, d=23.1.1.1 192.168.10.1 [45]
*Mar  1 00:17:41.447: NAT*: s=192.168.10.1 23.1.1.1, d=33.1.1.1 [46]
*Mar  1 00:17:41.467: NAT*: s=33.1.1.1, d=23.1.1.1 192.168.10.1 [46]
*Mar  1 00:17:41.479: NAT*: s=192.168.10.1 23.1.1.1, d=33.1.1.1 [47]
*Mar  1 00:17:41.495: NAT*: s=33.1.1.1, d=23.1.1.1 192.168.10.1 [47]
*Mar  1 00:17:41.499: NAT*: s=192.168.10.1 23.1.1.1, d=33.1.1.1 [48]
*Mar  1 00:17:41.515: NAT*: s=33.1.1.1, d=23.1.1.1 192.168.10.1 [48]
*Mar  1 00:17:41.519: NAT*: s=192.168.10.1 23.1.1.1, d=33.1.1.1 [49]
*Mar  1 00:17:41.531: NAT*: s=33.1.1.1, d=23.1.1.1 192.168.10.1 [49]
```

结论 4 : PAT 转换表在没有流量时是空的 , 有了流量后形成多对一的关系。

4. 实验总结

5 广域网和 IPv6 实验

5.1 实验 1-HDLC/Frame-Relay

1. 实验目的：

- 1) 掌握 WAN 接口的 HDLC 和 Frame-relay 封装。

2. 实验拓扑和需求



背景：某公司总部和分部网络通过 WAN 线路互连，考虑分别使用 HDLC（2M DDN 专线）和 Frame-relay（包交换）的差别。

3. 主要步骤和相关配置

1) 关键命令

HDLC

```
Router(config)#interface serial 0/0
Router(config-if)#encapsulation hdlc
```

Frame-relay

```
Router(config)#interface serial 0/0
Router(config-if)#encapsulation frame-relay
Router(config-if)#ip address x.x.x.x x.x.x.x
Router(config-if)#frame-relay map ip x.x.x.x dlci broadcast
Router(config-if)#no frame-relay inverse-arp
```

2) 基本配置,各设备预配和各接口 IP。

3) Frame-relay 动态配置

FRSW(帧中继交换机):

```
FRSW(config)#frame-relay switching
```

// 将路由器模拟成帧中继交换机


```
FRSW(config)#interface serial 0/0
FRSW(config-if)#encapsulation frame-relay
// 将接口封装为帧中继
FRSW(config-if)#frame-relay intf-type dce
// 指定接口类型为 DCE 端
FRSW(config-if)#clock rate 64000
// 为 DCE 端配置时钟频率
FRSW(config-if)#no shutdown
FRSW(config-if)#frame-relay route 102 interface searial 0/1 201
// 配置 frame-relay route,建立转发表,102 为本端接口 DLCI 号
FRSW(config-if)#interface searial 0/1
FRSW(config-if)#encapsulation frame-relay
FRSW(config-if)#frame-relay intf-type dce
FRSW(config-if)#clock rate 64000
FRSW(config-if)#frame-relay route 201 interface serial 0/0 102
FRSW(config-if)#no shutdown
FRSW(config-if)#end
```

R1 :

```
R1(config)#interface searial 0/0
R1(config-if)#encapsulation frame-relay
R1(config-if)#ip adderss 10.1.1.1 255.255.255.0
R1(config-if)#frame-relay lmi-type cisco
// 管理类型为 cisco
```

R2 :

```
R2(config)#interface serial 0/0
R2(config-if)#encapsulation frame-relay
R2(config-if)#ip address 10.1.1.2 255.255.255.0
R2(config-if)#frame-relay lmi-type cisco
```

FRSW 查看建立的 PVC，各接口与 DLCI 的对应关系，此时状态为 active.

```
FRSW#show frame-relay route
```

Input Intf	Input Dlci	Output Intf	Output Dlci	Status
Serial0/0	102	Serial0/1	201	active
Serial0/1	201	Serial0/0	102	active

R1 查看相关信息

```
R1#show frame-relay map
Serial0/0 (up): ip 10.1.1.2 dlci 102(0x66,0x1860), dynamic,
                broadcast,
                CISCO, status defined, active
```

R2 查看相关信息

```
R2#show frame-relay map
Serial0/0 (up): ip 10.1.1.1 dlci 201(0xC9,0x3090), dynamic,
                broadcast,
                CISCO, status defined, active
```

4) frame-relay 静态配置

FRSW 的配置保持不变

R1 :

```
R1(config)#interface searial 0/0
R1(config-if)#no frame-relay inverse-arp
R1(config-if)#frame-relay map ip 10.1.1.2 102 broadcast
```

// 通过手动指定的方式形成 IP 和 DLCI 的对应关系 由于 FR 是 NBMA 网络环境 故加上 broadcast 参数。

R2 :

```
R2(config)#interface searial 0/0
R2(config-if)#no frame-relay inverse-arp
R2(config-if)#frame-relay map ip 10.1.1.1 201 broadcast
R2(config-if)#no shutdown
```

FRSW 查看相关信息

```
FRSW#show frame-relay route
```

Input Intf	Input DlcI	Output Intf	Output DlcI	Status
Serial0/0	102	Serial0/1	201	active
Serial0/1	201	Serial0/0	102	active

R1 查看相关信息

```
R1#show frame-relay map
Serial0/0 (up): ip 10.1.1.2 dlci 102(0x66,0x1860), static,
                broadcast,
                CISCO, status defined, active
```

R2 查看相关信息

```
R2#show frame-relay map
```

```
Serial0/0 (up): ip 10.1.1.1 dlci 201(0xC9,0x3090), static,  
broadcast,  
CISCO, status defined, active
```

结论 2 查看 DTE 端的 MAP ,R1 上去往 10.1.1.2 的流量从 SERIAL 1/0 接口对应的 DLCI 102 的 PVC 走 , static 表示该条目由静态指定生成 , broadcast 表示二层具有广播能力 , LMI 的封装类型为默认的 CISCO , 此条目状态为 active。

5) 配置帧中继点到点

FRSW 的配置保持不变

R1 :

```
R1(config)#default int s0/0  
//删除接口 S0/0 的所有配置  
R1(config)#int s0/0  
R1(config-if)#encapsulation frame-relay  
R1(config-if)#no shutdown  
R1(config-if)#exit  
R1(config)#int s0/0.1 point-to-point  
R1(config-subif)#ip address 10.1.1.1 255.255.255.0  
//在子接口下指定 DLCI 值  
R1(config-subif)#frame-relay interface-dlci 102
```

R2 :

```
R2(config)#default int s0/0  
R2(config)#int s0/0  
R2(config-if)#encapsulation frame-relay  
R2(config-if)#no shutdown  
R2(config-if)#exit  
R2(config)#int s0/0.1 point-to-point  
R2(config-subif)#ip address 10.1.1.2 255.255.255.0  
R2(config-subif)#frame-relay interface-dlci 201  
//在子接口下指定 DLCI 值
```

查看 R1 路由器上的 frame-relay map

```
R1#show frame-relay map  
Serial0/0.1 (up): point-to-point dlci, dlci 102(0x66,0x1860), broadcast  
status defined, active
```

查看 R2 路由器上的 frame-relay map

```
R2#show frame-relay map  
Serial0/0.1 (up): point-to-point dlci, dlci 201(0xC9,0x3090), broadcast  
status defined, active
```

测试：

```
R1#ping 10.1.1.2  
  
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 10.1.1.2, timeout is 2 seconds:  
!!!!  
Success rate is 100 percent (5/5), round-trip min/avg/max = 80/104/128 ms
```

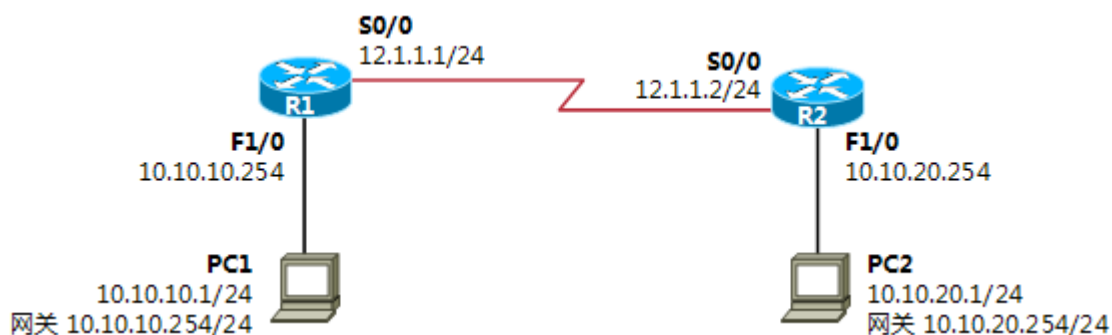
4. 实验总结

5.2 实验 2-PPP 实验

1. 实验目的：

- 1) 掌握 WAN 接口的 PPP 封装。掌握 PAP 和 CHAP 认证方式。

2. 实验拓扑和需求：



背景：某公司总部和分部网络通过 2M WAN 专线互连。

需求：现网中为了安全，对于接入 WAN 的路由器要求认证。常用的为 PAP 认证。

3. 主要步骤和相关配置

1) 关键命令

PPP PAP

```
Router(config)#username x password x
// 定义 PAP 认证所需的用户名及密码

Router(config)#interface serial 0/0
Router(config-if)#encapsulation ppp
Router(config-if)#ppp authentication pap
// 接口上启用 PAP 认证

Router(config-if)#ppp pap sent-username x password x
// 在接口上发送 PAP 认证所需的用户名和密码
```

PPP CHAP

```
Router(config)#username x password x
// 定义 CHAP 认证所需的用户名及密码

Router(config)#interface serial 0/0
Router(config-if)#encapsulation ppp
```

```
Router(config-if)#ppp authentication chap
```

```
// 接口上启用 chap 认证
```

2) 基本配置（略）。路由器模拟 PC 机，为路由器接口配 IP 和静态路由，实现两 PC 正常通信。

3) PAP 认证

R1 :

```
R1(config)#interface searial 0/0
R1(config-if)#encapsulation ppp
R1(config-if)#ppp authentication pap
R1(config-if)#ppp pap sent-username R1 password SPOTO
R1(config-if)#exit
R1(config)#username R2 password SPOTO
R1#debug ppp authentication
//debug PPP 的认证过程
```

R2 :

```
R2(config)#int serial 0/0
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication pap
R2(config-if)#ppp pap sent-username R2 password SPOTO
R2(config-if)#exit
R2(config)#username R1 password SPOTO
```

查看 R1debug 消息 :

```
*Mar  1 00:06:49.359: Se0/0 PPP: Authorization required
*Mar  1 00:06:49.415: Se0/0 PAP: Using hostname from interface PAP
*Mar  1 00:06:49.415: Se0/0 PAP: Using password from interface PAP
*Mar  1 00:06:49.415: Se0/0 PAP: O AUTH-REQ id 5 len 13 from "R1"
*Mar  1 00:06:49.419: Se0/0 PAP: I AUTH-REQ id 5 len 13 from "R2"
*Mar  1 00:06:49.419: Se0/0 PAP: Authenticating peer R2
*Mar  1 00:06:49.423: Se0/0 PPP: Sent PAP LOGIN Request
*Mar  1 00:06:49.427: Se0/0 PPP: Received LOGIN Response PASS
*Mar  1 00:06:49.431: Se0/0 PPP: Sent LCP AUTHOR Request
*Mar  1 00:06:49.435: Se0/0 PPP: Sent IPCP AUTHOR Request
*Mar  1 00:06:49.439: Se0/0 LCP: Received AAA AUTHOR Response PASS
R1#
*Mar  1 00:06:49.443: Se0/0 IPCP: Received AAA AUTHOR Response PASS
```

```
*Mar 1 00:06:49.443: Se0/0 PAP: O AUTH-ACK id 5 len 5
*Mar 1 00:06:49.495: Se0/0 PAP: I AUTH-ACK id 5 len 5
*Mar 1 00:06:49.499: Se0/0 PPP: Sent CDPCP AUTHOR Request
*Mar 1 00:06:49.503: Se0/0 CDPCP: Received AAA AUTHOR Response PASS
*Mar 1 00:06:49.527: Se0/0 PPP: Sent IPCP AUTHOR Request
```

4) CHAP 认证

R1 :

```
R1(config)#interface searial 0/0
R1(config-if)#no ppp authentication pap
R1(config-if)#ppp authentication chap
```

R2 :

```
R2(config)#interface serial 0/0
R2(config-if)#no ppp authentication pap
R2(config-if)#ppp authentication chap
R2(config-if)#shutdown
R2(config-if)#

*Mar 1 00:09:14.019: %LINK-5-CHANGED: Interface Serial0/0, changed state to
administratively down
*Mar 1 00:09:15.019: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0,
changed state to down
R2(config-if)#no shutdown
```

查看 debug 消息

```
*Mar 1 00:09:31.027: Se0/0 PPP: Authorization required
*Mar 1 00:09:31.035: Se0/0 CHAP: O CHALLENGE id 1 len 23 from "R1"
*Mar 1 00:09:31.135: Se0/0 CHAP: I CHALLENGE id 1 len 23 from "R2"
*Mar 1 00:09:31.143: Se0/0 CHAP: Using hostname from unknown source
*Mar 1 00:09:31.143: Se0/0 CHAP: Using password from AAA
*Mar 1 00:09:31.143: Se0/0 CHAP: O RESPONSE id 1 len 23 from "R1"
*Mar 1 00:09:31.179: Se0/0 CHAP: I RESPONSE id 1 len 23 from "R2"
*Mar 1 00:09:31.179: Se0/0 CHAP: I SUCCESS id 1 len 4
*Mar 1 00:09:31.183: Se0/0 PPP: Sent CHAP LOGIN Request
*Mar 1 00:09:31.187: Se0/0 PPP: Received LOGIN Response PASS
R1(config)#
*Mar 1 00:09:31.195: Se0/0 PPP: Sent LCP AUTHOR Request
```

```
*Mar 1 00:09:31.195: Se0/0 PPP: Sent IPCP AUTHOR Request
*Mar 1 00:09:31.199: Se0/0 LCP: Received AAA AUTHOR Response PASS
*Mar 1 00:09:31.203: Se0/0 IPCP: Received AAA AUTHOR Response PASS
*Mar 1 00:09:31.203: Se0/0 CHAP: O SUCCESS id 1 len 4
*Mar 1 00:09:31.207: Se0/0 PPP: Sent CDPCP AUTHOR Request
*Mar 1 00:09:31.211: Se0/0 CDPCP: Received AAA AUTHOR Response PASS
*Mar 1 00:09:31.247: Se0/0 PPP: Sent IPCP AUTHOR Request
```

思考：PAP 以及 CHAP 启用单向认证。

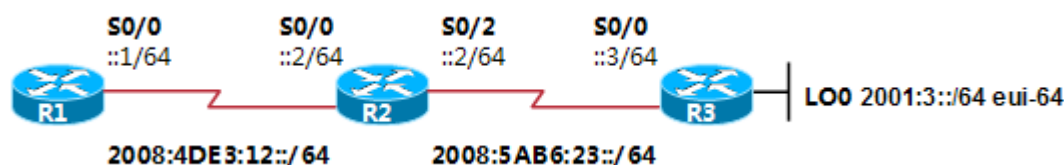
4. 实验总结

5.3 实验 3-IPv6 实验*

1. 实验目的：

- 1) 掌握 IPv6 的基本配置。

2. 实验拓扑和需求



需求：

- a. 在 R1 和 R3 上使用 IPv6 默认路由，R2 使用静态路由，实现全网互联；
- b. 删除静态路由，使用 RIPng 实现以上需求；
- c. 删除 RIPng，使用 OSPFv3 实现以上需求。

3. 主要步骤和相关配置

1) 静态路由

R1：

```

R1(config)#ipv6 unicast-routing      \启用 IPv6 数据包转发
R1(config)#int serial 0/0
R1(config-if)#ipv6 address 2008:4DE3:12::1/64
R1(config-if)#no shutdown
R1(config)#ipv6 route ::0 2008:4DE3:12::2 \配置默认路由
  
```

R2：

```

R2(config)#ipv6 unicast-routing
R2(config)#interface serial 0/0
R2(config-if)#ipv6 address 2008:4DE3:12::2/64
R2(config-if)#no shutdown
R2(config)#interface serial 0/1
R2(config-if)#ipv6 address 2008:5AB6:23::2/64
R2(config-if)#no shutdown
R2(config)#ipv6 route 2001:3::/64 2008:5AB6:23::3
  
```

R3:

```
R3(config)#ipv6 unicast-routing
R3(config)#interface serial 0/0
R3(config-if)#ipv6 address 2008:5AB6:23::3/64
R3(config-if)#no shutdown
R3(config)#interface loopback0
R3(config-if)#ipv6 address 2001:3::/64 eui-64
R3(config)#ipv6 route ::/0 2008:5AB6:23::2
```

查看 R3 接口

```
R3#show ipv6 interface brief
Loopback0                [up/up]
    FE80::200:CFF:FE31:C3B4
    2001:3::200:CFF:FE31:C3B4
Serial0/0                 [up/up]
    FE80::200:CFF:FE31:C3B4
    2008:5AB6:23::3
Serial0/1                 [administratively down/down]
Unassigned
```

查看 R1 路由表

```
R1#sh ipv6 route
IPv6 Routing Table - 5 entries
S   ::/0 [1/0]
    via 2008:4DE3:12::2, Null
C   2008:4DE3:12::/64 [0/0]
    via ::, Serial0
L   2008:4DE3:12::1/128 [0/0]
    via ::, Serial0
L   FE80::/10 [0/0]
    via ::, Null0
L   FF00::/8 [0/0]
    via ::, Null0
```

测试 1 :

R1#ping 2008:4DE3:12::2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 2008:4DE3:12::2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 28/30/32 ms

R1#ping 2001:3::200:CFF:FE31:C3B4

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 2001:3::200:CFF:FE31:C3B4, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 60/62/64 ms

2) RIPng 部分

R1 :

```
R1(config)#ipv6 router rip spoto
```

```
R1(config-rtr)#interface serial 0/0
```

```
R1(config-if)#ipv6 rip spoto enable    \\接口下宣告
```

R2:

```
R2(config)#ipv6 router rip spoto
```

```
R2(config-rtr)#interface serial 0/0
```

```
R2(config-if)#ipv6 rip spoto enable
```

```
R2(config-if)#interface searial 0/1
```

```
R2(config-if)#ipv6 rip spoto enable
```

R3 :

```
R3(config)#ipv6 router rip spoto
```

```
R3(config-rtr)#interface s0
```

```
R3(config-if)#ipv6 rip spoto enable
```

```
R3(config-if)#interface loop0
```

```
R3(config-if)#ipv6 rip spoto enable
```

Show

R1#show ipv6 rip database

```
RIP process "spoto", local RIB
```

```
2001:3::/64, metric 3, installed
```

```
    Serial0/FE80::200:CFF:FE75:AD10, expires in 166 secs
```

```
2008:4DE3:12::/64, metric 2
```

```
    Serial0/FE80::200:CFF:FE75:AD10, expires in 166 secs
```

```
2008:5AB6:23::/64, metric 2
```

```
Serial0/FE80::200:CFF:FE75:AD10, expires in 166 secs
```

R1#show ipv6 route rip

```
IPv6 Routing Table - 6 entries
```

```
R   2001:3::/64 [120/3]
    via FE80::200:CFF:FE75:AD10, Serial0/0
R   2008:5AB6:23::/64 [120/2]
    via FE80::200:CFF:FE75:AD10, Serial0/0
```

3) OSPFv3

R1 :

```
R1(config)#ipv6 router ospf 1
R1(config-rtr)#router-id 1.1.1.1    \\OSPFv3 的 router-id 同 OSPFv2 一样
R1(config-rtr)#interface s0/0
R1(config-if)#ipv6 ospf 1 area 0    \\接口下宣告 OSPFv3 , 并划分区域
```

R2 :

```
R2(config)#ipv6 router ospf 1
R2(config-rtr)#router-id 2.2.2.2
R2(config-rtr)#interface s0/0
R2(config-if)#ipv6 ospf 1 area 0
R2(config-rtr)#interface s0/1
R2(config-if)#ipv6 ospf 1 area 0
```

R3:

```
R3(config)#ipv6 router ospf 1
R3(config-rtr)#router-id 3.3.3.3
R3(config-rtr)#interface s0/0
R3(config-if)#ipv6 ospf 1 area 0
R3(config-rtr)#interface loop0
R3(config-if)#ipv6 ospf 1 area 0
```

Show Log

```
R2#show ipv6 ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Interface ID	Interface
3.3.3.3	1	FULL/ -	00:00:32	3	Serial0/1
1.1.1.1	1	FULL/ -	00:00:38	3	Serial0/0

```
R1#show ipv6 route ospf
```

IPv6 Routing Table - 7 entries

- 2001:3::200:CFF:FE8E:4301/128 [110/128]
via FE80::200:CFF:FE75:AD10, Serial0/0
- 2008:5AB6:23::/64 [110/128]
via FE80::200:CFF:FE75:AD10, Serial0/0

4. 实验总结

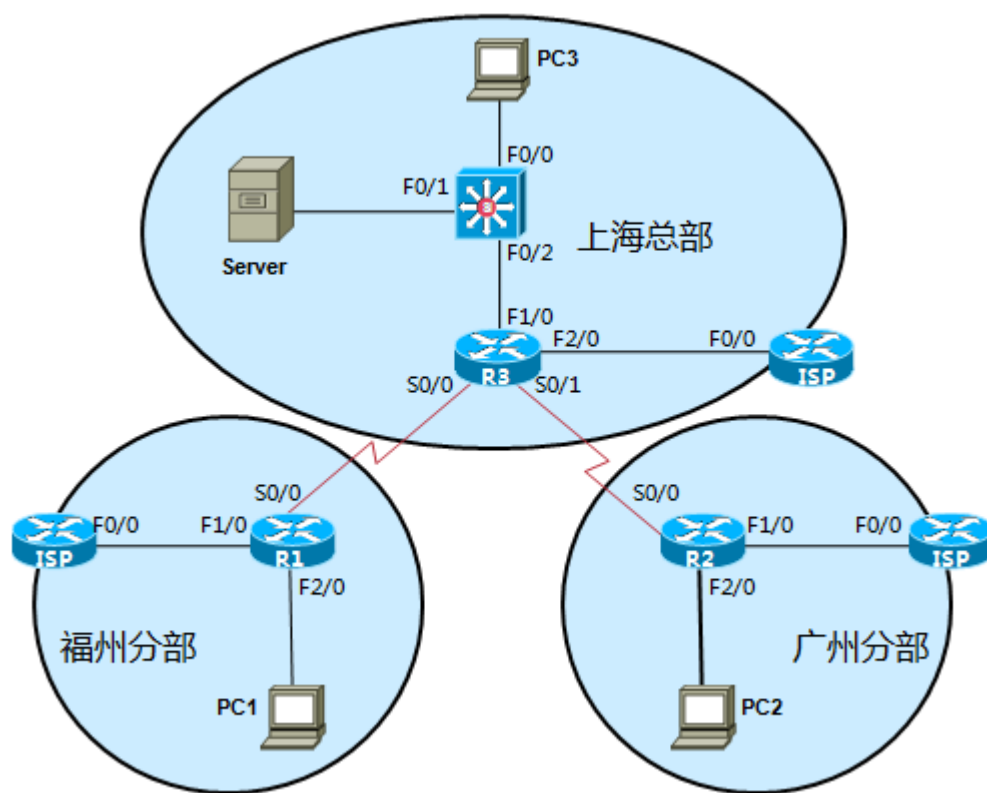
6 综合实验

6.1 中小型企业组网案例一

1. 实验目的：

- 1) 整合 CCNA 所学知识，独立完成模拟项目。

2. 实验拓扑和需求



背景：今天你作为 SPOTO 的网管，为 SPOTO 上海总部，福州和广州分部设计和规划网络。网络拓扑如上图所示。可用的 IP 地址为：192.168.10.0/24(私有 IP)，193.168.10.8/29 (公网 IP)。三地内网通过 WAN 线路进行互访，三地都有本地的 Internet 接入，上海总部的服务器对外开放 80 端口，其他分部不允许直接访问该内网服务器。

需求：

- 1) 三台 PC 模拟总部和分部的内网 LAN，内网通过 R1-R3-R2 的 WAN 线路互访。出于安全性考虑，总部的路由器 R3 的 serial 0/0 和 serial 0/1 接口需要启用单向 PAP 认证。
- 2) 三地组成小内网，启用动态路由协议 OSPF 实现互连，由于网络较小，故不需要划分多区域。

- 3) 为了保证办公人员正常上网，各 LAN 需要连入 Internet，可正常访问外网。
- 4) 上海总部为了存放重要数据，架设服务器 Server1，通过 TCP 映射 80 端口到外网供工作人员访问，为防止受非法攻击，禁止分部工作人员登录到服务器上修改数据，故服务器只允许 PC3 通过内网访问。

3. 实验主要步骤:

- 1) 规划内、外网 IP 地址。
- 2) 各设备的基本配置，预配的 IP 地址等。(需测试直连口的连通性)
- 3) R1,R2,R3 进行单向 PAP 认证的相关配置
- 4) 内网启用 OSPF 协议，观察 R1,R2,R3 的路由表的完整性，测试实现三台 PC 可互访。
- 5) 内网通过 PAT 实现内网工作人员可正常连入 Internet.
- 6) Server1 允许人们通过外网访问，使用静态 NAT 映射 80 端口到外网。(重点思考)
- 7) 利用命名 ACL，实现 Server1 除了 PC3 外其他网段均不能通过内网访问。测试访问结果。

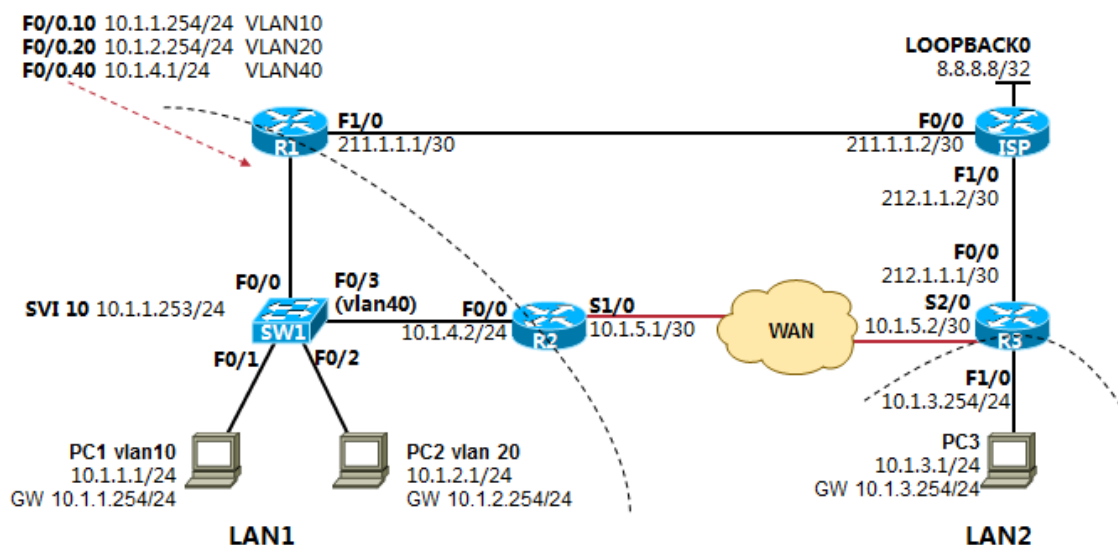
4. 实验总结

6.2 中小型企业组网案例二

1. 实验目的：

- 1) 根据 CCNA 所学知识点，独立完成以下模拟项目。

2. 实验拓扑和需求



需求:

- 1) R1 与 R3 分别为两个 LAN 连接 ISP 的出口路由器，内网通过 R2-R3 的 WAN 线路互访，出于安全性考虑，R2 的 serial 1/0 和 R3 的 searial 2/0 接口需要启用双向 CHAP 认证。
- 2) R1、R2 与 R3 之间使用静态路由互访。PC1、PC2 访问 ISP 通过 R1，不能通过 R3。PC3 访问 ISP 通过 R3，不允许通过 R2。
- 3) 只允许 PC1 与 PC3 互访，并且通过 WAN 链路，不允许 PC2 与 PC3 互访。

3. 实验主要步骤:

- 1) 规划内、外网 IP 地址。
- 2) 各设备的基本配置，预配的 IP 地址等，R2、R3 进行单向 CHAP 认证。(需测试直连口的连通性)
- 3) 使用静态路由，实现 PC1、PC2 与 PC3 之间的互访。
- 4) 内网通过 PAT 实现内网能访问到 ISP 得 8.8.8.8/32
- 5) 使用 ACL 实现 PC2 不能访问 PC3。

4. 实验总结