



百思学网络培训学校思科网络技术学院

## 思科网络工程师CCNA-CCNP认证培训

# 就业是硬道理 技术是保障

### 学校简介

廊坊百思学网络培训学校是一所民办职业技术培训学校，由廊坊开发区教育局批准成立办学许可证号为：1310057800012。我校现致力于计算机网络技术专业的教育培训，目前倾力推出思科网络工程师（CCNA-CCNP）就业班。

思科认证(CCNA/CCNP)是美国思科公司推出的一项职业认证，该认证在网络领域被公认为是最具权威性、含金量最高的认证。

### 课程描述

课程名程：CCNA-CCNP就业班（含计算机基础）

学习时间：（任选其一）

1、周一至周五（白天） 2、晚班（6:00-9:00） 3、周六、日（白天）

学 费：5800 元； 学 时：4-6个月，400 课时。

招生对象：高中、大中专在校生、有志于IT网络领域的社会青年，有无计算机基础均可。

开课时间：2009年3月18日（或每满20人随时开课）

### 推荐就业

就业方向：毕业生的工作方向是在专业的计算机网络公司、系统集成公司做售前或售后工程师，也可以在各企事业单位从事网络管理员的工作。

就业单位：我们推荐的单位主要是北京地区思科公司的合作伙伴，包括总经销商、金牌、银牌认证代理商，这些公司对通过CCNA/CCNP认证的学生有很大的需求。

就业案例：

冯 伟：现为CCNA 工资：4000元/月

百思学第六期CCNA班学员 北京财经学院2007届毕业生

单 位：丽视贸易（上海）有限公司 www.controlvc.com

刘镇意：现为CCNP 工资：3000元/月（试用期工资）

百思学第三期CCNP班学员，廊坊师范学院计算机专业2008届毕业生

单 位：北京南山之桥信息有限公司 www.nsbic.com

雷 霆：现为CCIE 工资 10000 元/月

单 位：中国数码信息有限公司 www.sino-i.com/home.htm

电话：0316-6028835 13722649808

地址：廊坊东方大学城一期教师公寓5号楼一单元401

网址：<http://www.bestxue.cn> E-mail: [tengshuwei@126.com](mailto:tengshuwei@126.com)

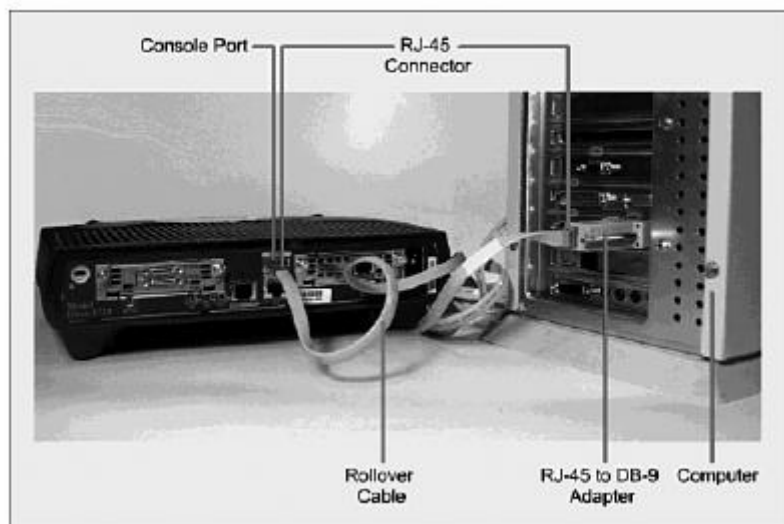
# 目 录

实验 1. 通过 CONSOLE 口控制路由器和交换机.....	-3-
实验 2. 熟悉路由器和交换机各种模式.....	-6-
实验 3. 配置一台路由器当做 PC.....	-7-
实验 4. 熟悉熟悉实验拓扑和基本配置.....	-8-
实验 5. CDP(思科发现协议).....	-14-
实验 6. 学会使用 Ping、Traceroute、Telnet、SSH.....	-19-
实验 7. CISCO 路由器，交换机密码恢复.....	-24-
实验 8. IOS 软件备份、升级与 NVRAM 备份.....	-26-
实验 9. 交换机 VLAN、TRUNK、VTP 配置.....	-33-
实验 10. 配置生成树（Spanning Tree）.....	-41-
实验 11. PVST+(每个 VLAN 的生成树 PVST 加).....	-45-
实验 12. Configuring Link Aggregation with EtherChannel.....	-51-
实验 13. 用 3 台交换机验证 PVST 实验.....	-52-
实验 14. 配置交换机的端口安全(1).....	-60-
实验 15. 配置交换机的端口安全(2).....	-62-
实验 16. 用路由器实现 VLAN 之间的访问.....	-64-
实验 17. 三层交换机实现 VLAN 之间的互访.....	-67-
实验 18. 静态路由.....	-70-
实验 19. RIP 配置.....	-72-
实验 20. 配置 EIGRP.....	-74-
实验 21. 配置单区域 OSPF.....	-75-
实验 22. 配置标准、扩展、命令 ACL 实验.....	-77-
实验 23. 静态 NAT 、动态 NAT、PAT 的配置.....	-80-
实验 24. PPP 的配置.....	-83-
实验 25. 配置帧中继.....	-84-

## 实验1. 通过 **CONSOLE** 口控制路由器和交换机

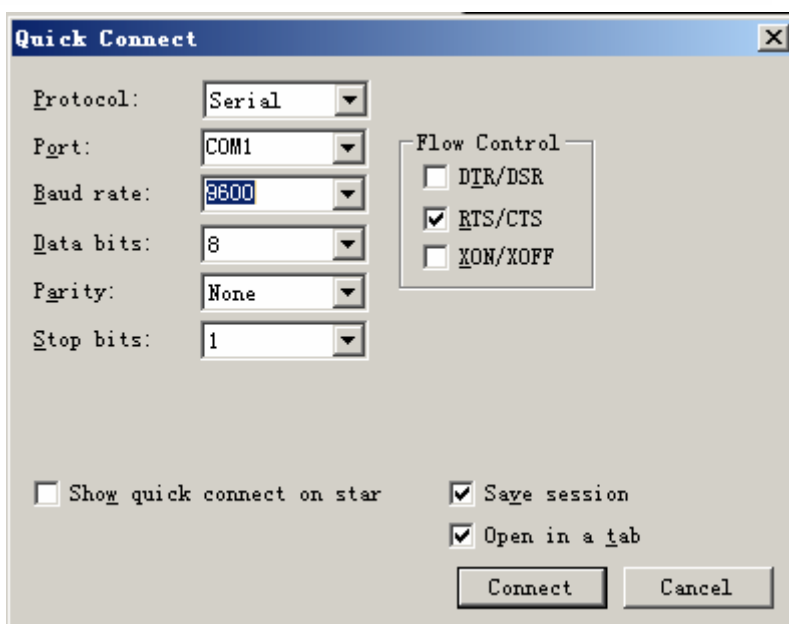
### 一、通过 console 口控制设备（图）

#### *Rollover Cable Connections*



### 1. 使用 SecureCRT（图）

#### *PC Settings to Connect to a Router or Switch*

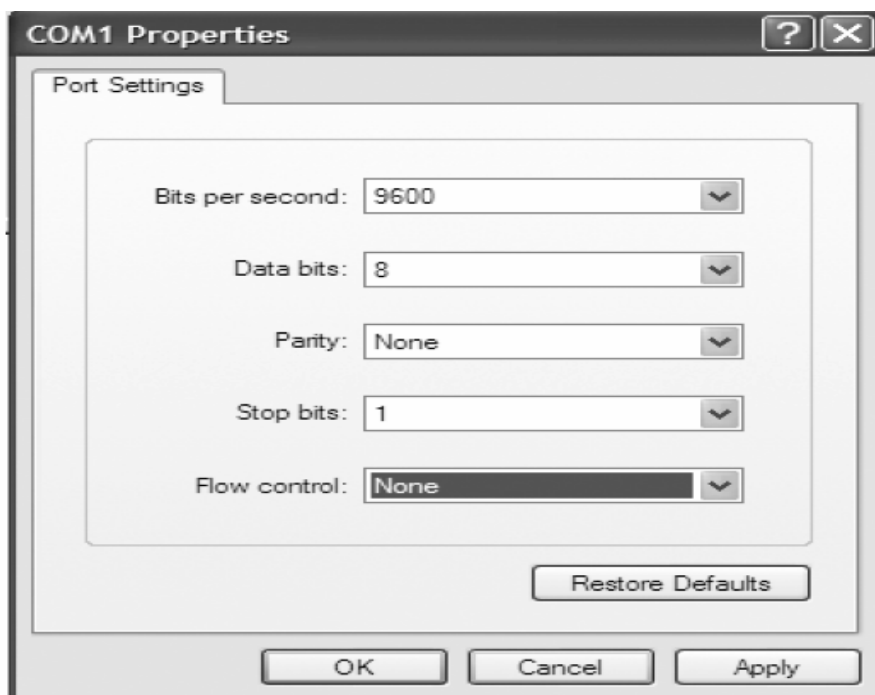


登录后界面显示:

```
Router#  
Router#show run  
Building configuration...  
  
Current configuration : 690 bytes  
!  
version 12.2  
service timestamps debug uptime  
service timestamps log uptime  
no service password-encryption  
!  
hostname Router  
!  
enable secret 5 $1$8GtA$RRjLnghAxMtYL9/C58JU2.  
!  
ip subnet-zero  
!  
!  
!  
ip audit notify log  
ip audit po max-events 100  
!  
call rsvp-sync  
!
```

## 2.使用 windows 的超级终端(图)

### *PC Settings to Connect to a Router or Switch*



3.如果是通过笔记本电脑且笔记本没有 com 口的情况下可以使用 usb-serial 的转接头（图）

*USB-to-Serial Connector for Laptops*



## 实验 2. 熟悉路由器和交换机各种模式

一、熟悉路由器和交换机各种模式（本实验用一台路由器就可以完成）

Router> //用户模式

Router# //特权模式（也叫 enable 模）

Router(config-if)# //接口模式

Router(config-router)# //router 模式

Router(config-line)# //line 模式

Router(config-subif)# //子接口模式

Router> //用户模式

Router>enable //在用户模式敲入 enable 进入特权模式（也叫 enable 模式）

Router#disable //在特权模式敲入 disable 退出到用户模式

Router>enable //在用户模式敲入 enable 进入特权模式

Router#configure terminal //在特权模式敲入 configure terminal 进入到配置模式

Router(config)#interface ethernet 0/0 //在配置模式敲入“interface+接口类型+接口编号”进入接口模式

Router(config-if)#exit //敲入 exit 退出接口模式

Router(config)#router rip //敲入“router + 路由协议”进入 router 模式

Router(config-router)#exit //退出 router 模式

Router(config)#line console 0 //进入 line 模式

Router(config-line)#end //从 line 模式退出（任何时候敲入 end 会退出到特权模式）

Router#conf t

Router(config)#interface ethernet 0/0.1 //进入子接口模式 Router(config-subif)#end//任何时候敲入 end 会退出到特权模式 Router#

注：有关路由器或交换机各种模式的应用后面的章节中会讲到，这里先认识一下就可以。

### 实验 3. 配置一台路由器当作 PC

做实验时通常用路由器当作 PC 使用，用来测试网络的连通性等。

步骤：（本例子中的当作 PC 的路由器接口为 Ethernet 0）

1. 关闭路由器的路由功能
2. 设置接口 IP 地址
3. 配置默认网关
4. 检查 Router#conf t Router(config)#host PC1

PC1(config)#no ip routing -----注：关闭路由器的路由功能

PC1(config)#interface ethernet 0

PC1(config-if)#ip address 11.1.1.4 255.255.255.0 -----注：设置接口 IP 地址并启用

PC1(config-if)#no shutdown

PC1(config-if)#exit

PC1(config)#ip default-gateway 11.1.1.1 -----注：配置默认网关

PC1(config)#end

PC1#show ip int brief

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0	11.1.1.4	YES	TFTP	up	up
Serial0	unassigned	YES	TFTP	administratively down	down
Serial1	unassigned	YES	TFTP	administratively down	down

PC1#show ip route

Default gateway is 11.1.1.1 -----注： Show ip route 可以看到网关

在路由器中关闭路由功能、配置 IP 地址、网关就可以作为 PC 来用了。



## 实验 4. 熟悉实验拓扑和基本配置

一、实验拓扑和各设备共同常用配置（可以复制再粘贴到设备中）

```
Enable
```

```
Conf t
```

```
Enable secret cisco //设置 enable 的密码为 cisco
```

```
Enable password cisco1 //设置 enable 的密码为 cisco1, 不能和 enable secret 设置的密码相同
```

```
No ip domain-lookup //关掉域名查找功能
```

```
service password-encryption //对口令进行加密, 加密 console 口或 VTY 或是 enable password 设置的密码, 密码不再显示明文
```

```
Line console 0 //设置 console 口
```

```
Password cisco //设置 console 口密码为 cisco
```

```
Login //启用密码
```

```
No exec-timeout //操作会话不会超时
```

```
logging synchronous //配置时光标跟随, 阻止那些烦人的控制台信息来打断你当前的输入, 从而使输入信息显得更为简单易读.
```

```
line vty 0 4 //设置 VTY
```

```
Password cisco
```

```
Login
```

```
No exec-timeout
```

```
logging synchronous
```

注: VTY（虚拟终端）在网络操作系统（包括 Cisco IOS）中接受 telnet 或 ssh 连接的一个逻辑接口。

<<粘贴版>>以下可以直接复制再粘贴到路由器或交换机

```
Enable
```

```
Conf t
```

```
Enable secret cisco
```

```
No ip domain-lookup
```

```
service password-encryption
```

```
Line console 0
```

```
Password cisco
```

```
Login
```

```
No exec-timeout
```

```
logging synchronous
```

```
line vty 0 4
```

```
Password cisco
```

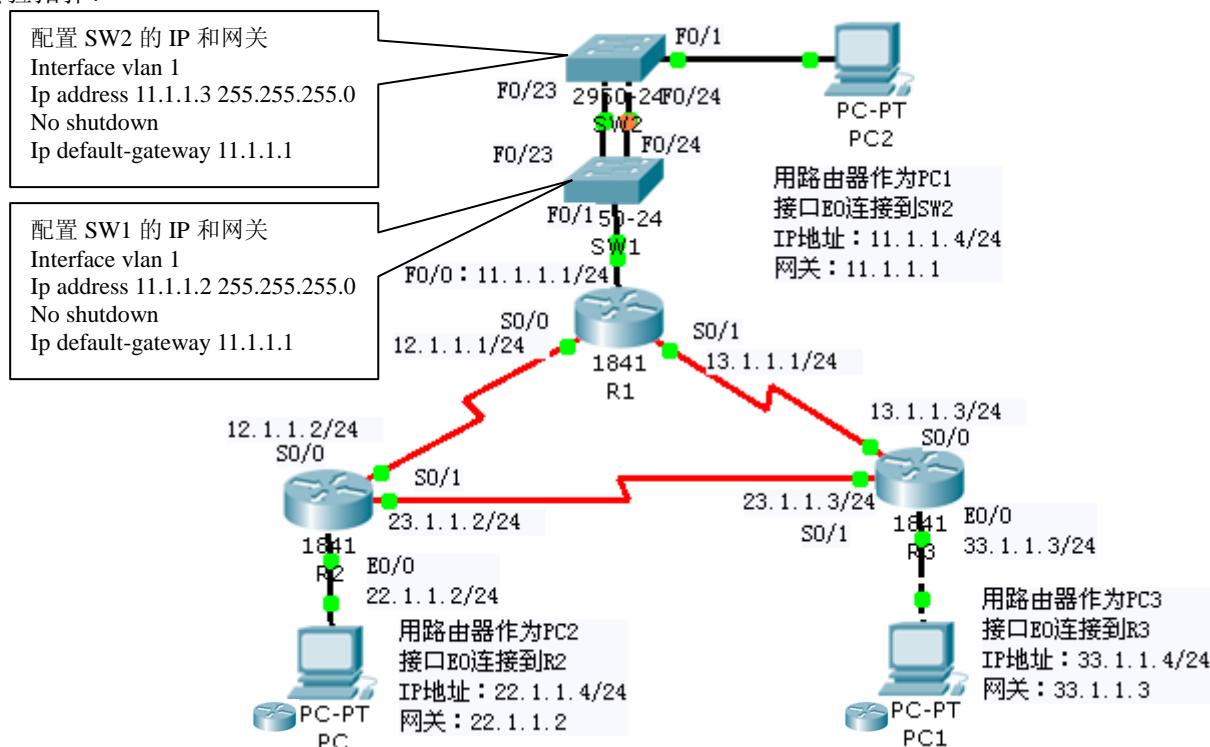
```
Login
```

```
No exec-timeout
```

```
logging synchronous
```



## 实验拓扑:



二、根据以上拓扑配置各设备的 IP 地址,其中 SW1, SW2, PC1, PC2, PC3 要设置网关。

### 1、配置 PC1

```
Router>enable
Router#conf t
Router(config)#no ip routing //关闭路由器的路由功能
Router(config)#hostname PC1 //命令主机名
PC1(config)#interface ethernet 0 //进入 ethernet 0 接口模式
PC1(config-if)#ip address 11.1.1.4 255.255.255.0 //配置 IP 地址
PC1(config-if)#no shutdown //启用接口
PC1(config-if)#exit //退出到配置模式
PC1(config)#ip default-gateway 11.1.1.1 //配置 PC1 的默认网关
PC1(config)#end //退出到 enable 模式
PC1#
```

### 2、配置 PC2

```
Router>enable
Router#conf t
Router(config)#no ip routing //关闭路由器的路由功能
Router(config)#hostname PC2 //命令主机名
PC2(config)#interface ethernet 0 //进入 ethernet 0 接口模式
PC2(config-if)#ip address 22.1.1.4 255.255.255.0 //配置 IP 地址
PC2(config-if)#no shutdown //启用接口
PC2(config-if)#exit //退出到配置模式
PC2(config)#ip default-gateway 22.1.1.2 //配置 PC1 的默认网关
PC2(config)#end //退出到 enable 模式
```

### 3、配置 PC3

```
Router>enable
```

```
Router#conf t
Router(config)#no ip routing      //关闭路由器的路由功能
Router(config)#hostname PC3      //命令主机名
PC3(config)#interface ethernet 0 //进入 ethernet 0 接口模式
PC3(config-if)#ip address 33.1.1.4 255.255.255.0 //配置 IP 地址
PC3(config-if)#no shutdown //启用接口
PC3(config-if)#exit //退出到配置模式
PC3(config)#ip default-gateway 33.1.1.3 //配置 PC1 的默认网关
PC3(config)#end //退出到 enable 模式
PC3#
```

#### 4、配置 R1

```
Router>enable
Router#conf t
Router(config)#hostname R1
R1(config)#int ethernet 0/0
R1(config-if)#ip add 11.1.1.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#int s0/0
R1(config-if)#ip add 12.1.1.1 255.255.255.0
R1(config-if)#clock rate 9600 //DCE 端要配置时钟速率
R1(config-if)#no shutdown
R1(config-if)#exit
R1(config)#int s0/1
R1(config-if)#ip add 13.1.1.1 255.255.255.0
R1(config-if)#clock rate 9600 //DCE 端要配置时钟速率
R1(config-if)#no shutdown
R1(config-if)#end
R1#
```

#### 5、配置 R2

```
Router>enable
Router#conf t
Router(config)#hostname R2
R2(config)#int ethernet 0/0
R2(config-if)#ip add 22.1.1.2 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#int s0/0
R2(config-if)#ip add 12.1.1.2 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#int s0/1
R2(config-if)#ip add 23.1.1.2 255.255.255.0
R2(config-if)#clock rate 9600 //DCE 端要配置时钟速率
R2(config-if)#no shutdown
R2(config-if)#end
```

R2#

## 6、配置 R3

Router>enable

Router#conf t

Router(config)#hostname R3

R3(config)#int ethernet 0/0

R3(config-if)#ip add 33.1.1.3 255.255.255.0

R3(config-if)#no shutdown

R3(config-if)#exit

R3(config)#int s0/0

R3(config-if)#ip add 13.1.1.3 255.255.255.0

R3(config-if)#no shutdown

R3(config-if)#exit

R3(config)#int s0/1

R3(config-if)#ip add 23.1.1.3 255.255.255.0

R3(config-if)#no shutdown

R3(config-if)#end

R3#

## 7、配置 SW1

Switch>enable

Switch#conf t

Switch(config)#host SW1

SW1(config)#interface vlan 1

SW1(config-if)#ip address 11.1.1.2 255.255.255.0

SW1(config-if)#no shutdown

SW1(config-if)#exit

SW1(config)#ip default-gateway 11.1.1.1

SW1(config)#end

SW1#

## 8、配置 SW2

Switch>enable

Switch#conf t

Switch(config)#host SW2

SW2(config)#interface vlan 1

SW2(config-if)#ip address 11.1.1.3 255.255.255.0

SW2(config-if)#no shutdown

SW2(config-if)#exit

SW2(config)#ip default-gateway 11.1.1.1

SW2(config)#end

SW2#

三、验证拓扑中各设备的配置是否正确和设备的接口状态（up 还是 down）

### 1、验证 PC1

PC1#show ip int brief //查看本路由器各接口基本配置, 包括 IP 地址, 接口状态等

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0	11.1.1.4	YES	manual	up	up
Serial0	unassigned	YES	unset	administratively down	down
Serial1	unassigned	YES	unset	administratively down	down

PC1#show ip route //在 PC1 上查看设置的默认网关

Default gateway is 11.1.1.1

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

ICMP redirect cache is empty

PC1#

## 2、验证 PC2

PC2#**show ip int brief** //查看本路由器各接口基本配置, 包括 IP 地址, 接口状态等

Interface	IP-Address	OK?	Method	Status	Protocol
<b>Ethernet0</b>	<b>22.1.1.4</b>	<b>YES</b>	<b>manual</b>	<b>up</b>	<b>up</b>
Serial0	unassigned	YES	unset	administratively down	down
Serial1	unassigned	YES	unset	administratively down	down

PC2#**show ip route** //在 PC2 上查看设置的默认网关

**Default gateway is 22.1.1.2**

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

ICMP redirect cache is empty

PC2#

## 3、验证 PC3

PC3#**show ip int brief** //查看本路由器各接口基本配置, 包括 IP 地址, 接口状态等

Interface	IP-Address	OK?	Method	Status	Protocol
<b>Ethernet0</b>	<b>33.1.1.4</b>	<b>YES</b>	<b>manual</b>	<b>up</b>	<b>up</b>
Serial0	unassigned	YES	unset	administratively down	down
Serial1	unassigned	YES	unset	administratively down	down

PC3#**show ip route** //在 PC3 上查看设置的默认网关

**Default gateway is 33.1.1.3**

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

ICMP redirect cache is empty

PC3#

## 4、验证 R1

R1#**show ip int brief**

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0/0	11.1.1.1	YES	manual	up	up
Serial0/0	12.1.1.1	YES	manual	up	up
Ethernet0/1	unassigned	YES	unset	administratively down	down
Serial0/1	13.1.1.1	YES	manual	up	up

R1#

## 5、验证 R2

R2#**show ip int brief**

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0/0	22.1.1.2	YES	manual	up	up
Serial0/0	12.1.1.2	YES	manual	up	up
Ethernet0/1	unassigned	YES	unset	administratively down	down
Serial0/1	23.1.1.2	YES	manual	up	up

## 6、验证 R3

R3#**show ip int brief**

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0/0	33.1.1.3	YES	manual	up	up
Serial0/0	13.1.1.3	YES	NVRAM	up	up
Ethernet0/1	unassigned	YES	NVRAM	administratively down	down
Serial0/1	23.1.1.3	YES	manual	up	up

## 四、用 ping 命令测试连通性

### 1、测试 PC1 到 R1, SW1, SW2 的连通性

PC1#ping 11.1.1.1 //测试 PC1 到 R1 的连通性

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 11.1.1.1, timeout is 2 seconds:

!!!!

-----注：!!!!表示连通性正常

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms

PC1#ping 11.1.1.2 //测试 PC1 到 SW1 的连通性

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 11.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms

PC1#

PC1#ping 11.1.1.3 //测试 PC1 到 SW2 的连通性

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 11.1.1.3, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms

2、测试 R1 到 R2，R3 的连通性

R1#ping 12.1.1.2 //测试 R1 到 R2 的连通性

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 12.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms

R1#

R1#ping 13.1.1.3 //测试 R1 到 R3 的连通性

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 13.1.1.3, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/4 ms

3、测试 R2 到 R3 的连通性

R2#ping 23.1.1.3

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), round-trip min/avg/max = 1/2/4 ms

R2#

4、测试 PC2 到 R2 的连通性

PC2#ping 22.1.1.2

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 22.1.1.2, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

PC2#

5、测试 PC3 到 R3 的连通性

PC3#ping 33.1.1.3

Type escape sequence to abort.

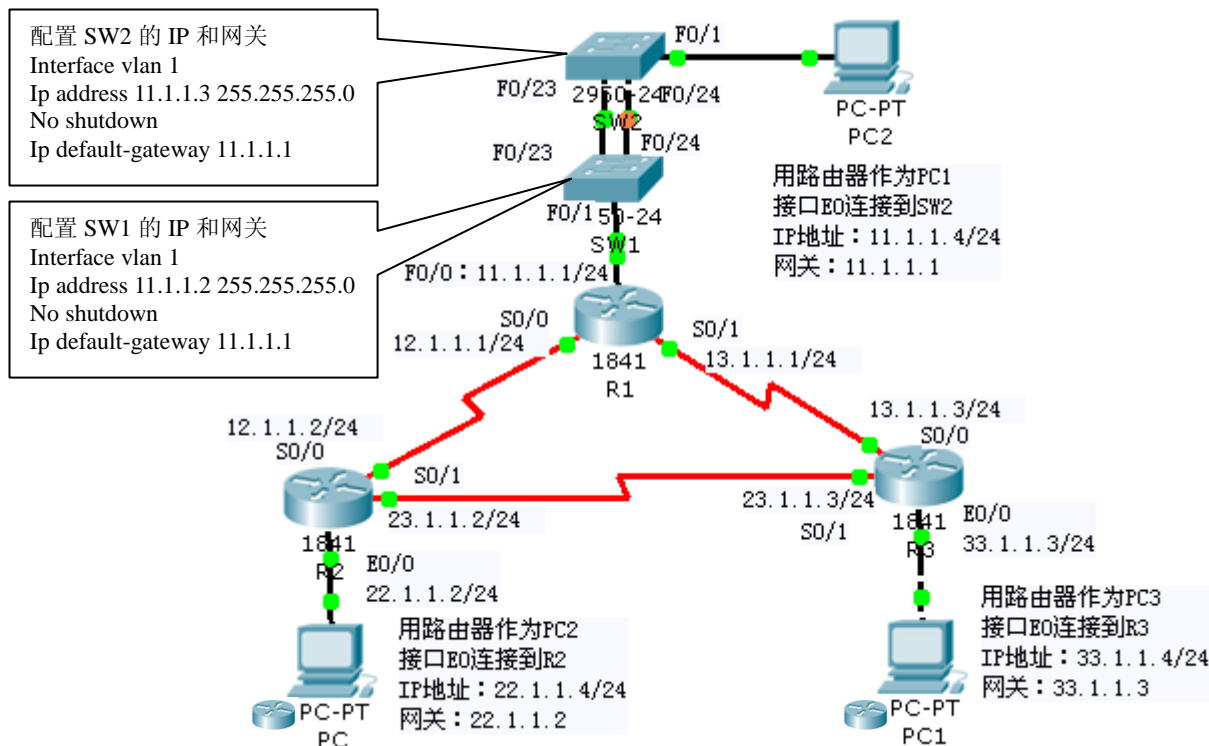
Sending 5, 100-byte ICMP Echos to 33.1.1.3, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms

## 实验 5. CDP (思科发现协议)

CDP 是 CISCO 私有的协议，CDP 对排错或实验有很大帮助。注：在做 CDP 实验时用 **R1,SW1,SW2** 三台设备就可以验证！



一. 配置接口 IP 地址和启用接口。

### 1 · 配置 R1

```
Router# Router#conf t
Router(config)#hostname R1
R1(config)#int ethernet 0/0
R1(config-if)#ip add 11.1.1.1 255.255.255.0
R1(config-if)#no shutdown
R1(config-if)#exit
```

### 2 · 配置 SW1

```
Switch# Switch#conf t
Switch(config)#host SW1
SW1(config)#int vlan 1
SW1(config-if)#ip add 11.1.1.2 255.255.255.0
SW1(config-if)#no sh
SW1(config-if)#end
```

### 3 · 配置 SW2

```
Switch# Switch#conf t
Switch(config)#host SW2
SW2(config)#int vlan 1
SW2(config-if)#ip add 11.1.1.3 255.255.255.0
SW2(config-if)#no sh
SW2(config-if)#end
```

配置完 R1, SW1, SW2 的接口 IP 和启用接口后用以下命令确认-包括所设置的 IP 地址是否正确, 接口状态是否为 up

确认 **R1:**

R1#show ip int brief

Interface	IP-Address	OK?	Method	Status	Protocol
Ethernet0/0	11.1.1.1	YES	manual	up	up

SW1#show int vlan 1

Vlan1 is up, line protocol is up

Hardware is CPU Interface, address is 0008.20ff.6400 (bia 0008.20ff.6400) Internet address is 11.1.1.2/24

.....

确认 **SW1:**

SW1#show ip int brief

Interface	IP-Address	OK?	Method	Status	Protocol	Vlan1
	11.1.1.2	YES	manual	up	up	
FastEthernet0/1	unassigned	YES	unset	up	up	

.....

FastEthernet0/23	unassigned	YES	unset	up	up
FastEthernet0/24	unassigned	YES	unset	up	up

确认 **SW2:**

SW2#show ip int brief

Interface	IP-Address	OK?	Method	Status	Protocol
Vlan1	11.1.1.3	YES	manual	up	up
.....					
FastEthernet0/23	unassigned	YES	unset	up	up
FastEthernet0/24	unassigned	YES	unset	up	up

## 二. 验证 CDP

### 1. 在 R1 上验证 CDP

R1#show cdp ?

```

entry      Information for specific neighbor entry
interface  CDP interface status and configuration
neighbors  CDP neighbor entries
traffic    CDP statistics
|          Output modifiers
<cr>
```

R1#show cdp interface //显示打开 CDP 协议的接口

Ethernet0/0 is up, line protocol is up //显示接口状态

Encapsulation ARPA

Sending CDP packets every 60 seconds //60 秒为周期发送 CDP 的包

Holdtime is 180 seconds //CDP 的包保持时间为 180 秒

Serial0/0 is administratively down, line protocol is down

Encapsulation HDLC

Sending CDP packets every 60 seconds

Holdtime is 180 seconds

Serial0/1 is administratively down, line protocol is down



Encapsulation HDLC

Sending CDP packets every 60 seconds

Holdtime is 180 seconds

以上通过 show cdp interface 发现 R1 有 E0/0 , S0/0 , S0/1 共 3 个接口打开了 CDP 协议.

R1#show cdp neighbors //显示 CDP 邻居

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge

S - Switch, H - Host, I - IGMP, r - Repeater

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
SW1	Eth 0/0	171	S I	WS-C2950	0/1
设备名字	本地接口	保持时间	Switch IGMP	型号	对端接口

R1#show cdp entry \* //显示所有邻居详细信息, 包括对方接口 IP 地址和 IOS 软件版本-----  
----- Device ID: SW1

Entry address(es):IP address: **11.1.1.2**

Platform: **cisco WS-C2950-24**, Capabilities: **Switch IGMP**

Interface: **Ethernet0/0**, Port ID (outgoing port): **FastEthernet0/1**

Holdtime : **163 sec**

Version :

Cisco Internetwork Operating System Software

IOS (tm) C2950 Software (C2950-I6K2L2Q4-M), **Version 12.1(22)EA10a**, RELEASE SOFTWARE (fc2) Copyright (c) 1986-2007 by cisco Systems, Inc.

Compiled Tue 24-Jul-07 17:37 by antonino

R1#show cdp traffic //显示 CDP 包, 出和进入的包数量

CDP counters :

**Total packets output: 12, Input: 11**

Hdr syntax: 0, Chksum error: 0, Encaps failed: 0

No memory: 0, Invalid packet: 0, Fragmented: 0

CDP version 1 advertisements output: 0, Input: 0

**CDP version 2 advertisements output: 12, Input: 11**

## 2. 在 SW1 上验证 CDP

SW1#show cdp ?

entry	Information for specific neighbor entry
interface	CDP interface status and configuration
neighbors	CDP neighbor entries
traffic	CDP statistics
	Output modifiers
<cr>	

SW1#show cdp interface

FastEthernet0/1 is up, line protocol is up

Encapsulation ARPA

Sending CDP packets every 60 seconds

Holdtime is 180 seconds

FastEthernet0/2 is down, line protocol is down

Encapsulation ARPA

Sending CDP packets every 60 seconds

Holdtime is 180 seconds

FastEthernet0/3 is down, line protocol is down

Encapsulation ARPA

Sending CDP packets every 60 seconds  
Holdtime is 180 seconds

.....

.....

SW1#show cdp neighbors

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge

S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
SW2	Fas 0/24	143	S I	WS-C2950-2Fas	0/24
SW2	Fas 0/23	143	S I	WS-C2950-2Fas	0/23
R1	Fas 0/1	176	R S I	2610	Eth 0/0

SW1#show cdp entry

\* Device ID: SW2

Entry address(es): IP address: 11.1.1.3

Platform: cisco WS-C2950-24, Capabilities: Switch IGMP

Interface: FastEthernet0/24, Port ID (outgoing port): FastEthernet0/24

Holdtime : 129 sec

Version :

Cisco Internetwork Operating System Software

IOS (tm) C2950 Software (C2950-I6K2L2Q4-M), Version 12.1(22)EA10a, RELEASE SOFTWARE (fc2) Copyright (c) 1986-2007 by cisco Systems, Inc.

Compiled Tue 24-Jul-07 17:37 by antonino----- Device ID: SW2

Entry address(es): IP address: 11.1.1.3

Platform: cisco WS-C2950-24, Capabilities: Switch IGMP

Interface: FastEthernet0/23, Port ID (outgoing port): FastEthernet0/23

Holdtime : 128 sec

Version :

Cisco Internetwork Operating System Software

IOS (tm)

C2950 Software (C2950-I6K2L2Q4-M), Version 12.1(22)EA10a, RELEASE SOFTWARE (fc2) Copyright (c) 1986-2007 by cisco Systems, Inc.

Compiled Tue 24-Jul-07 17:37 by antonino

Device ID: R1

Entry address(es):

IP address: 11.1.1.1

Platform: Cisco 2600, Capabilities: Router Switch IGMP

Interface: FastEthernet0/1, Port ID (outgoing port): Ethernet0/0

Holdtime : 160 sec

Version :

Cisco IOS Software, 3600 Software (C2600-JK9O3S-M), Version 12.4(5), RELEASE SOFTWARE (fc3) Technical Support: <http://www.cisco.com/techsupport>

Copyright (c) 1986-2005 by Cisco Systems, Inc. Compiled

Mon 31-Oct-05 21:24 by alnguyen SW1#show cdp traffic

CDP counters :

Total packets output: 223, Input: 200

Hdr syntax: 0, Chksum error: 0, Encaps failed: 0

No memory: 0, Invalid packet: 0, Fragmented: 0

CDP version 1 advertisements output: 0, Input: 0

CDP version 2 advertisements output: 223, Input: 200

### 3. 在 SW2 上验证 CDP

SW2#show cdp neighbors

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge

S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
SW1	Fas 0/24	165	S I	WS-C2950-2Fas	0/24
SW1	Fas 0/23	165	S I	WS-C2950-2Fas	0/23
SW2#					

#### 4. 改变 CDP 发包(sending)周期和保持(holdtime)时间

1. 默认时间为: R1#show cdp

interface

Ethernet0/0 is up, line protocol is up

Encapsulation ARPA

Sending CDP packets every 60 seconds

Holdtime is 180 seconds

2. 改变 CDP 发包和保持时间

R1(config)#cdp timer 5 //周期为 5 秒

R1(config)#cdp holdtime 15 //保持时间为 15 秒

R1#show cdp interface

Ethernet0/0 is up, line protocol is up

Encapsulation ARPA

Sending CDP packets every 5 seconds

Holdtime is 15 seconds

3. 关闭 CDP

R1(config)#no cdp run //全局关闭 CDP (意味着所有接口关闭了 CDP 协议)

R1(config)#end

R1#show cdp interface

% CDP is not enabled

R1#

在全局上启用了 cdp run 后可以在指定接口关闭 CDP 协议.

R1(config)#cdp run

R1(config)#int e0/0

R1(config-if)#no cdp enable //在接口模式关闭 CDP

R1(config-if)#end

R1#show cdp interface //在接口 E0/0 关闭了 CDP 后没有发现 E0/0 接口相关参数

Serial0/0 is administratively down, line protocol is down

Encapsulation HDLC

Sending CDP packets every 5 seconds

Holdtime is 15 seconds

Serial0/1 is administratively down, line protocol is down

Encapsulation HDLC

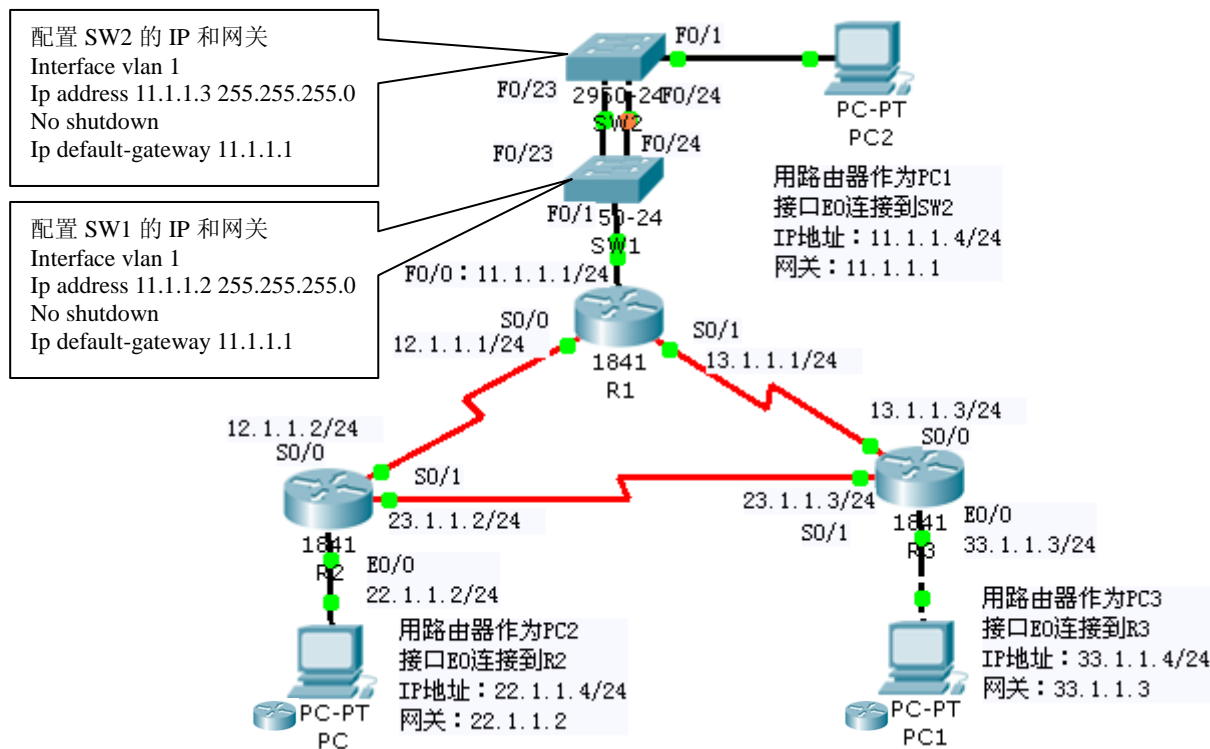
Sending CDP packets every 5 seconds

Holdtime is 15 seconds

## 实验6. 学会使用 Ping、Traceroute、Telnet、SSH (测试和网络连接的小工具) 功能

描述:

- ☑ 使用 Ping 命令是用于检测网络设备可达性,它使用 ICMP 有 echo 信息来决定远程设备是否可用,与远程主机通信来回的延迟 delay,或是数据包的丢失情况.
- ☑ 使用 Traceroute 跟踪从源到目的地所经过的路径.
- ☑ 使得 telnet , ssh (Secure Shell)远程控制设备.



### 1. 使用 PING 命令

注: 本实验使用 R1, R2, R3, SW1 四台设备测试, 配置好 R1, R2, R3, SW1 的 IP 地址, 检查接口状态为 up 后再测试。

在 R1 上测试到 R2, R3 的连通性

**R1#ping 12.1.1.2**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 12.1.1.2, timeout is 2 seconds:

!!!! //5 个!表明网络是通的

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/8 ms

**R1#ping 13.1.1.3**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 13.1.1.3, timeout is 2 seconds:

!!!! //5 个!表明网络是通的

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/8 ms

R1#

在 SW1 上检查到 R1 的连通性：

**SW1#ping 11.1.1.1**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 11.1.1.1, timeout is 2 seconds:

**!!!!** //5 个!表明网络是通的

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/8 ms

SW1#

通常在路由器或是交换机里使用 ping 命令时常碰到的问题。

叹号(!):表示成功收到响应信息.

句号(.):表示在等待应答超时.

U:表示目标主机不可达.

Q:表示目标主机繁忙.

M:代表不能分片(fragment).

?:表示未知数据包类型.

& :表示数据包已没有生存期(lifetime)

2. 使用 **tracert** 命令, 检查从源到目的地经过的路由器的个数. (或是数据包经过的路径)

注: 本实验先确认 R2 到 33.1.1.3 是否有路由, 和 R3 是否有到 12.1.1.2 的路由, 可以用 show ip route 命令 检查, 如果没有, 分别在 R1,R2, R3 上配置路由:

**R1: ip route 33.1.1.0 255.255.255.0 13.1.1.3**

**R2: ip route 0.0.0.0 0.0.0.0 12.1.1.1**

**R3: ip route 0.0.0.0 0.0.0.0 13.1.1.1**

在 R2 上使用 **tracert** 命令 :

**R2#tracert 33.1.1.3**

Type escape sequence to abort. Tracing

the route to 33.1.1.3

1 **12.1.1.1** 4 msec 4 msec 4 msec

2 **13.1.1.3** 4 msec 4 msec \*

从以上可以看出从 R2 到达 33.1.1.3 经过了 12.1.1.1 和 13.1.1.3 两个路由器

3· 在 PC 上使用的是 **tracert** 命令 (路由器或是交换机使用的是 **tracert**):

例如:跟踪从 PC 到厦门电信 DNS (202.101.103.55) 所经过的路径 (所经过的路由器个数, 相应 IP 地址)

```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 5.2.3790]
(C) 版权所有 1985-2003 Microsoft Corp.

C:\Documents and Settings\Administrator>tracert 202.101.103.55

Tracing route to dns.xm.fj.cn [202.101.103.55]
over a maximum of 30 hops:

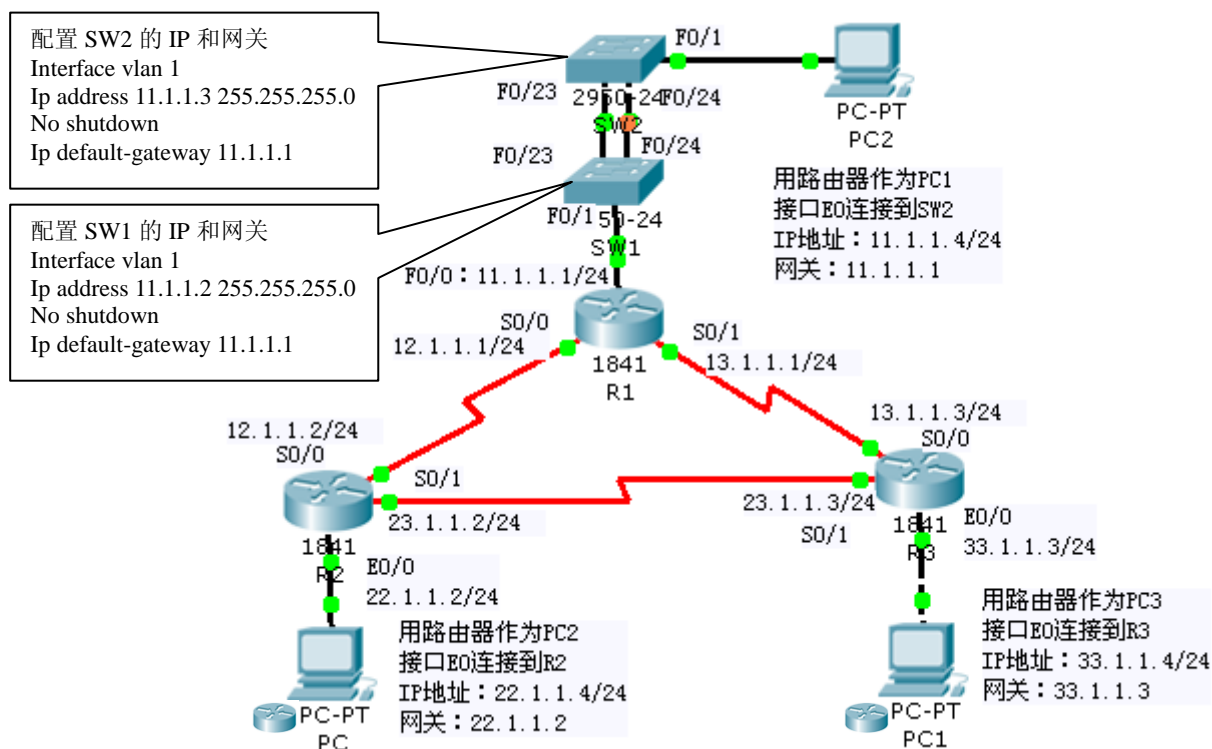
  0  10 ms  <1 ms  <1 ms  OFFICE [192.168.1.252]
  1  1 ms  <1 ms  <1 ms  172.16.1.1
  2  19 ms  9 ms  9 ms  1.100.160.220.board.xm.fj.dynamic.163data.com.cn
  3  9 ms  9 ms  9 ms  220.160.100.1
  4  9 ms  9 ms  9 ms  218.85.151.117
  5  11 ms  8 ms  9 ms  61.154.236.106
  6  8 ms  8 ms  8 ms  dns.xm.fj.cn [202.101.103.55]

Trace complete.

C:\Documents and Settings\Administrator>

```

二、使用 **telnet** 远程控制远端设备，（包括路由器或是交换机或是防火墙等网络设备）。  
Telnet 是基于 TCP 协议上的, 端口号为 23



实验目的:通过控制 R1 后，TELNET 到 R2(12.1.1.2), R3(13.1.1.3) 对 R2, R3 进行远程管理。

在 R1 上配置:

R1#telnet 12.1.1.2 //从 R1 TELNET 到 R2

Trying 12.1.1.2 ... Open User

Access Verification Password:

R2>

按<Ctrl-Shift-6> + x 把界面切换回 R1

R1#telnet 13.1.1.3 //从 R1 TELNET 到 R3

Trying 13.1.1.3 ... Open User

Access Verification Password:

R3>

按<Ctrl-Shift-6> + x 把界面切换回 R1

R1#show sessions //在 R1 上查看打开了多少个 TELNET 会话

Conn	Host	Address	Byte	Idle	Conn Name
1	12.1.1.2	12.1.1.2	0	0	12.1.1.2
* 2	13.1.1.3	13.1.1.3	0	0	13.1.1.3

R1#resume 2 //重新连接到 R3

[Resuming connection 2 to 13.1.1.3 ...] R3>

R1#按<Ctrl-Shift-6>x 切换回 R1

R1#show session

Conn	Host	Address	Byte	Idle	Conn Name
1	12.1.1.2	12.1.1.2	6	10	12.1.1.2
* 2	13.1.1.3	13.1.1.3	0	0	13.1.1.3

R1#disconnect 1 //从 R1 是主动断开到 R2 的连接(断开一个 TELNET 的会话)

Closing connection to 12.1.1.2 [confirm]

在 R2 上实验:

R2#show users //在 R2 上查看有谁登录到 R2 路由器

Line	User	Host(s)	Idle	Location
* 0 con 0		idle	00:00:00	
2 vty 0		idle	00:01:52	12.1.1.1

Interface	User	Mode	Idle Peer Address

R2#clear line 2 //发现有人 TELNET 过来,把连接清除

[confirm]

[OK] R2#show users

Line	User	Host(s)	Idle	Location
* 0 con 0		idle	00:00:00	

Interface	User	Mode	Idle Peer Address

提示:利用 TELNET 来管理的几条命令的区别.

show session //查看本路由器或是交换机主动打开了多少个 telnet 的会话

show user //查看本路由器或是交换机有谁 telnet 进来,可以显示他的 IP 地址

Disconnect //在本路由器上主动断开本路由器所打开的 telnet 会话

Clear line //清除远端连接,发现有 TELNET 的会话,把他清除

Ctrl+shift+6 x //telnet 会话的切换

Resume+会话 ID //重新再次连接,先 show session 查看会话 ID

一、 使用 SSH 远程控制远端设备, (包括路由器或是交换机或是防火墙等网络设备)。 注:

SSH 是基于 TCP 协议上的,端口号为 22

配置步骤(SSH Server):

1. Configure the IP domain name
2. Generate the RSA keys
3. Configure the SSH timeout interval <可选>
4. Configure the SSH retries <可选>
5. Disable vty inbound Telnet sessions
6. Enable vty inbound SSH sessions

例子:

R1#configure terminal

R1(config)#ip domain-name cisco.com

R1(config)#crypto key generate rsa general-keys modulus 512

R1(config)#ip ssh timeout 120

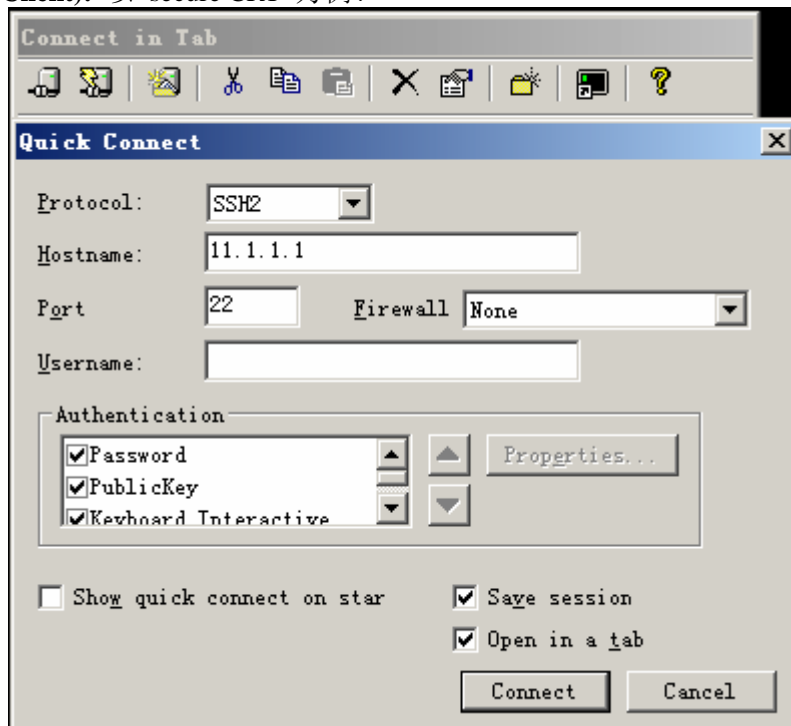
R1(config)#ip ssh authentication-retries 4

R1(config)#username cisco password cisco



```
R1(config)#line vty 0 4
R1(config-line)#transport input none
R1(config-line)#transport input ssh
R1(config-line)#login local
R1(config-line)#end
```

配置步骤(**SSH Client**): 以 secure CRT 为例:



在提示中输入用户名 cisco 密码 cisco 即可远程控制.

注: 用 SSH 代替 TELNET 的好处, SSH 的会话过程是加密的, 而 TELNET 的会话过程是明文

## 实验7. CISCO 路由器, 交换机密码恢复

### 一、CISCO 常用几条命令：

- show version 检查配置寄存器的值, 硬件配置, IOS 版本
- show flash 检查 Flash 中的 IOS, 或是 flash 大小, 使用情况(占用多少, 剩下多少)
- show startup-config 检查 NVRAM 中的启动配置文件(已保存后的配置)
- show running-config 检查 RAM 中的文件(当前所运行的配置)

### 二、路由器密码的恢复

2600 系列路由器步骤： 使用

console 控制路由器：

- 1、启动路由器，60 秒内按下 ctrl+break 键
- 2、rommon>confreg 0x2142
- 3、rommon>reset
- 4、router#copy startup-config running-config
- 5、router(config)#no enable secret //可以删除密码也可以更改, 这里为删除
- 6、router(config-line)#no enable password
- 7、router#copy running-config startup
- 8、router(config)#config-register 0x2102
- 9、router#reload //可选命令

2500 系列路由器步骤：

- 1、启动路由器，60 秒内按下 ctrl+break 键
- 2、>o/r 0x2142
- 3、>i

其余步骤跟 2610 一样

### 三、交换机密码的恢复. (以 CISCO 2950 交换机为例) 使用

console 控制交换机：

重启交换机：按 **MODE** 键进入到 **switch:** 模式 如下

C2950 Boot Loader (C2950-HBOOT-M) Version 12.1(11r)EA1, RELEASE SOFTWARE (fc1) Compiled

Mon 22-Jul-02 17:18 by antonino

WS-C2950-24 starting...

Base ethernet MAC Address: 00:13:1a:9a:2b:80

Xmodem file system is available.

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

**switch: flash\_init** //初始化 flash:

Initializing Flash...

flashfs[0]: 4 files, 1 directories

flashfs[0]: 0 orphaned files, 0 orphaned directories flashfs[0]:

Total bytes: 7741440

```
flashfs[0]: Bytes used: 3090944 flashfs[0]: Bytes available: 4650496
```

```
flashfs[0]: flashfs fsck took 6 seconds.
```

```
...done initializing flash.
```

```
Boot Sector Filesystem (bs:) installed, fsid: 3
```

```
Parameter Block Filesystem (pb:) installed, fsid: 4
```

```
switch: load_helper //加载帮助
```

```
switch: dir flash: //查看 flash:内容
```

```
Directory of flash:/
```

```
2  -rwx  736      <date>          vlan.dat
3  -rwx 3086336  <date>          c2950-i6q4l2-mz.121-22.EA2.bin
5  -rwx  1558    <date>          config.text //交换机启动时应用的配置
6  -rwx   5      <date>          private-config.text
```

```
4650496 bytes available (3090944 bytes used)
```

```
switch: rename flash:config.text flash:config-old.txt //重命名 config.text
```

```
switch: reset //重起交换机
```

```
Are you sure you want to reset the system (y/n)?y
```

```
System resetting...
```

重起交换机后由于交换机不会再应用配置文件，因为刚才已把配置文件的名字更改。交换机会进入到配置 的对话模式。

```
--- System Configuration Dialog ---
```

```
Would you like to enter the initial configuration dialog? [yes/no]: n
```

```
Switch#dir
```

```
Directory of flash:/
```

```
2  -rwx      736  Mar 01 1993 00:19:14 +00:00  vlan.dat
3  -rwx 3086336  Jan 01 1970 01:12:26 +00:00  c2950-i6q4l2-mz.121-22.EA2.bin
5  -rwx  1558    Mar 01 1993 02:36:44 +00:00  config-old.txt
6  -rwx   5      Mar 01 1993 02:36:44 +00:00  private-config.text
```

```
7741440 bytes total (4650496 bytes free)
```

```
Switch#copy config-old.txt running-config
```

```
Destination filename [running-config]?
```

```
1558 bytes copied in 1.152 secs (1352 bytes/sec)
```

```
sw#config t
```

```
Enter configuration commands, one per line. End with
```

```
CNTL/Z. sw(config)#line console 0
```

```
sw(config-line)#no pass cisco //清除 console 口密码
```

```
sw(config)#no enable secret //清除 enable 密码
```

```
sw(config)#no enable password //清除 enable 密
```

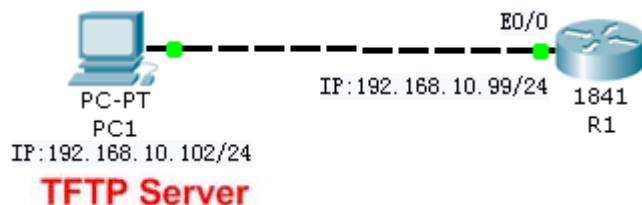
```
码 sw#write //保存
```

注：以上密码删除成功，一般情况下你可以不删除，就直接更改成你想要的密码就可以。

## 实验 8. IOS 软件备份、升级与 NVRAM 备份

Cisco 的网际操作系统 (IOS) 是思科设备的核心，随着网络技术的不断发展，通过升级以适应不断变化的技术，满足新的需求。

实验拓扑：



本实验目的：

1. 备份 R1 的 NVRAM (Startup-config)
2. 备份 R1 的 IOS 映像文件
3. 升级 R1 的 IOS 文件 描述：

首先保证 TFTP SERVER 和路由器是连通的。本实验中用 PC1 作为 TFTP 服务器 (PC1 装上 TFTP Server 的软件就可以作为一台 TFTP Server)

### 一、 备份 R1 的 NVRAM (Startup-config)

步骤：

配置 R1：

Router#**conf t**

Router(config)#**host R1**

R1(config)#**interface ethernet 0/0**

R1(config-if)#**ip address 192.168.10.99 255.255.255.0**

R1(config-if)#**no shutdown**

R1(config-if)#**exit**

R1(config)#**end**

配置 PC1 的 IP 地址：

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 5.2.3790]
(C) 版权所有 1985-2003 Microsoft Corp.

C:\Documents and Settings\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter wlan:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . . : 192.168.10.102
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.10.1

C:\Documents and Settings\Administrator>
```

保证 PC1 到 R1 的连通性：

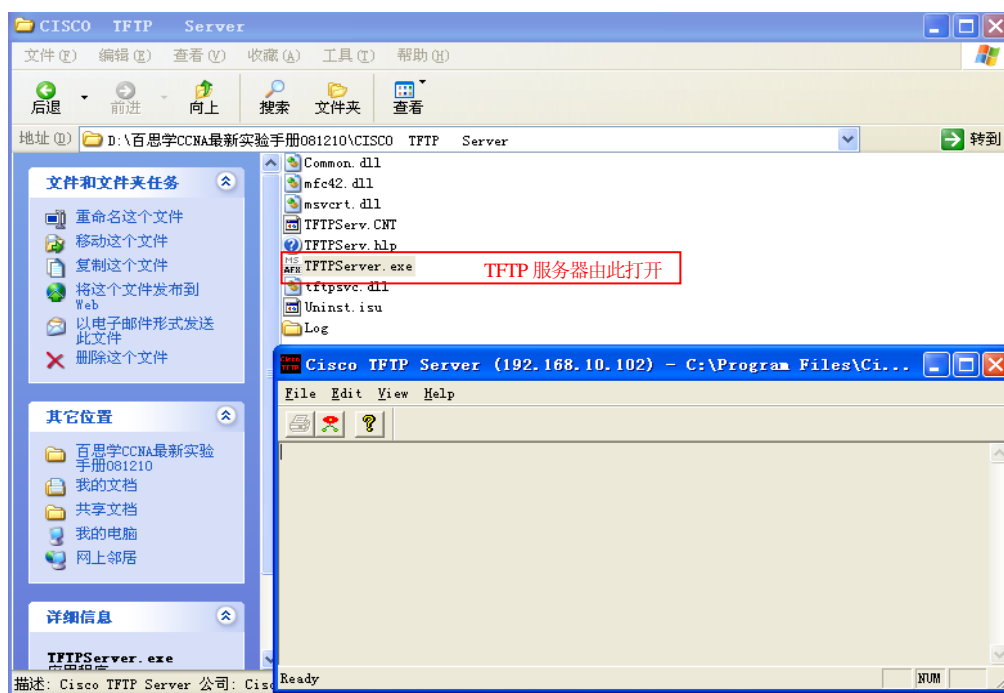
```
C:\Documents and Settings\Administrator>ping 192.168.10.99

Pinging 192.168.10.99 with 32 bytes of data:

Reply from 192.168.10.99: bytes=32 time=13ms TTL=255
Reply from 192.168.10.99: bytes=32 time=17ms TTL=255
Reply from 192.168.10.99: bytes=32 time=17ms TTL=255
Reply from 192.168.10.99: bytes=32 time=2ms TTL=255

Ping statistics for 192.168.10.99:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 17ms, Average = 12ms
```

安装 TFTP SERVER 软件（本实验使用 CISCO 的 TFTPServer）



在 R1 上使用 copy startup-config tftp:备份 NVRAM 到 TFTP SERVER

R1#copy startup-config tftp:

Address or name of remote host []? 192.168.10.102

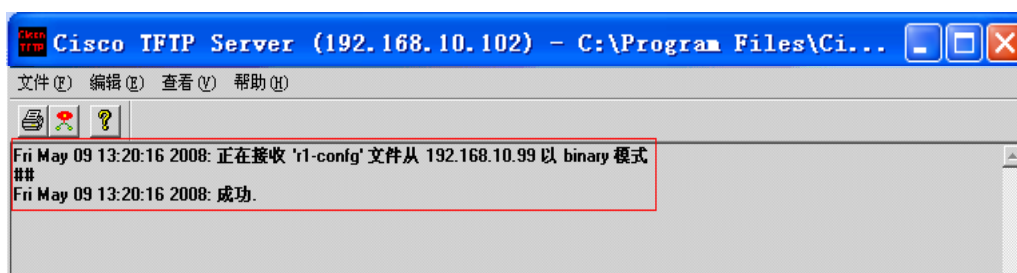
Destination filename [r1-config]?

!!

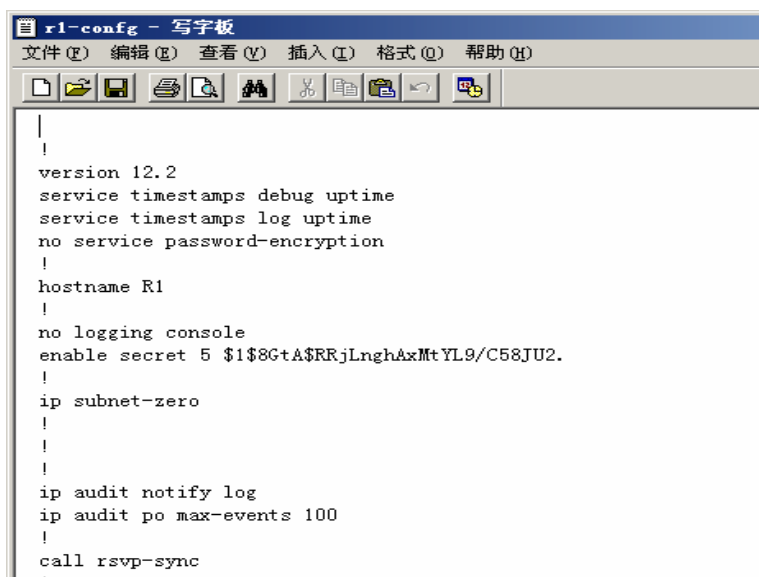
710 bytes copied in 0.072 secs (9861 bytes/sec)

R1#

在 TFTP SERVER 上查看已备份成功：



用写字板打开 r1-config 文件即可看到里面的具体配置



## 二、 备份 R1 的 IOS 映像文件

在 R1 上 show flash:查看映像文件的名字:

R1#show flash:

System flash directory:

File	Length	Name/status
1	12991168	c2600.bin

[12991232 bytes used, 3785984 available, 16777216 total]

16384K bytes of processor board System flash (Read/Write)

在 R1 上使用 copy flash: tftp:备份 IOS 的映像文件到 TFTP SERVER

R1#copy flash: tftp

Source filename []? c2600.bin

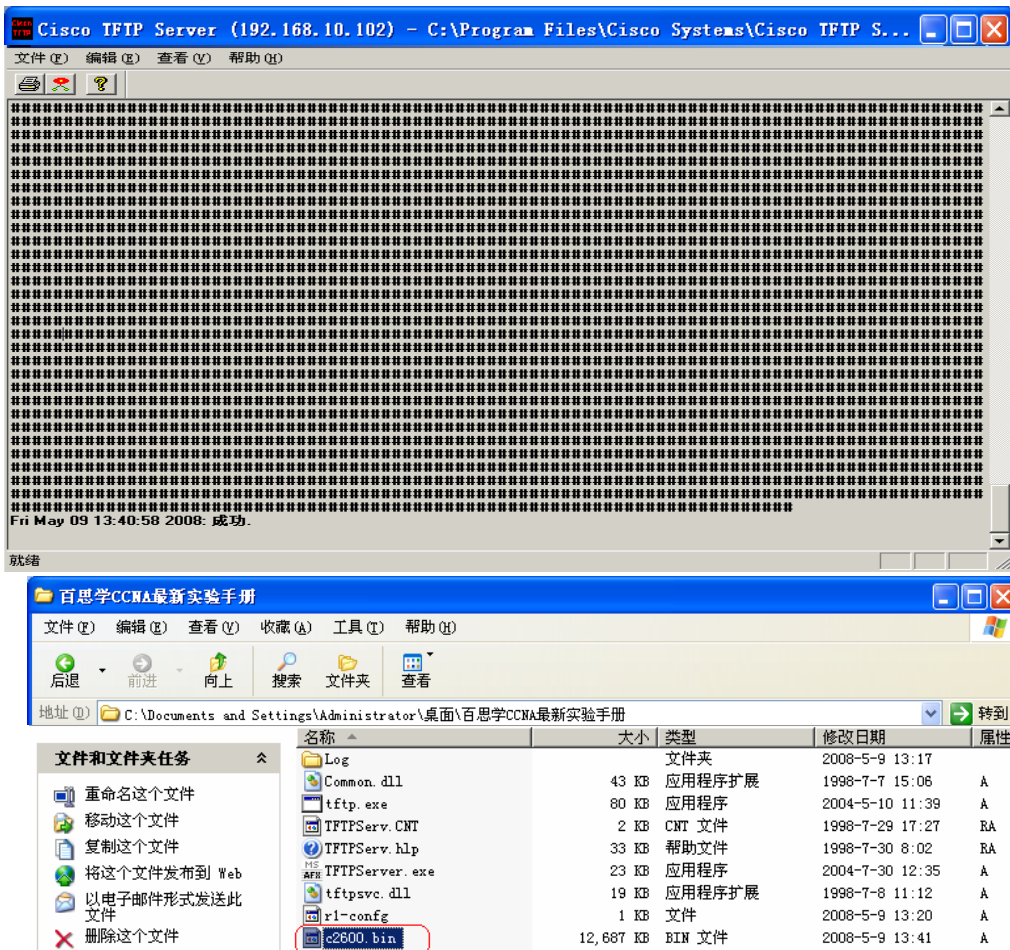
Address or name of remote host []? 192.168.10.102

Destination filename [c2600.bin]? c2600.bin

[illegible][illegible]

```
R1#copy flash: tftp  
Source filename []? c2600.bin  
Address or name of remote host []? 192.168.10.102  
Destination filename [c2600.bin]?  
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!  
1299168 bytes copied in 193.137 secs (67264 bytes/sec)  
R1#
```

在 R1 上查看备份的映像文件:



### 三、升级 R1 的 IOS 文件

(一) 升级 R1 的 IOS 文件 (在有 IOS 操作系统的情况下升级) 步骤:

查看正在运行的 IOS 映像文件:

R1#show flash:

System flash directory:

File	Length	Name/status
------	--------	-------------



1 12991168 c2600.bin -----注：正在运行的 IOS 映像文件  
[12991232 bytes used, 3785984 available, 16777216 total]

16384K bytes of processor board System flash (Read/Write) -----注：对 FLASH 有读或是写的权限

注：在 R1 上使用命令 copy tftp flash 升级 IOS 的映像文件，必须要保证 FLASH 有足够的空间存放 IOS 的映像文件，如果 FLASH 空间不够，用命令：delete flash:c2600.bin 删除已有的 IOS 文件。注意如果删除了 FLASH 里的映像文件后不要重启设备，也不能断电。

R1#copy tftp: flash:

Address or name of remote host []? 192.168.10.102

Source filename []? c2600\_new.bin

Destination filename [c2600\_new.bin]?

Accessing tftp://192.168.10.102/c2600\_new.bin...

（二）升级 R1 的 IOS 文件（在没有 IOS 操作系统的情况下升级）

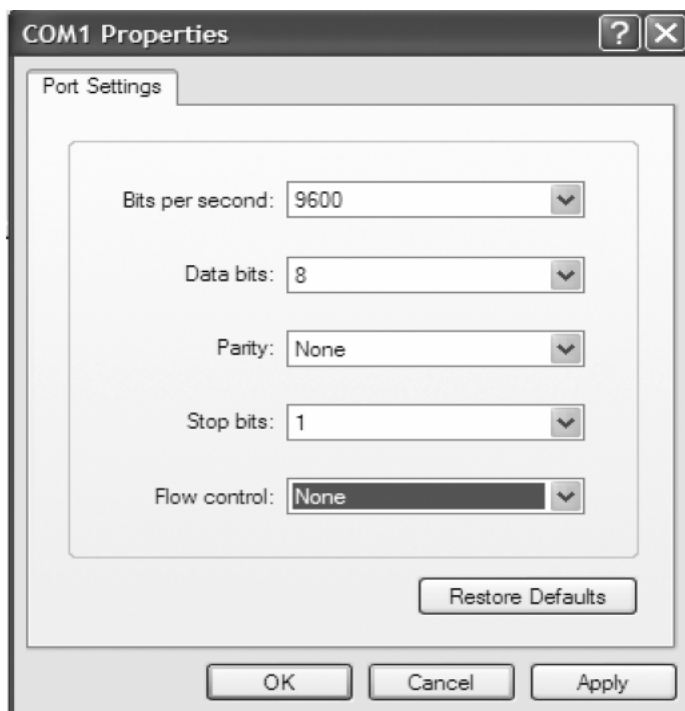
在路由器的 IOS 已经被删除，系统崩溃，也就是路由器当前运行在 rommon 的情况下：

升级步骤：（以 2600 路由器为例）

```
rommon 1 > IP_ADDRESS=192.168.10.99           // 设置本地 ip 地址
rommon 2 > IP_SUBNET_MASK=255.255.255.0       // 设置本地 ip 掩码
rommon 3 > DEFAULT_GATEWAY=192.168.10.102     // 设置默认网关
rommon 4 > TFTP_SERVER=192.168.10.102        // 设置 TFTP 服务器地址
rommon 5 > TFTP_FILE=c2600_new.bin            // 设置源文件名
rommon 6 > TFTPDNLD                           // 进行 TFTP 拷贝 注：这里要确保 TFTP SERVER 是打开的，且已存放有 c2600_new.bin 文件。
```

#### 四、通过 xmodem 升级 CISCO 交换机的 IOS 映像文件

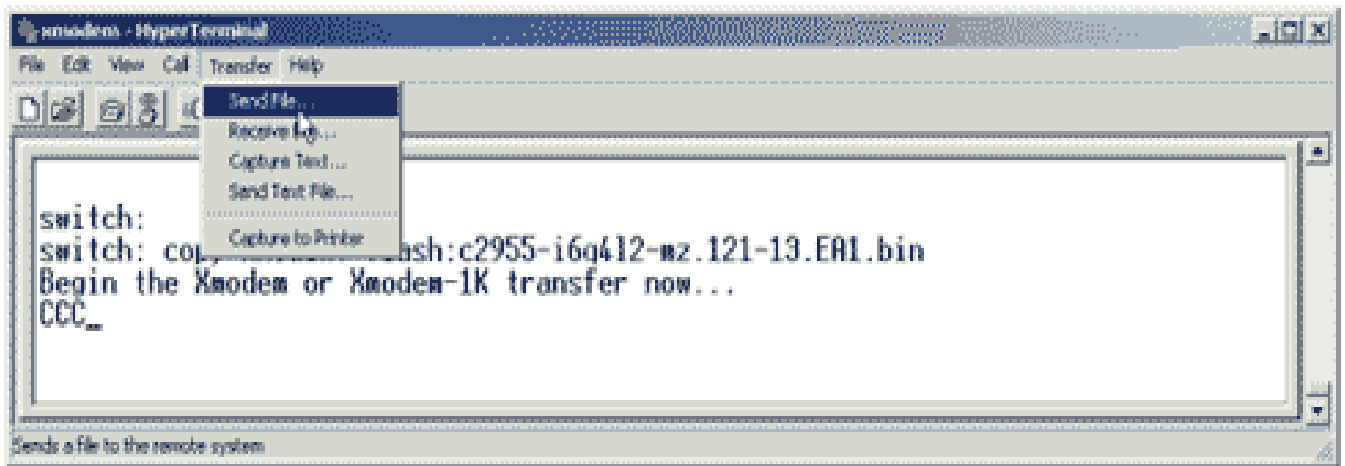
系统崩溃或是无法通过正常启动时，可以使用 xmodem 通信协议进行拷贝（通过 console 口控制交换机主机一端打开超级终端，配置方法如下图所示）



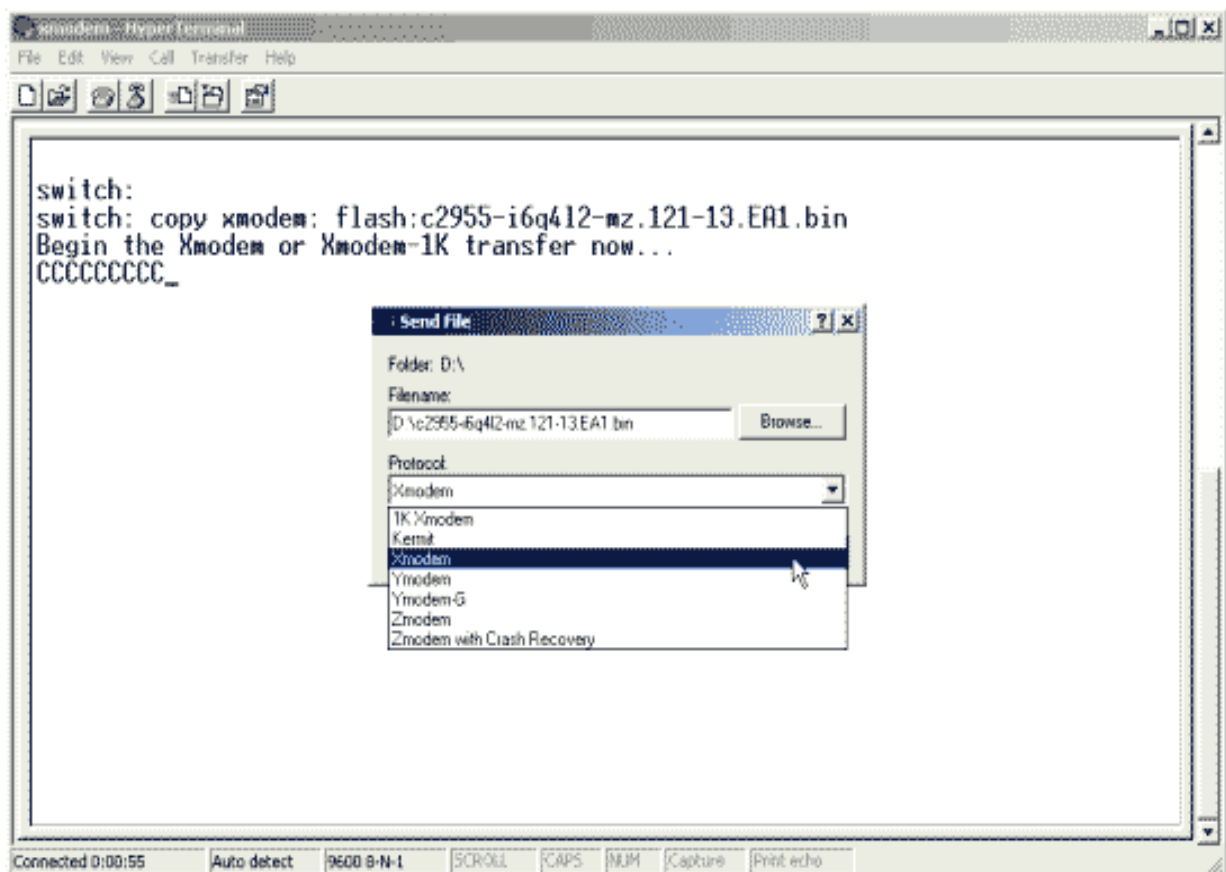
在交换机上配置：

```
switch: flash_init //进行 flash 的初始化
switch: load_helper //初始帮助文件的导入
switch: dir flash: //察看 flash 的状态
```

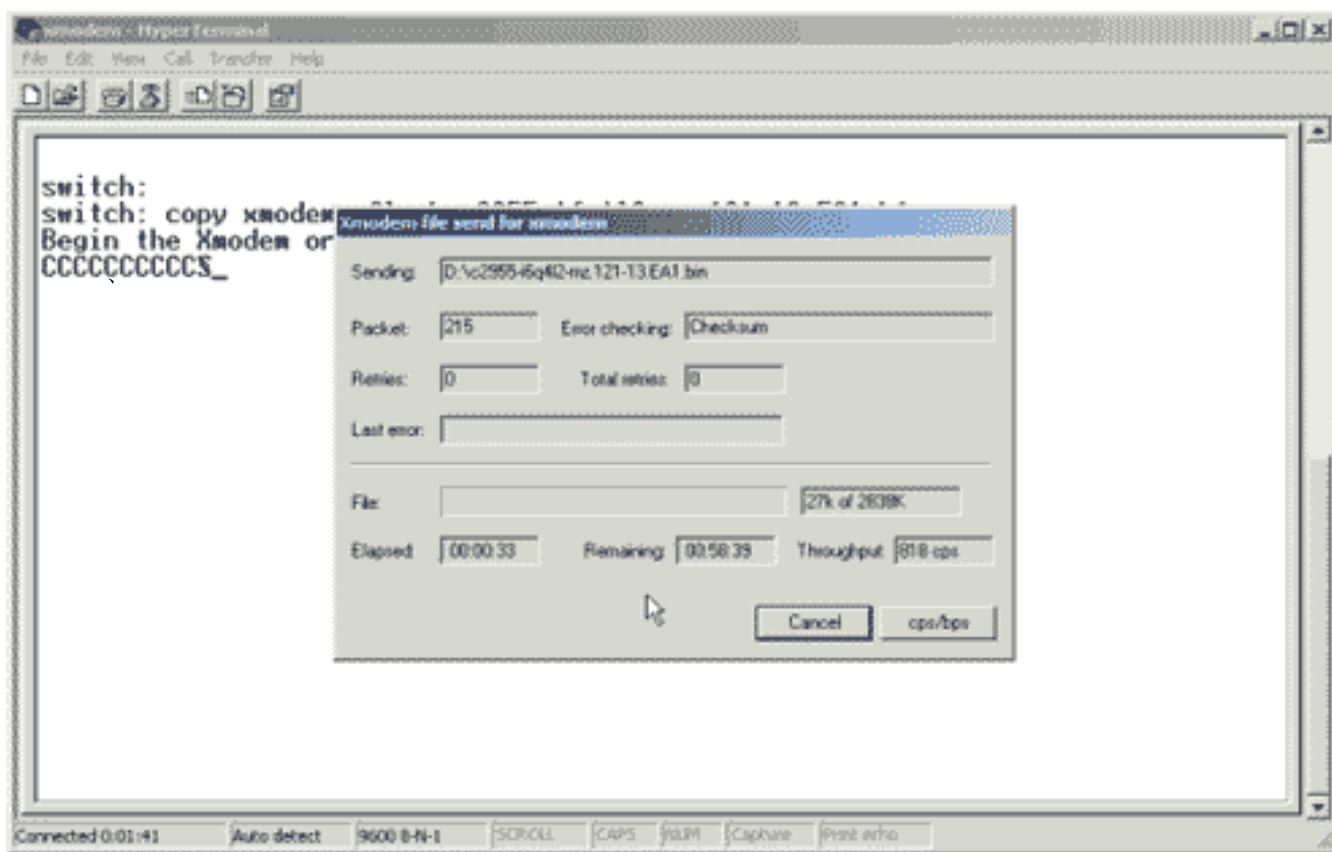
switch: copy xmodem: flash:c2950\_new.bin //以 xmodem 方式拷贝文件配置超级终端默认参数



选择 ‘传输’-‘发送文件’ 选项

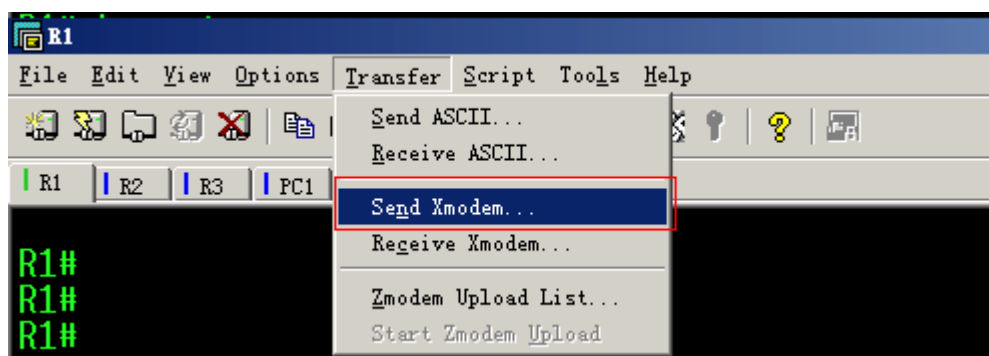


选择使用的协议 xmodem



确定后开始传送文件，传输完成后重新启动设备即完成拷贝

除了使用超级终端外，也可以用 secure CRT，选择 transfer→“send xmodem”：



找到 IOS 映像文件, 点 send(发送)即可

## 实验9. 交换机 VLAN、TRUNK、VTP 配置

### 1 · 配置 CISCO 二层交换机的 IP 地址(catalyst 2950 为例)

```
SW1(config)#int vlan 1           //进入管理接口 interface vlan 1
SW1(config-if)#ip address 11.1.1.2 255.255.255.0 //配置 IP 地址
SW1(config-if)#no shutdown
SW1(config-if)#exit
SW1(config)#ip default-gateway 11.1.1.1 //配置网关, 可通过 show run 查看
                                         所配置的 IP 地址:
```

```
SW1#show int vlan 1
```

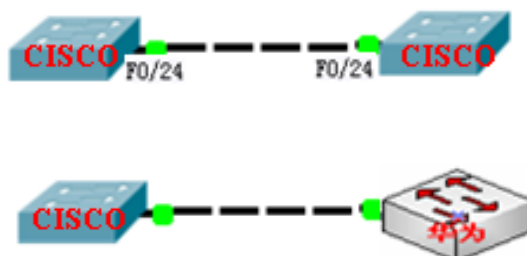
```
Vlan1 is up, line protocol is up
```

```
Hardware is CPU Interface, address is 0008.20ff.6400 (bia 0008.20ff.6400) Internet
address is 11.1.1.2/24
```

### 2 · 配置交换机的端口速度和双工 (Speed and Duplex)

```
SW1(config)#interface fa0/1
SW1(config-if)#speed {10 | 100 | auto} //10M/100M/自适应
SW1(config-if)#duplex {auto | full | half} //自适应/全双工/半双工
```

一般情况下, 交换机两端的端口速度和双工要匹配, 这样通信质量才能得到保证, 在相同厂家的产品 (比如说 Cisco 的交换机互连) 中端口协商不用配置一般不会有什问题, 可以通过 show interface 查看端口的 速度和双工。通常在不同厂家的产品中 (比如说 Cisco 和 华为互连) 如果通过查看发现端口速度和双工不 匹配, 可以通过手工配置来解决。如图:



通过 show interface 查看端口的速度和双工:

```
SW1#show interfaces fastEthernet 0/24
```

```
FastEthernet0/24 is up, line protocol is up (connected)
```

```
Hardware is Fast Ethernet, address is 0008.20ff.6418 (bia 0008.20ff.6418) MTU
1500 bytes, BW 100000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
```

**Full-duplex, 100Mb/s, media type is 100BaseTX**

```
input flow-control is unsupported output flow-control is unsupported
```

```
ARP type: ARPA, ARP Timeout 04:00:00
```

```
Last input 00:00:03, output 00:00:00, output hang never
```

注: CISCO 交换机端口默认值: Auto-duplex, Auto-speed, media type is 100BaseTX

Cisco 设备配置端口速度 (speed) 和双工 (duplex) 命令:

```
SW1(config)#interface fastEthernet 0/24
```

```
SW1(config-if)#speed ?
```

```
10 Force 10 Mbps operation -----注: 强制速度为 10M
```

100 Force 100 Mbps operation-----注：强制速度为 100M  
auto Enable AUTO speed configuration -----注：速度自动协议（默认值）

SW1(config-if)#**duplex** ?

auto Enable AUTO duplex configuration -----注：自动协商双工  
full Force full duplex operation -----注：强制为全双工  
half Force half-duplex operation -----注：强制为半双工

3. 设置永久 MAC 地址不过期

SW1(config)#mac-address-table static mac\_addr {vlan vlan\_id} [interface int1 [int2 ... int15]]

SW1(config)#mac-address-table static 1111.1111.a111 vlan 1 interface f0/1

配置后 show mac-address-table 可以看到静态添加的 MAC 地址 **1111.1111.a111**

SW1#show mac-address-table

Vlan	Mac Address	Type	Ports
All	000f.72db.4ec0	STATIC	CPU
<b>1</b>	<b>1111.1111.a111</b>	<b>STATIC</b>	<b>Fa0/1</b>

4 · 管理 MAC 地址表

SW1#show mac-address-table

Mac Address Table

Vlan	Mac Address	Type	Ports
All	000f.72db.4ec0	STATIC	CPU
1	0000.0c3f.0b05	DYNAMIC	Fa0/12
1	0030.94e6.391d	DYNAMIC	Fa0/11
1	00e0.b05a.5bfe	<b>DYNAMIC</b>	Fa0/10
<b>1</b>	<b>1111.1111.a111</b>	<b>STATIC</b>	<b>Fa0/1</b>

5.设置交换机端口的访问模式(三种) **access /dynamic / trunk**

SW1(config-if)#switchport mode ?

access Set trunking mode to ACCESS unconditionally  
dynamic Set trunking mode to dynamically negotiate access or trunk mode trunk  
Set trunking mode to TRUNK unconditionally

SW1(config)#int fastEthernet 0/1

SW1(config-if)#switchport mode access //设置 FA0/1 为 ACCESS 模式

SW1(config-if)#switchport mode trunk //设置 FA0/1 为 TRUNK 模式

SW1(config-if)#switchport mode dynamic auto //设置 FA0/1 为自动直协商模式(默认)

交换机 TRUNK 端口配置:

1) 一般交换机与交换机相连接的端口要设置为 trunk 模式

2) 如果是梳心交换机与非网管的交换机相连, 梳心交换机端口设置为 access 模式 如下图:



**配置 SW1:**

```
SW1(config)#int f0/24
SW1(config-if)#no shutdown
SW1(config-if)#switchport mode trunk //设置 Fa0/24 为 TRUNK 模式
```

**配置 SW2:**

```
SW2(config)#int f0/24
SW2(config-if)#no shutdown
SW2(config-if)#switchport mode trunk //设置 FA0/24 为 TRUNK 模式 验证 TRUNK 命令:
Show interface trunk
Show interface f0/24 switchport
```

**SW1#sho run int f0/24**

```
interface FastEthernet0/24 switchport
```

```
mode trunk SW1#show int trunk
```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/24	on	802.1q	trunking	1

```
SW1#show int f0/24 switchport
```

```
Name: Fa0/24
```

```
Switchport: Enabled Administrative
```

```
Mode: trunk Operational Mode: trunk
```

```
Administrative Trunking Encapsulation: dot1q
```

```
Operational Trunking Encapsulation: dot1q
```

```
Negotiation of Trunking: On
```

```
Access Mode VLAN: 1 (default)
```

```
Trunking Native Mode VLAN: 1 (default) Voice
```

```
VLAN: none
```

```
Administrative private-vlan host-association: none Administrative
```

```
private-vlan mapping: none Administrative private-vlan trunk native
```

```
VLAN: none Administrative private-vlan trunk encapsulation: dot1q
```

```
Administrative private-vlan trunk normal VLANs: none
```

```
Administrative private-vlan trunk private VLANs: none Operational
```

```
private-vlan: none
```

```
Trunking VLANs Enabled: ALL Pruning
```

```
VLANs Enabled: 2-1001
```

```
Capture Mode Disabled
```

```
Capture VLANs Allowed: ALL Protected:
```

```
falseAppliance trust: none
```

```
SW1#
```

**SW2:**

```
SW2#sho run int f0/24 interface
```

```
FastEthernet0/24
```

```
switchport mode trunk
```

```
end
```

```
SW2#show int trunk
```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/24	on	802.1q	trunking	1

## 6.配置 VLAN

### 1) .创建 VLAN

```
SW1(config)#vlan 2 //创建 VLAN2
SW1(config)#name VLAN2 //命名为 VLAN2, 默认为 vlan002, 命名是可选命令
```

### 2) .将端口加入 VLAN

```
SW1(config-if)#switchport mode access //设配端口为 access 模式
SW1(config-if)#switchport access vlan 2 //把端口加到 vlan2
```

### 3) . 检查的命令

Switch#show vlan 例子:

创建 VLAN2 命令为 wisdom , 创建  
VLAN3 命令为 market 把端口 F0/4  
加入 VLAN2

```
SW1#config t
SW1(config)#vlan 2
SW1(config-vlan)#name wisdom
SW1(config-vlan)#exit SW1(config)#vlan 3
SW1(config-vlan)#name market
SW1(config-vlan)#exit SW1(config)#int
fas0/4
```

```
SW1(config-if)#switchport mode access
```

```
SW1(config-if)#switchport access vlan 2
```

可以同时把多个端口加入到相应 **VLAN**

```
SW1(config)#interface range fastEthernet 0/10 – 15 //可以同时把多个端口加入到一个 VLAN 里
```

```
SW1(config-if-range)#switchport mode access
```

```
SW1(config-if-range)#switchport access vlan 3 //把端口 10-15 都加入到 VLAN3 里
```

```
SW1#show vlan //检查 VLAN 信息
```

VLAN Name	Status	Ports
1 default	active	Fa0/1, Fa0/5, Fa0/6, Fa0/7 Fa0/8, Fa0/9, Fa0/16, Fa0/17 Fa0/18, Fa0/19, Fa0/20, Fa0/21 Fa0/22, Fa0/23, Fa0/24
2 wisdom	active	<b>Fa0/4</b>
3 market	active	<b>Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15</b>

SW1#

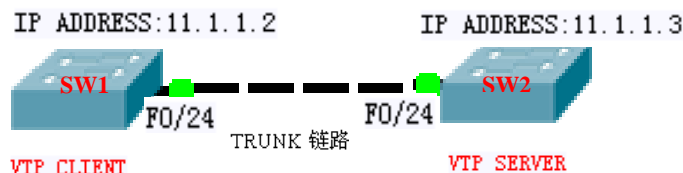
## 7.配置 VTP---VTP 技术使得在大型的网络里布置多个 VLAN 变得简单.

VTP 的配置步骤:

- 1、配置 Trunk (交换机和交换机相连的端口要设置为 TRUNK)
- 2、配置 VTP DOMAIN
- 3、配置 VTP MODE
- 4、配置 VLAN
- 5、将端口加入 VLAN
- 6、检查



```
Switch#show interface xx/xx switchport //查看相应接口的 trunk 状态
Switch#show vtp status //查看 vtp 状态
Switch#show vlan //查看 VLAN
```



配置 SW1:

```
SW1#configure terminal
SW1(config)#interface fastEthernet 0/24
SW1(config-if)#switchport mode trunk
SW1(config-if)#no shutdown
SW1(config-if)#exit
SW1(config)#vtp domain wisdom Domain name already set to wisdom.
SW1(config)#vtp mode server
Device mode already VTP SERVER.
SW1(config)#vtp password cisco
Setting device VLAN database password to cisco
SW1(config)#vtp pruning //配置 VTP 的修剪, 只要在 SERVER 端配置就行.
Pruning switched on
```

```
SW1#sho vtp status
VTP Version : 2
Configuration Revision : 1
Maximum VLANs supported locally : 64
Number of existing VLANs : 5
VTP Operating Mode : Server VTP Domain
Name : wisdom VTP
Pruning Mode : Enabled VTP V2
Mode : Disabled VTP Traps
Generation : Disabled
MD5 digest : 0xE8 0x5A 0x7D 0xB1 0x0E 0xBC 0xEB 0x1F
Configuration last modified by 11.1.1.2 at 3-1-93 02:56:31
Local updater ID is 11.1.1.2 on interface V11 (lowest numbered VLAN interface found)
```

创建 VLAN2 VLAN3 VLAN4 VLAN5

```
SW1(config)#vlan 2
SW1(config-vlan)#vlan 3
SW1(config-vlan)#vlan 4
SW1(config-vlan)#vlan 5
SW1(config-vlan)#
```

在 SW1 上创建 VLAN 后查看修订版本: SW1#show vtp status

```
VTP Version : 2
Configuration Revision : 5
Maximum VLANs supported locally : 128
Number of existing VLANs : 9
VTP Operating Mode : Server VTP Domain Name : wisdom
VTP Pruning Mode : Enabled VTP V2 Mode : Disabled VTP
Traps Generation : Disabled
MD5 digest : 0x9C 0x64 0xD6 0x44 0x5E 0x54 0x9E 0xFC Configuration last modified by
11.1.1.2 at 3-1-93 02:59:04
```

Local updater ID is 11.1.1.2 on interface V11 (lowest numbered VLAN interface found)

在交换机 **SW1** 上检查已创建的 **VLAN**

SW1#sho vlan

VLAN Name	Status	Ports
1 default	active	Fa0/1, Fa0/5, Fa0/6, Fa0/7 Fa0/8, Fa0/9, Fa0/16, Fa0/17 Fa0/18, Fa0/19, Fa0/20, Fa0/21 Fa0/22, Fa0/23, Fa0/24
2 VLAN0002	active	
3 VLAN0003	active	
4 VLAN0004	active	
5 VLAN0005	active	

检查 **trunk** 端口:

SW1#show interfaces fastEthernet 0/24 switchport

Name: Fa0/24

Switchport: Enabled Administrative

Mode: trunk **Operational Mode: trunk**

Administrative Trunking Encapsulation: dot1q

**Operational Trunking Encapsulation: dot1q**

Negotiation of Trunking: On

Access Mode VLAN: 1 (default)

Trunking Native Mode VLAN: 1 (default) Voice

VLAN: none

Administrative private-vlan host-association: none Administrative

private-vlan mapping: none Administrative private-vlan trunk native

VLAN: none Administrative private-vlan trunk encapsulation: dot1q

Administrative private-vlan trunk normal VLANs: none

Administrative private-vlan trunk private VLANs: none

Operational private-vlan: none Trunking VLANs Enabled: ALL Pruning VLANs Enabled:

2-1001

Capture Mode Disabled

Capture VLANs Allowed: ALL Protected: false

Appliance trust: none

配置 **SW2**: Switch# Switch#config t

Enter configuration commands, one per line. End with CNTL/Z. SW2(config)#hostname SW2

SW2(config)#int fastEthernet 0/24

SW2(config-if)#switchport mode trunk

SW2(config-if)#exit

SW2(config)#vtp domain wisdom

Changing VTP domain name from test to wisdom

SW2(config)#vtp mode client

Setting device to VTP CLIENT mode. SW2(config)#vtp password cisco

Setting device VLAN database password to cisco

SW2(config)#

查看 **F0/24** 的 **trunk** 状态:

SW2#show interfaces fastEthernet 0/24 switchport

Name: Fa0/24

Switchport: Enabled Administrative Mode: trunk **Operational Mode: trunk**

Administrative Trunking Encapsulation: dot1q

**Operational Trunking Encapsulation: dot1q**

Negotiation of Trunking: On

Access Mode VLAN: 1 (default)

Trunking Native Mode VLAN: 1 (default) Voice VLAN: none

Administrative private-vlan host-association: none Administrative private-vlan mapping: none Administrative private-vlan trunk native VLAN: none Administrative private-vlan trunk encapsulation: dot1q Administrative private-vlan trunk normal VLANs: none Administrative private-vlan trunk private VLANs: none Operational private-vlan: none

Trunking VLANs Enabled: ALL Pruning VLANs Enabled: 2-1001

Capture Mode Disabled

Capture VLANs Allowed: ALL Protected: false

Appliance trust: none

查看 **VTP** 状态:

SW2#show vtp status

VTP Version : 2

---

**Configuration Revision** : 5  
Maximum VLANs supported locally : 128  
Number of existing VLANs : 9  
VTP Operating Mode : Client VTP Domain  
Name : wisdom VTP  
Pruning Mode : Enabled VTP V2  
Mode : Disabled  
VTP Traps Generation : Disabled  
MD5 digest : 0x9C 0x64 0xD6 0x44 0x5E 0x54 0x9E 0xFC  
Configuration last modified by **11.1.1.2** at 3-1-93 02:59:04

-----此处可以看到 SW2 的 VLAN 信息是从 SW1-11.1.1.2 同步过来的

SW2#show vlan

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, Fa0/16, Fa0/17 Fa0/18, Fa0/19, Fa0/20, Fa0/21 Fa0/22, Fa0/23, Fa0/24
2 VLAN0002	active	
3 VLAN0003	active	
4 VLAN0004	active	
5 VLAN0005	active	

实验结果:

SW2 不用配置 VLAN,已同步了 SW1 的 VLAN 信息. 实验成功

SW2 同步 VTP SERVER 的 VLAN 后,接下来的操作就是安要求把相应的端口加入到相应的 VLAN

## 实验10. 配置生成树(Spanning Tree)

本实验只讨论 CST(公共的生成树)

一、实验前先理解生成树决策的 4 个步骤:

生成树决策4步:  
最低根桥BID→选出根桥  
到根桥最低开销→选出根端口  
最低发送者 BID→选出指定端口  
最低端口 ID→选出指定端口

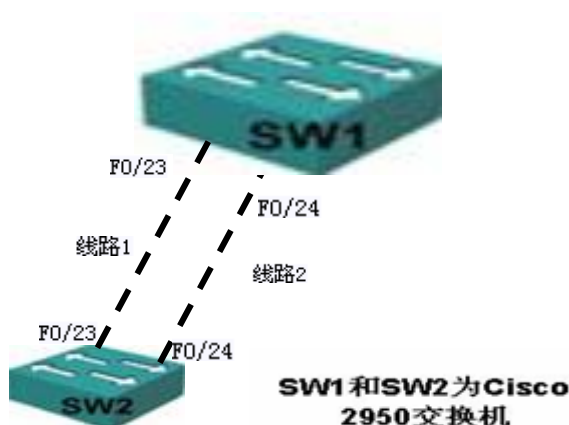
二、实验拓扑

1. 实验描述: 由于业务的要求, 要有可靠的链路, 要对链路实现冗余, 但链路的冗余有可能给网络带来广播风暴, 重复 帧, MAC 地址表不稳定的问题, Spanning Tree protocol 在可以实现冗余的同时又可以解决以上三个问题.

2. 实验 1: 验证 CST (公共生成树)

CST 是 IEEE 在虚拟局域网上处理生成树的特有方法, 这是一种 VLAN 解决方案, 称为单一或者公共生成树. 生成树协议运行在 VLAN1 即缺省的 VLAN 上. 所有的交换机都举出同一个根网桥, 并建立与该根网桥的关系

CISCO 的交换机默认就启用了生成树 Spanning Tree protocol, 不需配置就可以实现链路冗余能力. 实验拓扑:



实验拓扑描述:

在以上交换网络拓扑中只有交换机默认存在的 VLAN1, 交换机默认情况下所有端口属于 VLAN1。 根据以上拓扑, 也就是只有 VLAN1 存在的情况下我们来验证 CST (公共生成树) 建议手工配置 SW1, SW2 的 F0/23, F0/24 接口的 trunk 模式 配置 SW1:

SW1(config)#int range fastEthernet 0/23 - 24 -----注: 进入一个接口范围, 同时对多个端口进行配置

SW1(config-if-range)#switchport mode trunk

SW1(config-if-range)#no shutdown

配置 SW2:

SW2(config)#int range fastEthernet 0/23 - 24

SW2(config-if-range)#switchport mode trunk

SW2(config-if-range)#no shutdown

在 SW1 上用 **show spanning-tree** 验证生成树：

SW1#show spanning-tree

VLAN0001

Spanning tree enabled protocol ieee

Root ID Priority 32769

Address 0008.20ff.6400

**This bridge is the root**

-----注： SW1 为根网桥

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

-----注：根交换机的网桥 ID

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1) Address  
0008.20ff.6400

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 15

-----注： SW1 的网桥 ID

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

Fa0/23	<b>Desg</b>	FWD	19	128.23	P2p

注： **Desg**：表示 F0/23 为指定端口 **FWD** 表示：端口的生树成状态 为转发状态表示 F0/23 到根网桥 的花费为 19 128.23：这里 128 表 示端口优先级，23 表示端口号， 这里可以先不用考虑这个。

Fa0/24	<b>Desg</b>	FWD	19	128.24	P2p
--------	-------------	-----	----	--------	-----

注： CST 是 IEEE 在虚拟局域网上处理生成树的特有方法，这是一种 VLAN 解决方案，称为单一或者公共生成树。生成树协议运行在 VLAN1 即缺省的 VLAN 上。

<**Desg** 表示指定端口>

在 SW2 上用 **show spanning-tree** 验证生成树：

SW2#show spanning-tree

VLAN0001

Spanning tree enabled protocol ieee

Root ID Priority 32769

-----注：根交换机的网桥 ID

Address 0008.20ff.6400

Cost 19

Port 23 (FastEthernet0/23)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)

Address 000d.bce7.5940

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300 ----- SW2 的网桥 ID

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

Fa0/23	<b>Root</b>	FWD	19	128.23	P2p
Fa0/24	<b>Altn</b>	BLK	19	128.24	P2p

注：解释：F0/24 为 altn(后备端口)，端口的生成树状态为 BLK(blocking)状态，Root 表示根端口，Altn 表示后备端口。

回顾生成树决策的 4 个步骤：

**生成树决策4步：**  
**最低根桥BID→选出根桥**  
**到根桥最低开销→选出根端口**  
**最低发送者 BID→选出指定端口**  
**最低端口 ID→选出指定端口**

通过在 SW1,SW2 上 show spanning-tree 得到以下结果.

SW1 (根交换机)

F0/23-----指定端口

F0/24-----指定端口 SW2 (非根交换机)

F0/23-----根端口

F0/24-----blocking 端口 生成

树决策过程：

- 选根交换机：

**One root bridge per network** (每个网络只有一个根桥)

根桥的选举：**Lowest BID** (最小的BID)

SW1 的竞选根交换机参数：

Bridge ID: Priority **32768**

MAC Address 0008.20ff.6400

SW2 的竞选根交换机参数: Bridge

ID: Priority **32768**

MAC Address **000d.bce7.5940**

选择根交换机第一个条件先看优先级 Priority 再看 MAC 地址, 缺省 SW1 和 SW2 的优先级都是 32768, 通过优先级无法选择根交换机, 只能通过 MAC 地址选择, 交换机的 MAC 地址可以通过 show version 查看.

SW1 的 MAC 地址 **0008.20ff.6400** 比 SW2 的 MAC 地址 **000d.bce7.5940** 小, 越小越优先, 所以 **SW1** 为根交换机

- 选根端口：

**One root port per nonroot bridge** (每个非根桥都有一个根端口)

根端口 (RP) : **Lowest path cost to root bridge** 每个非根桥有且只有一个根端口, 非根桥到达根桥所需开销最小的那个端口即为根端口。(可转发流量)

选举RP/DP的方法: (RP—根端口, DP—指定端口)

- 1.Lowest RID(最小的RID) 这里是(根桥)的BID
- 2.Lowest path cost to root bridge (到达根的最小路径开销)
- 3.lowest sender BID (最小的发送BID)
- 4.Lowest sender port ID 当两台交换机之间有两条线路直连时会用到这一项来选 (比如本实验中指定端口的选择就会用到这一项)

SW1 为根桥, 不会有根端口, 因为只有非根桥有根端口: SW2 为非根桥, 根端口肯定是 F0/23, F0/24 的其中一个:

F0/23, F0/24 这个两个端口选举根端口的条件: (非根桥到达根桥所需开销最小的那个端口即为根端口)

F0/23 到达根网桥的开销(cost)为 19

F0/24 到达根网桥的开销(cost)为 19 通过非根桥到达根桥所需开销最小这个条件没法选出根端口, 只能再看生成树决策的第3个条件即lowest sender BID (最小的发送BID)

通过 lowest sender BID (最小的发送 BID)选举, 但 F0/23, F0/24 都是在 SW2 (非根桥)上的两个端口

lowest sender BID 都是:

Bridge ID: Priority **32768**

MAC Address **000d.bce7.5940**



所以这里无法通过lowest sender BID选出根端口, 只能再看生成树决策的第4个条件Lowest sender port ID, 当两台交换机之间有两条线路直连时会用到这一项(端口号越小越优先) F0/23比F0/24小, 所以F0/23优先. 最终可以选出根端口F0/23

SW2上剩下F0/24的就是blocking端口

在SW2上show spanning-tree验证:

SW2#show spanning-tree

VLAN0001

Spanning tree enabled protocol ieee

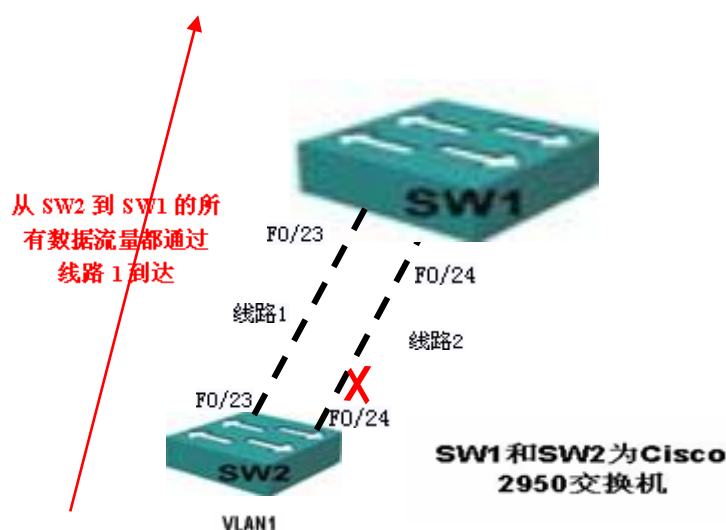
```
Root ID    Priority    32769
           Address    0008.20ff.6400
           Cost      19
           Port      23 (FastEthernet0/23)
           Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
```

```
Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1) Address
           000d.bce7.5940
           Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
           Aging Time 300
```

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/23	Root	FWD	19	128.23	P2p
Fa0/24	Altn	BLK	19	128.24	P2p

-----注: F0/23 为根端口  
-----注: SW2 上 F0/24 就是 blocking 端口

总结:



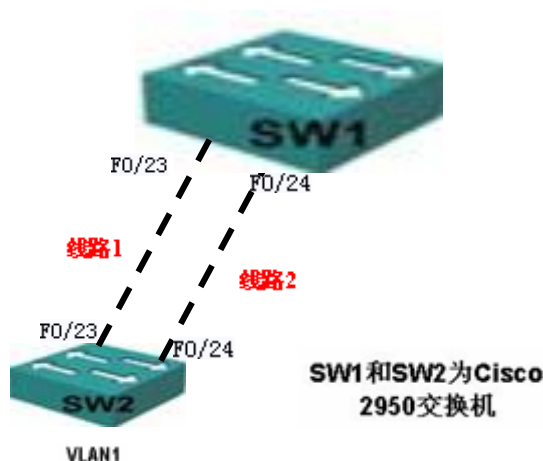
看上图, 从SW2到SW1的所有数据流量最终通过线路1到达, SW2的F0/24处于备份状态, 在SW2上show spanning-tree可以看到F0/24的端口角色为Altn, 即线路2作为线路1上备份链路.

**CST** 的缺点: 最终有一条链路总是处在备份的状态, 就像本实验中线路 2 处在备份的状态, 我们想象假如线路 1 永远不会出问题, 如果这样, 那好像线路 2 的存在是多余的, 于是我们有个想法就是能不能 两条链路都利用起来, 比如说一部分流量走线路 1, 另一部分流量走线路 2, 其实是可以的, CISC0 的 PVST+产生就由来于此, 我们将在下个实验中介绍 PVST+.



### 实验11. PVST+(每个 VLAN 的生成树 PVST 加)

实验拓扑:



分别在 SW1 和 SW2 上 show spanning-tree 查看结果:

SW1#show spanning-tree

VLAN0001

Spanning tree enabled protocol ieee

Root ID	Priority	32769
	Address	0008.20ff.6400

## This bridge is the root

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID	Priority	32769 (priority 32768 sys-id-ext 1)	Address
		0008.20ff.6400	

Hello Time 2 sec    Max Age 20 sec    Forward Delay 15 sec  
Aging Time 15

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

Fa0/23	Desg FWD 19	128.23	P2p
Fa0/24	Desg FWD 19	128.24	P2p

SW2#show spanning-tree

VLAN0001

Spanning tree enabled protocol ieee

Root ID	Priority	32769
	Address	0008.20ff.6400

Cost	19
Port	23 (FastEthernet0/23)

Hello Time 2 sec    Max Age 20 sec    Forward Delay 15 sec

Bridge ID	Priority	32769 (priority 32768 sys-id-ext 1)	Address
		000d.bce7.5940	

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec  
Aging Time 300

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

Fa0/23	Root	FWD	19	128.23	P2p
--------	------	-----	----	--------	-----

Fa0/24      Altn      BLK      19      128.24      P2p

注：以上拓扑中经过选举最终 SW1 为根网桥，SW2 的 F0/24 为 blocking 端口，也就是就线路 2 成为了备份链路。

分别在 SW1 和 SW2 上创建 VLAN2，VLAN3，VLAN4，按如下拓扑要求完成本实验：

在 SW1 上创建 VLAN2，VLAN3，VLAN4 并查看：

SW1(config)#vlan 2-4

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, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22
2    VLAN0002	active	
3    VLAN0003	active	
4    VLAN0004	active	

在 SW2 上创建 VLAN2，VLAN3，VLAN4 并查看：

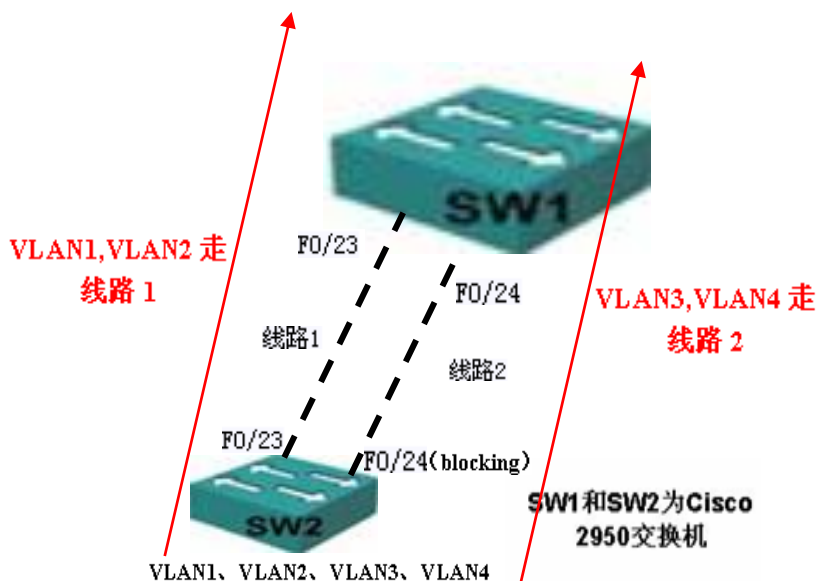
SW2(config)#vlan 2-4

SW2#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, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22
2    VLAN0002	active	
3    VLAN0003	active	
4    VLAN0004	active	

按如下拓扑要求完成本实验：

要求：SW2 的 VLAN1 和 VLAN2 流量通过线路 1 到达 SW1，SW2 的 VLAN3 和 VLAN4 的流量经过线路 2 到达 SW1，当其中一条链路有问题走同一链路。



分别在 **SW1** 和 **SW2** 上创建了 **VLAN2-4** 后查看目前的生成树状态。

在 **SW1** 上查看：

命令：show spanning-tree vlan 1

show spanning-tree vlan 2 show spanning-tree vlan 3 show spanning-tree vlan 4

SW1#show spanning-tree vlan 1

VLAN0001

Spanning tree enabled protocol ieee

```
Root ID    Priority 32769
           Address 0008.20ff.6400
           This bridge is the root
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 32769 (priority 32768 sys-id-ext 1)
           Address 0008.20ff.6400
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
           Aging Time 300
```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/23	Desg FWD	19	128.23	P2p		
Fa0/24	Desg FWD	19	128.24	P2p		

SW1#show spanning-tree vlan 2

VLAN0002

Spanning tree enabled protocol ieee

```
Root ID    Priority 32770
           Address 0008.20ff.6400
           This bridge is the root
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 32770 (priority 32768 sys-id-ext 2)
           Address 0008.20ff.6400
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
           Aging Time 300
```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/23	Desg FWD	19	128.23	P2p		
Fa0/24	Desg FWD	19	128.24	P2p		

SW1#show spanning-tree vlan 3

VLAN0003

Spanning tree enabled protocol ieee

```
Root ID    Priority 32771
           Address 0008.20ff.6400
           This bridge is the root
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 32771 (priority 32768 sys-id-ext 3) Address
           0008.20ff.6400
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
           Aging Time 300
```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
Fa0/23	Desg FWD	19	128.23	P2p		
Fa0/24	Desg FWD	19	128.24	P2p		

SW1#show spanning-tree vlan 4

VLAN0004

Spanning tree enabled protocol ieee

```
Root ID    Priority 32772
           Address 0008.20ff.6400
           This bridge is the root
```

```

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 32772 (priority 32768 sys-id-ext 4)
Address 0008.20ff.6400
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 300

```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/23	Desg	FWD	19	128.23	P2p	
--------	------	-----	----	--------	-----	--

Fa0/24	Desg	FWD	19	128.24	P2p	
--------	------	-----	----	--------	-----	--

结果: 查看生成树状态后发现针对VLAN1, VLAN2, VLAN3, VLAN3, SW1 都是扮演根网桥角色. SW1连接 SW2 的两个端口 F0/23, F0/24 都是指定端口.

在 SW2 上查看:

```

命令: show spanning-tree vlan 1
      show spanning-tree vlan 2 show
      spanning-tree vlan 3 show
      spanning-tree vlan 4

```

SW2#show spanning-tree vlan 1

VLAN0001

Spanning tree enabled protocol ieee

```

Root ID    Priority    32769
           Address    0008.20ff.6400
           Cost      19
           Port      23 (FastEthernet0/23)
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID   Priority    32769 (priority 32768 sys-id-ext 1) Address
           000d.bce7.5940
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
           Aging Time 300

```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/23	Root	FWD	19	128.23	P2p	
--------	------	-----	----	--------	-----	--

Fa0/24	Altn	BLK	19	128.24	P2p	
--------	------	-----	----	--------	-----	--

SW2#show spanning-tree vlan 2

VLAN0002

Spanning tree enabled protocol ieee

```

Root ID    Priority    32770
           Address    0008.20ff.6400
           Cost      19
           Port      23 (FastEthernet0/23)
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID   Priority    32770 (priority 32768 sys-id-ext 2) Address
           000d.bce7.5940
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
           Aging Time 300

```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/23	Root	FWD	19	128.23	P2p	
--------	------	-----	----	--------	-----	--

Fa0/24	Altn	BLK	19	128.24	P2p	
--------	------	-----	----	--------	-----	--

SW2#show spanning-tree vlan 3

VLAN0003

Spanning tree enabled protocol ieee

```

Root ID    Priority    32771

```

```

Address      0008.20ff.6400
Cost         19
Port         23 (FastEthernet0/23)
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Bridge ID    Priority 32771 (priority 32768 sys-id-ext 3)
Address 000d.bce7.5940
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Aging Time   300

Interface      Role Sts Cost      Prio.Nbr Type
-----
Fa0/23        Root FWD 19        128.23   P2p
Fa0/24        Altn BLK 19        128.24   P2p
SW2#show spanning-tree vlan 4
VLAN0004
Spanning tree enabled protocol ieee
Root ID      Priority    32772
Address      0008.20ff.6400
Cost         19
Port         23 (FastEthernet0/23)
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Bridge ID    Priority    32772 (priority 32768 sys-id-ext 4)
Address      000d.bce7.5940
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Aging Time   300

Interface      Role Sts Cost      Prio.Nbr Type
-----
Fa0/23        Root FWD 19        128.23   P2p
Fa0/24        Altn BLK 19        128.24   P2p

```

结果:查看生成树状态后发现针对 VLAN1, VLAN2, VLAN3, VLAN3, SW2 都是扮演非根网桥角色, SW2 的端口 F0/24 都处在 blocking 状态。

问题所在:通过在 SW1 和 SW2 查看得到的结果我们分析一下,现在 SW1 的 VLAN1 和 VLAN2 是通过线路 1 到达 SW1 的,这符合本实验的要求,但 VLAN3, VLAN4 到 SW1 走线路 1 不符合本实验的要求, VLAN3, VLAN4 到 SW1 要求走线路 2。

分析:SW2 的 VLAN3, VLAN4 到 SW1 为什么走线路 1 呢?因为 SW2 相对于 VLAN3, VLAN4 来说到达 SW1 F0/24 都是 blocking 状态, F0/23 为转发状态, 导致 VLAN3, VLAN3 通过 F0/23 走线路 1 到达 SW1。

解决方法:在 SW2 上 VLAN2, VLAN3 要到达 SW1 只要 SW2 的 F0/23 为 blocking 状态, F0/24 为转发状态, 就达到本实验的要求(SW2 的 VLAN3, VLAN4 到 SW1 走线路 2)

之所以 SW2 的 F0/24 为 blocking 状态, 是因为相对于 VLAN3, VLAN4 来说在通过生成树决策的第 4 步时, F0/23 端口号小于 F0/23, 小的优先, Lowest sender port ID 当两台交换机之间有两条线路直连时会用到这一项, <本实验就是这样>。通过以上分析, 在 SW2 上相对于 VLAN3, VLAN4 来说, 我们可以更改参数, 也就是在生成树决策的第 2 步就可以做出决策, 即使用 Lowest path cost to root bridge (到达根的最小路径开销)。

在 SW2 上相对于 VLAN3, VLAN4 更 F0/24 到根(SW1)的花费, 目前为 19, F0/23 到根也为 19, 我们更改 F0/24 端口到达 SW1 的花费为 18 就可以满足本实验的要求。

在 SW2 上配置以下命令:

```

SW2(config)#interface fastEthernet 0/24
SW2(config-if)#spanning-tree vlan 3-4 cost 18
查看更改的配置: SW2#show
run int f0/24
interface FastEthernet0/24

```

```
spanning-tree vlan 3-4 cost 18
```

```
end
```

做上以上配置后在 SW2 上查看当前的 VLAN3, VLAN4 的生成树状态:

```
SW2#show spanning-tree vlan 3
```

```
VLAN0003
```

```
Spanning tree enabled protocol ieee    Root ID          Priority 32771
                                          Address          0008.20ff.6400
                                          Cost             18
                                          Port            24 (FastEthernet0/24)
                                          Hello Time      2 sec  Max Age 20 sec  Forward Delay 15 sec
Bridge ID Priority    32771 (priority 32768 sys-id-ext 3) Address          000d.bce7.5940
                                          Hello Time      2 sec  Max Age 20 sec  Forward Delay 15 sec
                                          Aging Time      300
```

```
Interface      Role Sts Cost      Prio.Nbr Type
```

```
-----
Fa0/23         Altn BLK 19      128.23   P2p
Fa0/24         Root FWD 18      128.24   P2p
```

```
SW2#show spanning-tree vlan 4
```

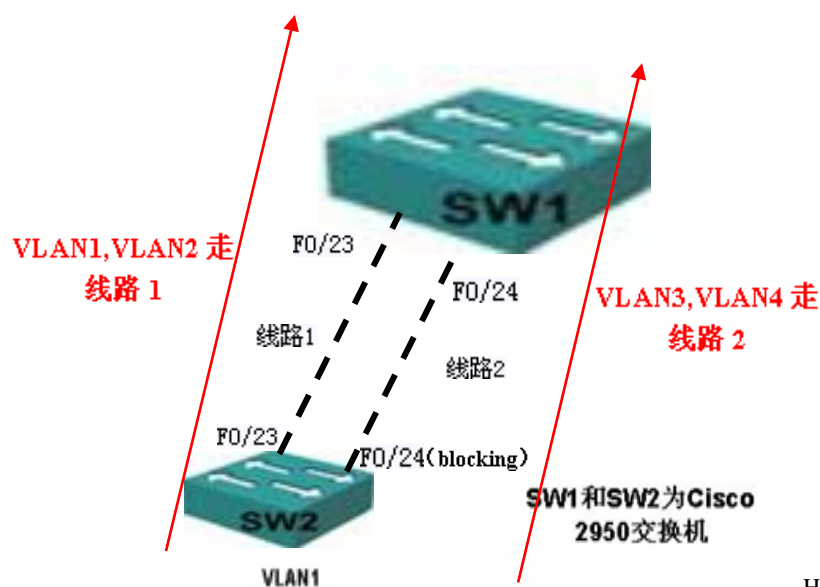
```
VLAN0004
```

```
Spanning tree enabled protocol ieee
Root ID          Priority    32772
Address          0008.20ff.6400
Cost             18
Port            24 (FastEthernet0/24)
Hello Time      2 sec  Max Age 20 sec  Forward Delay 15 sec
Bridge ID Priority    32772 (priority 32768 sys-id-ext 4) Address          000d.bce7.5940
                                          Hello Time      2 sec  Max Age 20 sec  Forward Delay 15 sec
                                          Aging Time      300
```

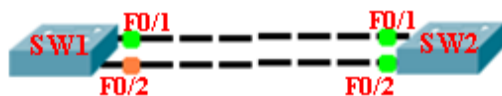
```
Interface      Role Sts Cost      Prio.Nbr Type
```

```
-----
Fa0/23         Altn BLK 19      128.23   P2p
Fa0/24         Root FWD 18      128.24   P2p
```

通过命令 `show spanning-tree vlan 3` , `show spanning-tree vlan 4` 查看,我们发现 F0/23 为 blocking 状态,SW2 的 VLAN3,VLAN4 到达 SW1 的通过 F0/24 到达,也就是走线路 2,实现在本实验的要求。最终线路1, 线路 2 流量走势图:



## 实验12. Configuring Link Aggregation with EtherChannel



把两个100Mbps的端口捆绑,成为200Mbps端口channel

配置 SW1 :

```
interface FastEthernet0/1
  switchport trunk encapsulation dot1q switchport
  mode trunk
  channel-group 1 mode on // Enable Etherchannel only interface
FastEthernet0/2
  switchport trunk encapsulation dot1q
  switchport mode trunk
  channel-group 1 mode on // Enable Etherchannel only
```

配置 SW2 :

```
interface FastEthernet0/1
  switchport trunk encapsulation dot1q switchport
  mode trunk
  channel-group 1 mode on // Enable Etherchannel only interface
FastEthernet0/2
  switchport trunk encapsulation dot1q
  switchport mode trunk
  channel-group 1 mode on // Enable Etherchannel only
```

其它参数:

```
sw1(config-if)#channel-group 1 mode ?
  active      Enable LACP unconditionally
  auto        Enable PAgP only if a PAgP device is detected
  desirable   Enable PAgP unconditionally
  on          Enable Etherchannel only
  passive     Enable LACP only if a LACP device is detected
sw2(config-if)#channel-group 1 mode ?
  active      Enable LACP unconditionally
  auto        Enable PAgP only if a PAgP device is detected
  desirable   Enable PAgP unconditionally
  on          Enable Etherchannel only
  passive     Enable LACP only if a LACP device is detected
```

相关检查命令:

```
sw1#show etherchannel 1 detail
Group state = L2
Ports: 2    Maxports = 8
Port-channels: 1 Max Port-channels = 1
Protocol:   -
Port: Fa0/1
```

Ports in the group:

```
-----
Port state      = Up Mstr In-Bndl
```

```
Channel group = 1          Mode = On/FEC          Gcchange = -
Port-channel  = Po1        GC   =   -          Pseudo port-channel = Po1
Port index    = 0          Load = 0x00        Protocol =   - Age of the
port in the current state: 00d:00h:19m:18s
```

#### Port: Fa0/2

#### Port state = Up Mstr In-Bndl

```
Channel group = 1          Mode = On/FEC          Gcchange = -
Port-channel  = Po1        GC   =   -          Pseudo port-channel = Po1
Port index    = 0          Load = 0x00        Protocol =   - Age of the
port in the current state: 00d:00h:19m:18s
```

Port-channels in the group:

#### Port-channel: Po1

```
Age of the Port-channel  = 00d:00h:35m:33s
Logical slot/port      = 1/0          Number of ports = 2
GC                     = 0x00000000    HotStandBy port = null
Port state             = Port-channel Ag-Inuse
Protocol               =   - Ports in the
```

Port-channel:

Index	Load	Port	EC state	No of bits
0	00	Fa0/1	On/FEC	0
0	00	Fa0/2	On/FEC	0

Time since last port bundled: 00d:00h:19m:19s Fa0/2

Time since last port Un-bundled: 00d:00h:23m:27s Fa0/2

sw1#show etherchannel ?

```
<1-64>      Channel group number
detail      Detail information
load-balance Load-balance/frame-distribution scheme among ports in  port-channel

port        Port information
port-channel Port-channel information protocol
            protocol enabled

summary     One-line summary per channel-group
|           Output modifiers
<cr>
```

查看 port-channel 1 的端口状态: Sw1#show

interfaces port-channel 1

Port-channel1 is up, line protocol is up (connected)

```
Hardware is EtherChannel, address is 0011.92e4.2782 (bia 0011.92e4.2782) MTU 1500
bytes, BW 200000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
```

Sw2#show interfaces port-channel 1

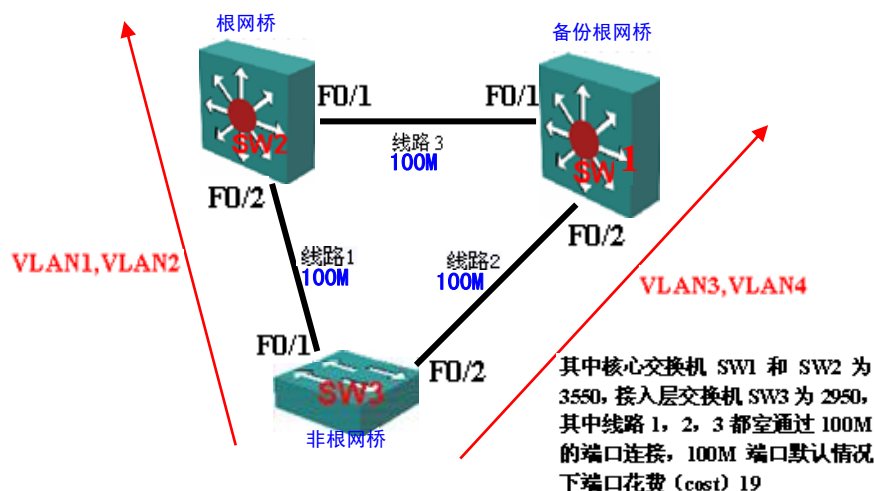
Port-channel1 is up, line protocol is up (connected)

```
Hardware is EtherChannel, address is 0011.92e4.2782 (bia 0011.92e4.2782) MTU 1500
bytes, BW 200000 Kbit, DLY 100 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, loopback not set
```



## 实验13. 用 3 台交换机验证 PVST 实验

实验拓扑：



实验要求：

1. 在 SW1, SW2, SW3 上创建 VLAN2 , VLAN3 , VLAN4.
2. 确保核心交换机 SW1 为 VLAN1-4 的根网桥, 当 SW1 出现故障时 SW2 成为 VLAN1-4 的 根网桥.
3. 确保交换机 SW3 的 VLAN1, VLAN2 到核心网络 (SW1, SW2) 的流量走线路 1
4. 确保交换机 SW3 的 VLAN3, VLAN4 到核心网络 (SW1, SW2) 的流量走线路 2

实验步骤：

一、实现实验要求 1：

1. 在 SW1, SW2, SW3 上创建 VLAN2 , VLAN3 , VLAN4.

配置 SW1:

创建 VLAN2-4

SW1(config)#vlan 2-4

查看 VLAN:

SW1#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, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24, Gi0/1, Gi0/2
2 VLAN0002	active	
3 VLAN0003	active	
4 VLAN0004	active	

配置 SW2:

创建 VLAN2-4

SW2(config)#vlan 2-4

查看 VLAN:

SW2#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, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18

Fa0/19, Fa0/20, Fa0/21, Fa0/22  
Fa0/23, Fa0/24, Gi0/1, Gi0/2

2	VLAN0002	active
3	VLAN0003	active
4	VLAN0004	active

配置 SW2:

创建 VLAN2-4

SW3(config)#vlan 2-4

查看 VLAN: SW3#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, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24
2 VLAN0002	active	
3 VLAN0003	active	
4 VLAN0004	active	

二、实现实验要求 2:

要求 2. 确保核心交换机 SW1 为 VLAN1-4 的根网桥, 当 SW1 出现故障时 SW2 成为 VLAN1-4 的根网桥.

配置 SW1: (确保核心交换机 SW1 为 VLAN1-4 的根网桥)

SW1(config)#spanning-tree vlan 1-4 root primary

或

SW1(config)#spanning-tree vlan 1-4 priority 24576

(root primary= priority 24576)

配置 SW2: (当 SW1 出现故障时 SW2 成为 VLAN1-4 的根网桥)

SW2(config)#spanning-tree vlan 1-4 root secondary

或

SW2(config)#spanning-tree vlan 1-4 priority 28672

(root secondary= priority 28672)

三、实现实验要求 3:

要求 3: 确保交换机 SW3 的 VLAN1, VLAN2 到核心网络 (SW1, SW2) 的流量走线路 1

SW3#show spanning-tree vlan 1

VLAN0001

Spanning tree enabled protocol ieee

Root ID Priority 24577

Address 000d.bcb4.c500

Cost 19

Port 1 (FastEthernet0/1)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 28673 (priority 28672 sys-id-ext 1) Address

0013.1a9a.2b80

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300

Interface Role Sts Cost Prio.Nbr Type

Fa0/1 Root FWD 19 128.1 P2p Fa0/2 Altn BLK 19 128.2

## P2p SW3#show spanning-tree vlan 2

VLAN0002

Spanning tree enabled protocol ieee

```

Root ID    Priority    24578
           Address    000d.bcb4.c500
           Cost      19
           Port      1 (FastEthernet0/1)
           Hello Time 2 sec   Max Age 20 sec   Forward Delay 15 sec
Bridge ID  Priority    32770 (priority 32768 sys-id-ext 2) Address    0013.1a9a.2b80
           Hello Time 2 sec   Max Age 20 sec   Forward Delay 15 sec
           Aging Time 300

```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/1	Root FWD	19	128.1	P2p	Fa0/2	Altn BLK	19	128.2	P2p
-------	----------	----	-------	-----	-------	----------	----	-------	-----

通过在 SW3 上查看 VLAN1, VLAN2 的生成树状态,发现相对于 VLAN1, VLAN2 来说 F0/1 为转发状态, F0/2 为 blocking 状态, 可以确定 VLAN1 和 VLAN2 到达核心网络的流量通过 F0/1 走线路 1, 已符合本实验的要求, 不需要做配置。

四、实现实验要求 4:

要求 4: 确保交换机 SW3 的 VLAN3, VLAN4 到核心网络 (SW1, SW2) 的流量走线路 2

配置 SW3:

SW3(config)#int f0/1

SW3(config-if)#spanning-tree vlan 3-4 cost 39

配置后, 确认交换机 SW3 的 VLAN3, VLAN4 到核心网络 (SW1, SW2) 的流量是否走线路 2:

SW3#show spanning-tree vlan 3

VLAN0003

Spanning tree enabled protocol ieee

```

Root ID    Priority    24579
           Address    000d.bcb4.c500
           Cost      38
           Port      2 (FastEthernet0/2)
           Hello Time 2 sec   Max Age 20 sec   Forward Delay 15 sec
Bridge ID  Priority    32771 (priority 32768 sys-id-ext 3) Address    0013.1a9a.2b80
           Hello Time 2 sec   Max Age 20 sec   Forward Delay 15 sec
           Aging Time 300

```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/1	Altn BLK	39	128.1	P2p	Fa0/2	Root FWD	19	128.2	P2p
-------	----------	----	-------	-----	-------	----------	----	-------	-----

SW3#show spanning-tree vlan 4

VLAN0004

Spanning tree enabled protocol ieee

```

Root ID    Priority    24580
           Address    000d.bcb4.c500
           Cost      38
           Port      2 (FastEthernet0/2)
           Hello Time 2 sec   Max Age 20 sec   Forward Delay 15 sec
Bridge ID  Priority    32772 (priority 32768 sys-id-ext 4) Address    0013.1a9a.2b80
           Hello Time 2 sec   Max Age 20 sec   Forward Delay 15 sec
           Aging Time 300

```

Interface	Role	Sts	Cost	Prio.	Nbr	Type
-----------	------	-----	------	-------	-----	------

Fa0/1	Altn BLK	39	128.1	P2p	Fa0/2	Root FWD	19	128.2	P2p
-------	----------	----	-------	-----	-------	----------	----	-------	-----

Fa0/2      Root FWD 19      128.2      P2p

结果：通过在 SW3 上查看 VLAN3 , VALN4 的生成树状态,发现相对于 VLAN3,VLAN4 来说 F0/2 为转发状态,F0/1 为 blocking 状态,可以确定 VLAN3 和 VLAN4 到达核心网络的流量通过 F0/2 走线路 2, 符合本实验的要求。

附：根网桥和备份根网桥和非根网桥 SW3 的确认信息：

根网桥确认信息：

SW1#show spanning-tree vlan 1

VLAN0001

Spanning tree enabled protocol ieee

Root ID      Priority      24577

Address      000d.bcb4.c500

**This bridge is the root**

Hello Time    2 sec    Max Age 20 sec    Forward Delay 15 sec

Bridge ID    Priority      24577 (priority 24576 sys-id-ext 1) Address

000d.bcb4.c500

Hello Time    2 sec    Max Age 20 sec    Forward Delay 15 sec

Aging Time 300

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Desg	FWD	19	128.1	P2p
Fa0/2	Desg	FWD	19	128.2	P2p

SW1#show spanning-tree vlan 2

VLAN0002

Spanning tree enabled protocol ieee

Root ID      Priority      24578

Address      000d.bcb4.c500

**This bridge is the root**

Hello Time    2 sec    Max Age 20 sec    Forward Delay 15 sec

Bridge ID    Priority      24578 (priority 24576 sys-id-ext 2) Address

000d.bcb4.c500

Hello Time    2 sec    Max Age 20 sec    Forward Delay 15 sec

Aging Time 300

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Desg	FWD	19	128.1	P2p
Fa0/2	Desg	FWD	19	128.2	P2p

SW1#show spanning-tree vlan 3

VLAN0003

Spanning tree enabled protocol ieee

Root ID      Priority      24579

Address      000d.bcb4.c500

**This bridge is the root**

Hello Time    2 sec    Max Age 20 sec    Forward Delay 15 sec

Bridge ID    Priority      24579 (priority 24576 sys-id-ext 3) Address

000d.bcb4.c500

Hello Time    2 sec    Max Age 20 sec    Forward Delay 15 sec

Aging Time 300

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Desg	FWD	19	128.1	P2p
Fa0/2	Desg	FWD	19	128.2	P2p

SW1#show spanning-tree vlan 4

VLAN0004

Spanning tree enabled protocol ieee

Root ID Priority 24580

Address 000d.bcb4.c500

**This bridge is the root**

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 24580 (priority 24576 sys-id-ext 4) Address 000d.bcb4.c500

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

Fa0/1	Desg	FWD	19	128.1	P2p
-------	------	-----	----	-------	-----

Fa0/2	Desg	FWD	19	128.2	P2p
-------	------	-----	----	-------	-----

备份根网桥的确认信息：

SW2#show spanning-tree vlan 1

VLAN0001

Spanning tree enabled protocol ieee

Root ID Priority 24577

Address 000d.bcb4.c500

Cost 19

Port 1 (FastEthernet0/1)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 28673 (priority 28672 sys-id-ext 1)

Address 0011.92e4.2780

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

Fa0/1	Root	FWD	19	128.1	P2p
-------	------	-----	----	-------	-----

Fa0/2	Desg	FWD	19	128.2	P2p
-------	------	-----	----	-------	-----

SW2#show spanning-tree vlan 2

VLAN0002

Spanning tree enabled protocol ieee

Root ID Priority 24578

Address 000d.bcb4.c500

Cost 19

Port 1 (FastEthernet0/1)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 28674 (priority 28672 sys-id-ext 2)

Address 0011.92e4.2780

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300

Interface	Role	Sts	Cost	Prio.Nbr	Type
-----------	------	-----	------	----------	------

Fa0/1	Root	FWD	19	128.1	P2p
-------	------	-----	----	-------	-----

Fa0/2	Desg	FWD	19	128.2	P2p
-------	------	-----	----	-------	-----

SW2#show spanning-tree vlan 3

VLAN0003

Spanning tree enabled protocol ieee

Root ID Priority 24579

```

Address      000d.bcb4.c500
Cost         19
Port         1 (FastEthernet0/1)
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Bridge ID    Priority 28675 (priority 28672 sys-id-ext 3)
Address      0011.92e4.2780
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Aging Time   300
Interface    Role Sts Cost      Prio.Nbr Type
-----
Fa0/1        Root FWD 19      128.1   P2p
Fa0/2        Desg FWD 19      128.2   P2p

```

SW2#show spanning-tree vlan 4

VLAN0004

Spanning tree enabled protocol ieee

```

Root ID      Priority    24580
Address      000d.bcb4.c500
Cost         19
Port         1 (FastEthernet0/1)
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Bridge ID    Priority 28676 (priority 28672 sys-id-ext 4)
Address      0011.92e4.2780
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Aging Time   300

```

```

Interface    Role Sts Cost      Prio.Nbr Type
-----
Fa0/1        Root FWD 19      128.1   P2p
Fa0/2        Desg FWD 19      128.2   P2p

```

SW2#

非根网桥 **SW3** 的确认信息:

SW3#show spanning-tree vlan 1

VLAN0001

Spanning tree enabled protocol ieee

```

Root ID      Priority    24577
Address      000d.bcb4.c500
Cost         19
Port         1 (FastEthernet0/1)
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Bridge ID    Priority 28673 (priority 28672 sys-id-ext 1) Address
              0013.1a9a.2b80
Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
Aging Time   300

```

```

Interface    Role Sts Cost      Prio.Nbr Type
-----
Fa0/1        Root FWD 19      128.1   P2p
Fa0/2        Altn BLK 19      128.2   P2p

```

SW3#show spanning-tree vlan 2

VLAN0002

Spanning tree enabled protocol ieee

```

Root ID      Priority    24578
Address      000d.bcb4.c500

```

	Cost	19		
	Port	1 (FastEthernet0/1)		
	Hello Time	2 sec	Max Age 20 sec	Forward Delay 15 sec
Bridge ID	Priority	32770	(priority 32768 sys-id-ext 2)	Address 0013.1a9a.2b80
	Hello Time	2 sec	Max Age 20 sec	Forward Delay 15 sec
	Aging Time	300		
Interface	Role	Sts Cost	Prio.Nbr	Type
-----				
<b>Fa0/1</b>	<b>Root FWD</b>	<b>19</b>	<b>128.1</b>	<b>P2p</b>
<b>Fa0/2</b>	<b>Altn BLK</b>	<b>19</b>	<b>128.2</b>	<b>P2p</b>

SW3#show spanning-tree vlan 3

VLAN0003

Spanning tree enabled protocol ieee

Root ID Priority 24579

Address 000d.bcb4.c500

Cost 38

Port 2 (FastEthernet0/2)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32771 (priority 32768 sys-id-ext 3) Address 0013.1a9a.2b80

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Aging Time 300

Interface Role Sts Cost Prio.Nbr Type

-----				
<b>Fa0/1</b>	<b>Altn BLK</b>	<b>39</b>	<b>128.1</b>	<b>P2p</b>
<b>Fa0/2</b>	<b>Root FWD</b>	<b>19</b>	<b>128.2</b>	<b>P2p</b>

SW3#show spanning-tree vlan 4

VLAN0004

Spanning tree enabled protocol ieee

Root ID Priority 24580

Address 000d.bcb4.c500

Cost 38

Port 2 (FastEthernet0/2)

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

Bridge ID Priority 32772 (priority 32768 sys-id-ext 4) Address 0013.1a9a.2b80

Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec

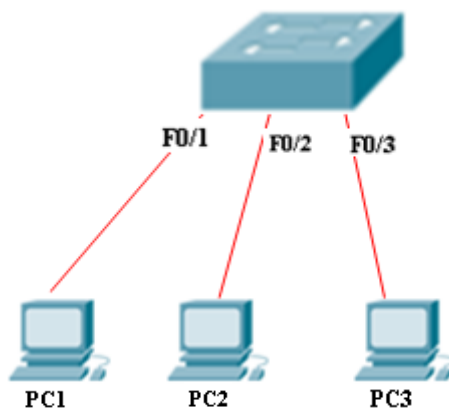
Aging Time 300

Interface Role Sts Cost Prio.Nbr Type

-----				
<b>Fa0/1</b>	<b>Altn BLK</b>	<b>39</b>	<b>128.1</b>	<b>P2p</b>
<b>Fa0/2</b>	<b>Root FWD</b>	<b>19</b>	<b>128.2</b>	<b>P2p</b>

## 实验14. 配置交换机的端口安全(1)

实验拓扑：



实验需求：

F0/1 的已划分到 VLAN2，配置 F0/1 的端口安全，使得端口 F0/1 只允许 PC1 的 MAC 地址

(aaaa.aaaa.aaa1) 进入，如果接收到违规的包（也就是进入 F0/1 非 MAC: aaaa.aaaa.aaa1 的数据包）接口会 shutdown. 实

验步骤： 第一

步：

创建 VLAN2，把 F0/1 加入 VLAN2

```
SW2950(config)#vlan 2
```

```
SW2950(config-vlan)#exit
```

```
SW2950(config)#int f0/1
```

```
SW2950(config-if)#switchport mode access
```

```
SW2950(config-if)#switchport access vlan 2
```

第二步：

设置 F0/1 的端口安全：

```
SW2950(config)#int f0/1
```

```
SW2950(config-if)#switchport port-security
```

-----注：启用端口安全

```
SW2950(config-if)#switchport port-security maximum 1
```

-----注：允许进入 F0/1 的 MAC 地址的最大数目，为 1，这是默认值。

```
SW2950(config-if)#switchport port-security mac-address aaaa.aaaa.aaa1
```

-----注：设置所允许的具体 MAC 地址

```
SW2950(config-if)#switchport port-security violation shutdown
```

-----注：当接收到不是允许的 MAC 时的动作为 shutdown，这是默认设置  
查看以上配置

```
SW2950#show run int f0/1
```

```
Building configuration...
```

```
Current configuration : 163 bytes
```

```
!
```



```
interface FastEthernet0/1
  switchport access vlan 2
switchport mode access
switchport port-security
switchport port-security mac-address aaaa.aaaa.aa1
end
SW1#show port-security
```

-----注： 通过查看配置发现上面我们配置了 4 条端口安全命令，但只看到两条，因为有其中两条是只要启用了端口安全，就默认设置了，不会显示，可以 show port-security 看到

Secure Port	MaxSecureAddr (Count)	CurrentAddr (Count)	SecurityViolation (Count)	Security Action
-------------	--------------------------	------------------------	------------------------------	-----------------

Fa0/10	1	1	0	Shutdown
--------	---	---	---	----------

Total Addresses in System (excluding one mac per port): 0

Max Addresses limit in System (excluding one mac per port) : 1024

端口安全的另外两个动作参数：(protect 和 restrict)

SW1(config-if)#switchport port-security violation ?

**protect** Security violation protect mode

-----注： Protect 意思是当 F0/1 接收到不是所允许的 MAC 地址时动作作为做保护，不会关闭接口，但数据包被拒绝通过，接口的状态是好的，不给管理员任何提示，不产生告警。

**restrict** Security violation restrict mode

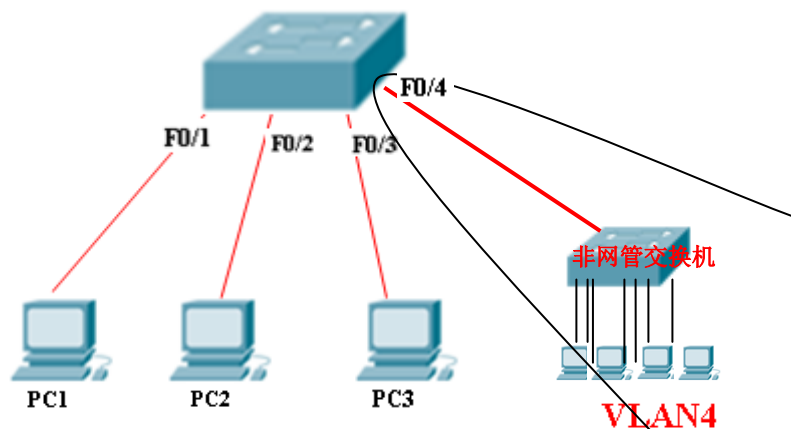
-----注： restrict 意思是当 F0/1 接收到不是所允许的 MAC 地址时动作作为做限制，不会关闭接口，但数据包被拒绝通过，接口的状态是好的，但会给管理员提示信息，产生告警。

**shutdown** Security violation shutdown mode

-----注： shutdown 意思是接收到不是所允许的 MAC 地址时动作关闭接口，并产生告警，被 shutdown 的接口要 shutdown 再 no shutdown,接口状态才会再次 up

## 实验15. 配置交换机的端口安全(2)

实验拓扑:



实验拓扑描述: SW2950 的 F0/4 下接了一个非网管的交换机。 实验需求:

F0/4 的已划分到 VLAN4, 配置 F0/4 的端口安全, 使得端口 F0/4 最多允许 10 个 MAC 进入, 如果接收到违规的包 (也就是进入 F0/4 接口的 MAC 地址超过 10 个, 则丢弃违规的数据包, 并且产生告警。 实验步骤:

第一步:

创建 VLAN4, 把 F0/4 加入 VLAN4

```
SW2950(config)#vlan 4
```

```
SW2950(config)#int f0/4
```

```
SW2950(config-if)#switchport mode access
```

```
SW2950(config-if)#switchport access vlan 4
```

-----注: 把端 F0/4 加入到 VLAN4

```
SW2950(config-if)#switchport port-security
```

```
SW2950(config-if)#switchport port-security maximum 10
```

```
SW2950(config-if)#switchport port-security mac-address sticky
```

```
SW2950(config-if)#switchport port-security violation restrict
```

-----注: 启用端口安全, 进入接口最大的 MAC 地址数, sticky 命令作用是会记录前 10 个合法的 MAC 地址, 超过则被丢弃。 Restrict 的作为违规的数据包 不可以通过, 会产生告警。

通过 show run 查看以上所做的配置:

```
SW2950#sho run int f0/4
```

```
interface FastEthernet0/4
```

```
switchport access vlan 4
```

```
switchport mode access
```

```
switchport port-security
```

```
switchport port-security maximum 10
```

```
switchport port-security violation restrict
```

```
switchport port-security mac-address sticky
```

查看端口安全参数:

```
SW2#show port-security
```

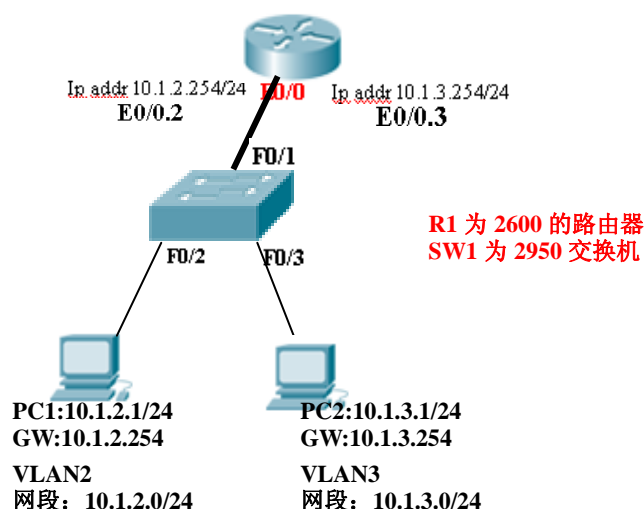
Secure Port	MaxSecureAddr (Count)	CurrentAddr (Count)	SecurityViolation (Count)	Security Action
-----				
Fa0/4	10	0	0	Restrict

Total Addresses in System (excluding one mac per port) : 0

Max Addresses limit in System (excluding one mac per port) : 1024

## 实验16. 用路由器实现 VLAN 之间的访问

实验拓扑:



实验要求:

1. 在 SW1 上划分 VLAN2,VLAN3,把 F0/2 加入 VLAN2,F0/3 加入 VLAN3,端口 F0/1 设置为 802.1Q 的 Trunk.
2. 配置 R1,使得 VLAN2 和 VLAN3 的主机之间可以互相访问.

配置步骤:

配置 SW1:

```
SW1(config)#vlan 2-3 -----注: 创建 VLAN2,VLAN3
SW1(config)#int f0/2
SW1(config-if)#switchport mode access
SW1(config-if)#switchport access vlan 2 -----注: 把端口 F0/2 加入 VLAN2
SW1(config)#int f0/3
SW1(config-if)#switchport mode access
SW1(config-if)#switchport access vlan 3 -----注: 把端口 F0/3 加入 VLAN3
SW1(config)#int f0/1
SW1(config-if)#switchport mode trunk -----注: 把端口 F0/1 设置为 trunk 端口
```

配置 R1:

```
R1(config)#inte0/0
R1(config-if)#no shutdown -----注: 启用物理接口 E0/0
R1(config-if)#exit
R1(config)#int e0/0.2
R1(config-subif)#encapsulation dot1Q 2
R1(config-subif)#ip address 10.1.2.254 255.255.255.0 -----注: 创建并进入 E0/0.2 子接口,封装 802.1Q
(dot1Q),2 为 VLAN_ID ,并配置 IP 地址
R1(config-subif)#end
R1#conf t
R1(config)#int e0/0.3
R1(config-subif)#encapsulation dot1Q 3
R1(config-subif)#ip add 10.1.3.254 255.255.255.0
-----注: 创建并进入 E0/0.3 子接口,封装 802.1Q (dot1Q),3 为 VLAN_ID ,并配置 IP 地址
```

查看 E0/0.2 的子接口:

```
R1#sho run int e0/0.2
```

```
interface Ethernet0/0.2
 encapsulation dot1Q 2
 ip address 10.1.2.254 255.255.255.0
end
```

查看 **E0/0.3** 的子接口:

```
R1#sho run int e0/0.3
```

Building configuration...

Current configuration : 91 bytes

!

```
interface Ethernet0/0.3
 encapsulation dot1Q 3
 ip address 10.1.3.254 255.255.255.0
end
```

检查 **R1** 的接口状态:包括物理接口和子接口:

```
R1#show ip int brief
```

Interface	IP-Address	OK?	Method	Status	Protocol
<b>Ethernet0/0</b>	<b>unassigned</b>	<b>YES</b>	<b>unset</b>	<b>up</b>	<b>up</b>
Ethernet0/0.2	10.1.2.254	YES	manual	up	up
Ethernet0/0.3	10.1.3.254	YES	manual	up	up
Serial0/0	unassigned	YES	unset	administratively down	down
Serial0/1	unassigned	YES	unset	administratively down	down

检查 **SW1 F0/1** 的 **trunk** 状态:

```
SW1#show int f0/1 switchport
```

Name: Fa0/1

**Switchport: Enabled**

Administrative Mode: trunk

**Operational Mode: trunk** -----注: 802.1Q trunk

Administrative Trunking Encapsulation: dot1q

**Operational Trunking Encapsulation: dot1q** -----注: 802.1Q trunk

Negotiation of Trunking: On

Access Mode VLAN: 1 (default)

Trunking Native Mode VLAN: 1 (default) Voice

VLAN: none

Administrative private-vlan host-association: none Administrative

private-vlan mapping: none Administrative private-vlan trunk native

VLAN: none Administrative private-vlan trunk encapsulation: dot1q

Administrative private-vlan trunk normal VLANs: none

Administrative private-vlan trunk private VLANs: none Operational

private-vlan: none

Trunking VLANs Enabled: ALL

Pruning VLANs Enabled: 2-1001

Capture Mode Disabled

Capture VLANs Allowed: ALL Protected:

false

Appliance trust: none

```
SW1#
```

用两台 2500 的路由器作为 PC 测试:

配置 PC1:

```
PC1(config)#no ip routing
```

```
PC1(config)#int e0
PC1(config-if)#ip address 10.1.2.1 255.255.255.0
PC1(config-if)#no sh
PC1(config-if)#exit
PC1(config)#ip default-gateway 10.1.2.254
配置 PC2:
```

```
PC2(config)#no ip routing
PC2(config)#int e0
```

-----注：把路由器作为 PC,关闭路由功能,设置接口 IP 地址和网关PC2(config-if)#ip address 10.1.3.1 255.255.255.0

```
PC2(config-if)#no sh
PC2(config-if)#exit
PC2(config)#ip default-gateway 10.1.3.254
```

-----注：把路由器作为 PC,关闭路由功能,设置接口 IP 地址和网关

测试:

测试 PC1 到网关的连通性:

```
PC1#ping 10.1.2.254
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.2.254, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms

测试 PC1 到 PC2 的连通性:

```
PC1#ping 10.1.3.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.3.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/8 ms

在 PC1 用 traceroute 跟踪到目标主机 10.1.3.1 是否经过网关.

```
PC1#traceroute 10.1.3.1
```

Type escape sequence to abort. Tracing the route to 10.1.3.1

```
1 10.1.2.254 0 msec 4 msec 4 msec
```

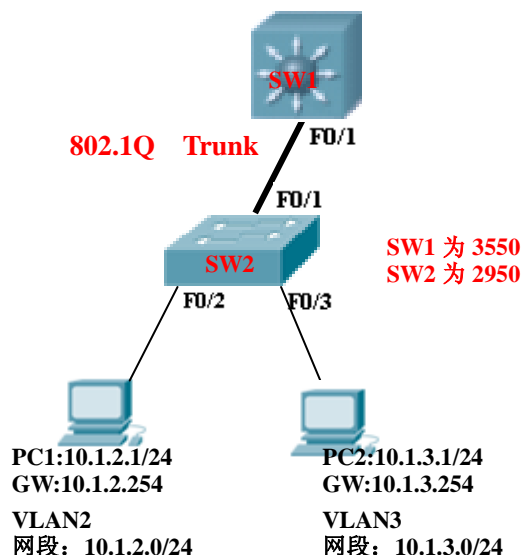
```
2 10.1.3.1 4 msec * 4 msec
```

```
PC1#
```

-----注：经过网关到达目的地址

## 实验17. 三层交换机实现 VLAN 之间互访

实验拓扑:



实验要求:

1. 分别在 SW1 和 SW2 上创建 VLAN2 和 VLAN3，在 SW2 上把 F0/2 加入 VLAN2, F0/3 加入 VLAN3, 端口 F0/1 设置为 802.1Q 的 Trunk.
2. 把 SW1 的 F0/1 设置成 802.1Q 的 Trunk, 启用 SVI 接口, 使得 SW2 的 VLAN2 和 VLAN3 主机之间可以互相访问.

配置步骤:

配置 SW2: (2950 交换机)

```
SW2(config)#vlan 2-3
SW2(config)#int f0/2
SW2(config-if)#switchport mode access
SW2(config-if)#switchport access vlan 2
SW2(config)#int f0/3
SW2(config-if)#switchport mode access
SW2(config-if)#switchport access vlan 3
SW2(config)#int f0/1
SW2(config-if)#switchport mode trunk
```

-----注: 在 SW2 上创建 VLAN2, VLAN3

-----注: 把端口 F0/2 加入到 VLAN2

-----注: 把端口 F0/3 加入到 VLAN3

-----注: 设置 F0/1 为 trunk 端口

配置 SW1: (3550 交换机)

```
SW1(config)#ip routing
SW1(config)#vlan 2-3
SW1(config)#int f0/1
SW1(config-if)#switchport trunk encapsulation dot1q
SW1(config-if)#switchport mode trunk
SW1(config)#int vlan 2
SW1(config-if)#ip address 10.1.2.254 255.255.255.0
SW1(config-if)#no sh
```

-----注: 启用三层交换机的路由功能, 默认是关闭的

-----注: 创建 VLAN2, VLAN3

-----注: 设置 F0/1 的 802.1Q trunk, 必须要先封装再设置为 trunk 模式

-----注: 创建 SVI 2 的三层接口并配置 IP 地址, 并启用接口, 这个接口 IP 作为 VLAN2 主机的网关

```
SW1(config)#int vlan 3
```

```
SW1(config-if)#ip address 10.1.3.254 255.255.255.0
```

SW1(config-if)#no sh -----注：创建 SVI 3 的三层接口并配置 IP 地址，并启用接口，这个接口 IP 作为 VLAN3 主机的网关在 SW1 上用 show 命令查看刚才的配置：

```
SW1#show run int vlan 2
```

```
interface Vlan2
```

```
ip address 10.1.2.254 255.255.255.0
```

```
end
```

```
SW1#show int vlan 2
```

```
Vlan2 is up, line protocol is up
```

```
Hardware is EtherSVI, address is 000d.bcb4.c500 (bia 000d.bcb4.c500) Internet  
address is 10.1.2.254/24
```

```
SW1#show run int vlan 3
```

```
interface Vlan3
```

```
ip address 10.1.3.254 255.255.255.0
```

```
end
```

```
SW1#show int vlan 3
```

```
Vlan3 is up, line protocol is up
```

```
Hardware is EtherSVI, address is 000d.bcb4.c500 (bia 000d.bcb4.c500)
```

```
Internet address is 10.1.3.254/24
```

```
SW1#show int f0/1 switchport
```

```
Name: Fa0/1
```

```
Switchport: Enabled
```

```
Administrative Mode: trunk
```

```
Operational Mode: trunk
```

```
Administrative Trunking Encapsulation: dot1q
```

```
Operational Trunking Encapsulation: dot1q
```

```
SW1#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, E - EGP
```

```
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
```

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

```
C 10.1.3.0 is directly connected, Vlan3
```

```
C 10.1.2.0 is directly connected, Vlan2
```

在 SW2 上用 show 命令查看刚才的配置：

```
SW2#show run int f0/2
```

```
interface FastEthernet0/2
```

```
switchport access vlan 2
```

```
switchport mode access
```

```
end
```

```
SW2#show run int f0/3
```

```
interface FastEthernet0/3
```

```
switchport access vlan 3
```

```
switchport mode access
```

```
end
```

```
SW2#show int f0/1 switchport
```

```
Name: Fa0/1
```

```
Switchport: Enabled
```



Administrative Mode: trunk

Operational Mode: trunk

Administrative Trunking Encapsulation: dot1q

Operational Trunking Encapsulation: dot1q

Negotiation of Trunking: On

Access Mode VLAN: 1 (default)

Trunking Native Mode VLAN: 1 (default) Voice VLAN: none

Administrative private-vlan host-association: none Administrative private-vlan mapping: none Administrative private-vlan

trunk native VLAN: none Administrative private-vlan trunk encapsulation: dot1q Administrative private-vlan trunk normal

VLANs: none Administrative private-vlan trunk private VLANs: none Operational private-vlan: none

Trunking VLANs Enabled: ALL

Pruning VLANs Enabled: 2-1001

Capture Mode Disabled

Capture VLANs Allowed: ALL Protected: false

Appliance trust: none

测试：本实验用两台 2500 的路由器作为 PC 测试。

配置 PC1:

PC1(config)#no ip routing

PC1(config)#int e0

PC1(config-if)#ip address 10.1.2.1 255.255.255.0

PC1(config-if)#no sh

PC1(config)#ip default-gateway 10.1.2.254

-----注：把路由器作为 PC,关闭路由功能,设置接口 IP 地址和网关

配置 PC2:

PC2(config)#no ip routing

PC2(config)#int e0

PC2(config-if)#ip address 10.1.3.1 255.255.255.0

PC2(config-if)#no sh

PC2(config)#ip default-gateway 10.1.3.254

-----注：把路由器作为 PC,关闭路由功能,设置接口 IP 地址和网关

测试 PC1 到网关的连通性:

PC1#ping 10.1.2.254

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.2.254, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/4 ms

测试 PC1 到 PC2 的连通性:

PC1#ping 10.1.3.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.3.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 4/4/8 ms

在 PC1 用 **tracert** 跟踪到目标主机 10.1.3.1 是否经过网关.

PC1#tracert 10.1.3.1

Type escape sequence to abort. Tracing

the route to 10.1.3.1

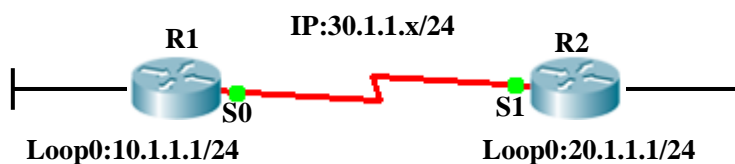
1 **10.1.2.254** 0 msec 4 msec 4 msec

2 **10.1.3.1** 4 msec \* 4 msec

-----注：经过网关到达目的地址

## 实验18. 配置静态路由

实验拓扑：



试验环境：两台2500系列路由器通过串口相连。

要求：在R1和R2上启用静态路由，使两端网络能够互通。

步骤一、基本配置

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

步骤二:接口配置

R1的配置：

```
R1(config)#interface s0
R1(config-if)#ip address 30.1.1.1 255.255.255.0
R1(config-if)#clock rate 64000    →时钟速率端
R1(config-if)#no shutdown
R1(config-if)#interface loopback 0
R1(config-if)#ip address 10.1.1.1 255.255.255.0
R1(config-if)#
```

R2的配置：

```
Router(config)#hostname R2
R2(config)#interface s1
R2(config-if)#ip address 30.1.1.2 255.255.255.0
R2(config-if)#no shutdown
R2(config-if)#interface loopback 0
R2(config-if)#ip address 20.1.1.1 255.255.255.0
R2(config-if)#exit
R2(config)#
```

步骤三、指定静态路由

```
R1(config)#
R1(config)#ip route 20.1.1.0 255.255.255.0 serial 0    →指定静态路由
R2(config)#
```

R2(config)#ip route 10.1.1.0 255.255.255.0 30.1.1.1 →指定静态路由

步骤四、作ping 测试

R1#ping

Protocol [ip]:

Target IP address: 20.1.1.1 →目的地址

Extended commands [n]: y →是否用扩展命令

Source address or interface: 10.1.1.1 →源地址

!!!! →显示已通

步骤五、显示静态路由条目

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

20.0.0.0/24 is subnetted, 1 subnets

S 20.1.1.0 is directly connected, Serial0 →静态路由条目

10.0.0.0/24 is subnetted, 1 subnets

C 10.1.1.0 is directly connected, Loopback0

30.0.0.0/24 is subnetted, 1 subnets

C 30.1.1.0 is directly connected, Serial0

步骤六、显示当前配置

R1 的配置结果:

R1#show running-config

!

hostname R1

!

no ip domain-lookup

!

interface Loopback0

ip address 10.1.1.1 255.255.255.0

!

interface Serial0

ip address 30.1.1.1 255.255.255.0

clockrate 64000

!

ip route 20.1.1.0 255.255.255.0 Serial0

!

line con 0

exec-timeout 0 0

logging synchronous

transport input none

R2的配置结果:

R2#show running-config

!

hostname R2

!

no ip domain-lookup

!

interface Loopback0

ip address 20.1.1.1 255.255.255.0

!

interface Serial1

ip address 30.1.1.2 255.255.255.0

!

ip route 10.1.1.0 255.255.255.0 30.1.1.1

!

line con 0

exec-timeout 0 0

logging synchronous

!

end

## 实验 19. 配置 RIP

### 1.1 实验目的

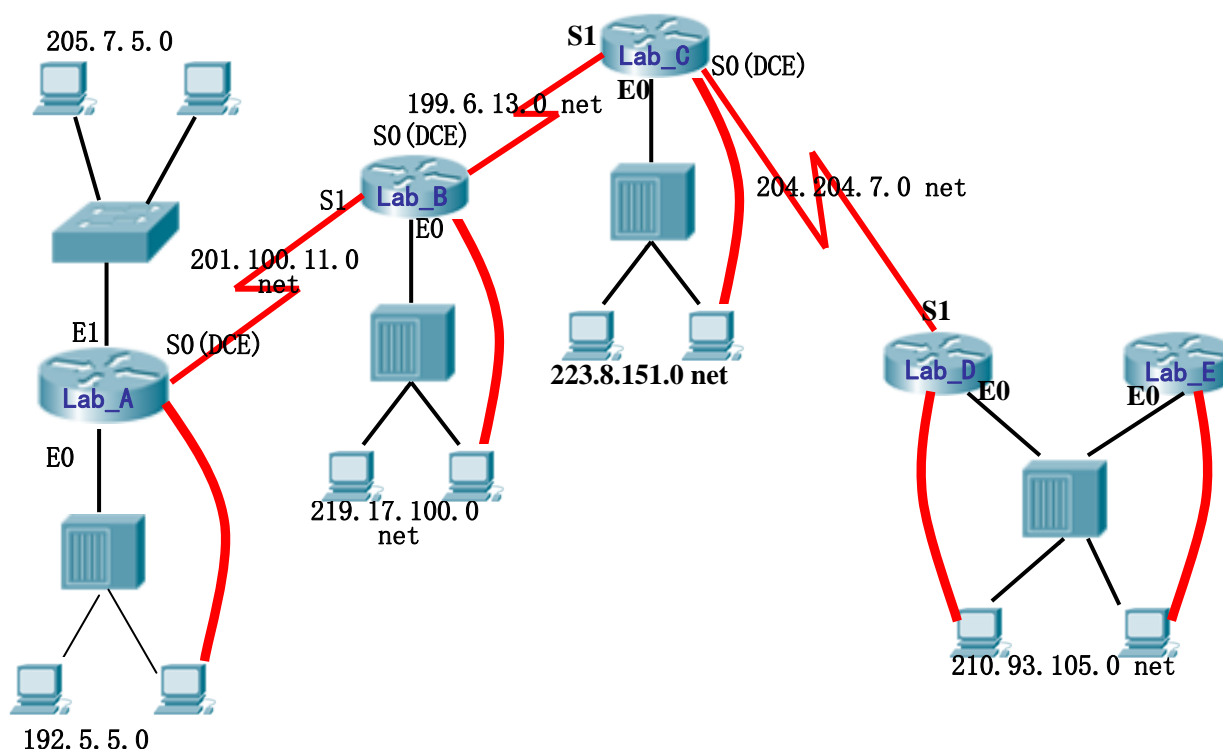
学会用 RIP 协议配置动态路由；

### 1.2 实验内容

- 学习关于 RIP 一些基本知识；
- 了解 RIP 协议的特点；
- 明确 RIP 缺点和优点；
- 学习如何配置基本 RIP 命令；
- 知道使用 RIP 协议所带来的问题。

### 1.3 实验步骤

按照下面的指示，连接网络设备



Router Name	Router Type	E0	E1	S0	S1	SM	Enable Pass-word	Vty Pass-word
Lab_A	2514	192.5.5.1	205.7.5.1	201.100.11.1	--	255.255.255.0	class	cisco
Lab_B	2501	219.17.100.1	--	199.6.13.1	201.100.11.2	255.255.255.0	class	cisco
Lab_C	2501	223.8.151.1	--	204.204.7.1	199.6.13.2	255.255.255.0	class	cisco
Lab_D	2501	210.93.105.1	--	--	204.204.7.2	255.255.255.0	class	cisco
Lab_E	2501	210.93.105.2	--	--	--	255.255.255.0	class	cisco

- 1、在全局配置模式下键入 `router rip`（启动路由协议）；
- 2、在（`config-router`）模式下键入 `network network number`(选择相关的网络或子网络号)；
- 3、保存当前配置（使用 `copy run start`）。

### 1.4 实验参考：

列出本小组所用的网络号清单：

```
RouterA(config)#router rip
RouterA(config-router)#network 201.100.11.0
RouterA(config-router)#network 192.5.5.0
```

```
RouterB(config)#router rip
RouterB(config-router)#network 201.100.11.0
RouterB(config-router)#network 199.6.13.0
RouterB(config-router)#network 219.17.100.0
```

```
RouterC(config)#router rip
RouterC(config-router)#network 199.6.13.0
RouterC(config-router)#network 204.204.7.0
RouterC(config-router)#network 223.8.151.0
```

```
RouterD(config)#router rip
RouterD(config-router)#network 204.204.7.0
RouterD(config-router)#network 210.93.105.0
```

```
RouterD(config)#router rip
RouterD(config-router)#network 210.93.105.0
```

设置 RIP 协议的过程：

首先配置了各台路由器上的基本设置，比如路由器标识、端口的地址配置；

然后检查各路由器上是否已存在 RIP 路由协议，或其他动态路由协议，或静态协议，如果有，则将这些协议关闭掉；

随后启用 RIP 动态协议，将上面的子网地址配置上去就可以了；

配置完以后，还应进行相应的检查，检查 RIP 协议是否成功的在起作用了，最简单的检查命令是 PING 命令；

如果成功那么配置就算完成了，如果没有成功 PING 通的话，则需要再做具体的检测通过，检查一下物理连接，协议配置是否成功等等的问题。

如下 RIP 协议特性保证网络的稳定性

在 RIP 中，路由器向所有直接相连的网络广播所有它已知的网络；

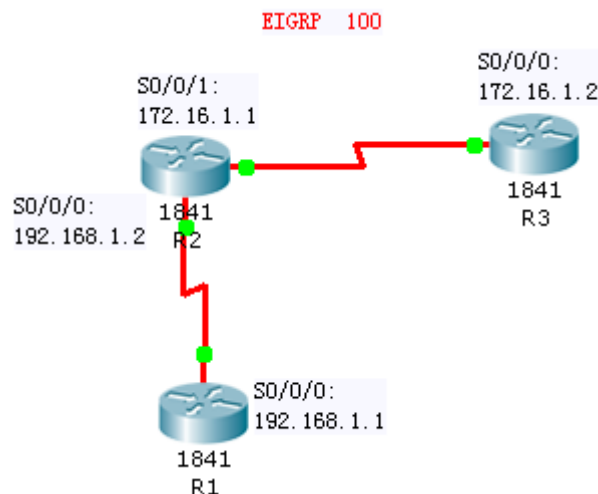
RIP 路由器监听 RIP 广播，从而了解到不能直接知道的那些网络；

RIP 的度量是跳数，RIP 的最大跳数是 15，每个网络的度量在 RIP 广播中公告；

更新定时发送。

## 实验 20. 配置 EIGRP

实验拓扑：



实验目的：

确保路由器之间接口 IP 地址信息已经设置好，能够端到端的 ping 通。

实验步骤：

命令：

1. 定义自治系统（AS）号并启用 EIGRP;

R1(config)#router eigrp {AS-number}

2. 宣告直连主类网络号：

R1(config-router)#network {network-number}

解析：EIGRP 作为 IGRP 的扩展，它同时集合了距离矢量和链路状态路由协议的特点。

参与同一 EIGRP 进程的 EIGRP 路由器必须处于相同的 AS 里，接下来只需要宣告直连的主类网络号即可。

自治系统（AS）：指定区域里，有统一的管理策略，对外表现出一个单一实体的属性，每个自治系统有一个全局唯一的自治系统号。

R1 配置

R1>enable

R1#configure terminal

R1(config)#interface Serial0/0/0

R1(config-if)#ip address 192.168.1.1 255.255.255.0

R1(config-if)#no shutdown

R1(config)#router eigrp 100

R1(config-router)#network 192.168.1.0

R2 配置与 R3 配置跟 R1 类似，同学们自己配置

验证：

R3#show ip route

省略。。。

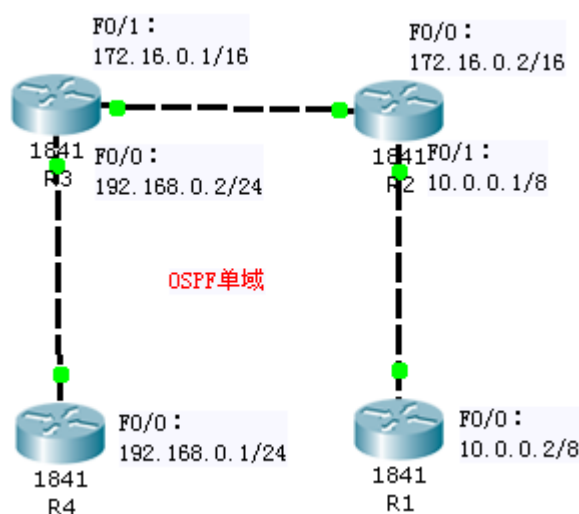
Gateway of last resort is not set

C 172.16.0.0/16 is directly connected, Serial0/0/0

D 192.168.1.0/24 [90/21024000] via 172.16.1.1, 00:00:36, Serial0/0/0

## 实验 21. 配置单区域 OSPF

实验拓扑：



实验目的：

在配置协议之前，要确保路由器之间接口 IP 地址信息已经设置好，能够端到端的 ping 通

实验步骤：

命令模式：

1. 启用 OSPF 进程：

Router(config)#router ospf {process-id}

2. 定义参与 OSPF 进程的接口和网络：

Router(config-router)#network {ip-address} {wildcard-mask} area{area-id}

解析：OSPF 进程 ID 可以使用 1 到 65535 中任何一个整数，改 ID 只是本地的一个标识，即一个 OSPF 网络每台路由器的进程 ID 是否一样，和 OSPF 网络能够正常运行无关，在定义 OSPF 路由器要宣告的区域的时候，反掩码用开控制要宣告的范围，0 表示精确匹配，255 表示任意匹配。OSPF 网络中骨干区域为区域 0，因此必须要有区域 0。

Router1 配置如下：

Router(config)#router ospf 3

Router(config-router)#network 172.16.0.0 0.0.255.255 area 0

Router(config-router)#network 192.168.0.0 0.0.255.255 area 0

其他路由器配置与 Router1 类似，同学们自己配置

测试：

Router#show ip route

```省略

Gateway of last resort is not set

O 10.0.0.0/8 [110/2] via 172.16.0.2, 00:08:48, FastEthernet0/1 通过 OSPF 学习的路由

C 172.16.0.0/16 is directly connected, FastEthernet0/1

C 192.168.0.0/24 is directly connected, FastEthernet0/0

Router#show ip ospf neighbor

| Neighbor ID | Pri | State    | Dead Time | Address     | Interface       |
|-------------|-----|----------|-----------|-------------|-----------------|
| 192.168.0.1 | 1   | FULL/BDR | 00:00:32  | 192.168.0.1 | FastEthernet0/0 |
| 172.16.0.2  | 1   | FULL/BDR | 00:00:32  | 172.16.0.2  | FastEthernet0/1 |

观察、检验 OSPF 配置：

Router#debug ip ospf events //OSPF 协议运作的各种信息

用 no shut 命令打开各路由器上的以太网口。

Router(config-)#int f0

Router(config-if)#no shut

观察路由器输出的 debug 信息。

在各个路由器确立邻居关系之后：

```
Router#show ip ospf neighbor //检查路由器邻接状态
Router#show ip protocols //查看运行的路由协议及协议相关的信息
Router#show ip ospf //查看 OSPF 协议信息及各种计时器
Router#show ip ospf interface e0 //查看 OSPF 的接口相关配置，比如 hello 间隔
Router#debug ip ospf adj //查看邻接关系相关的信息
```

#### 4. 优先修改接口优先级，观察优先级对选举过程的影响

观察默认优先级：R1#show ip ospf interface

观察默认优先级下，router ID 如何影响 DR/BDR 的选举：

```
R1#show ip ospf neighbor
```

修改优先级：

```
路由器 1: R1(config)#int e0
          R1(config-if)#ip ospf priority 0
```

```
路由器 2: R2(config)#int e0
          R2(config-if)#ip ospf priority 1
```

```
路由器 3: R3(config)#int e0
          R3(config-if)#ip ospf priority 2
```

重新开始 DR/BDR 选举过程（把各个路由器的以太网口 shut down，过了 down 机间隔时间之后，重新用 no shut 命令打开）

观察选举结果：R1#show ip ospf neighbor

```
R1#show ip ospf neighbor detail
```

#### 5. 修改 update timer （选做）

查看默认的 hello 间隔和 down 判定间隔：

```
R1#sh ip ospf int e0
```

使用以下命令修改

```
R1#interface e0
R1#ip ospf hello-interval *
R1#ip ospf dead-interval *
```

由于 hello 时间间隔有相等的要求，建议大家先改动一台 router 的参数后，用 debug ip ospf events 命令观察：不一致的 timer 使得 ospf 路由器无法正常通信。

#### 6. 配置认证

在 R1 接口上配置 OSPF 认证口令：

```
R1(config)#int e0
R1(config-if)# ip ospf authentication-key cisco
```

以整个 OSPF 区域为基础启用认证功能：

```
R1(config-router)#area 0 authentication
```

过了 down 机间隔时间之后，在路由器 R1 发出 “show ip ospf neighbor” 命令，查看 R1 的 OSPF 邻居。用 “debug ip ospf events” 确定该结果的原因。

在 R2 与 R3 上作与 R1 相同的配置。

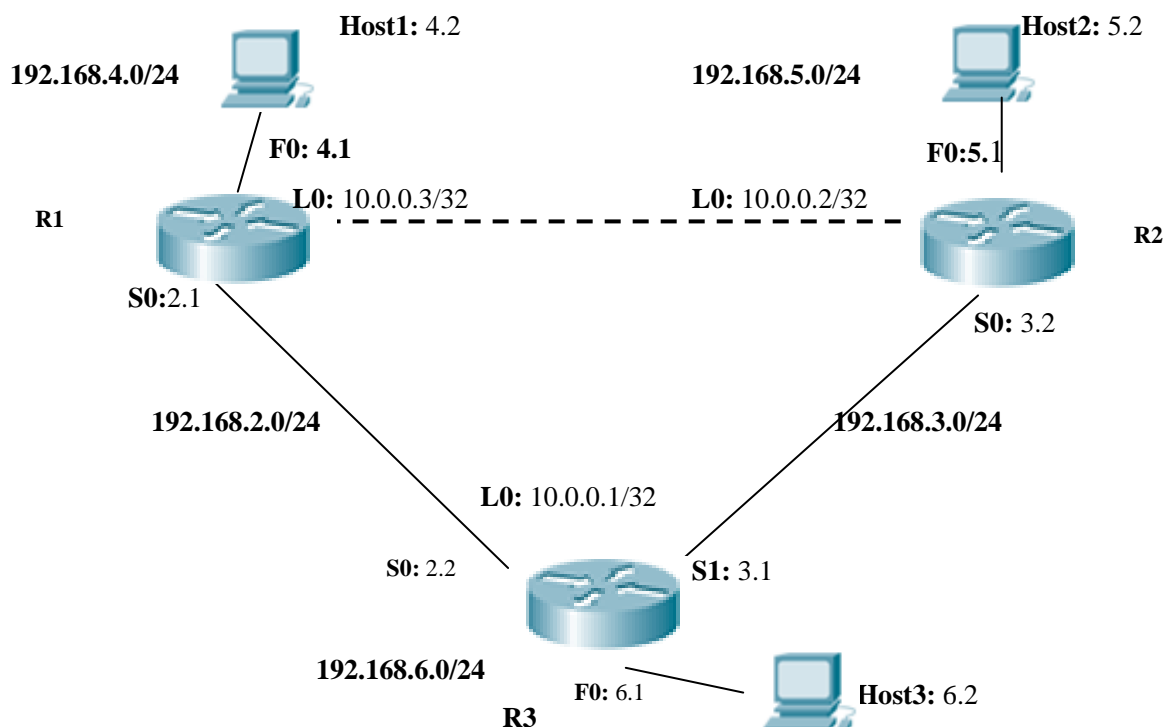
再次用 “show ip ospf neighbor” 来查看结果。

说明：在 debug 信息中，可以看到三种认证类型，0 表示不使用认证，类型 1 代表明文认证，类型 2 代表用 MD5 加密认证。



## 实验 22. 配置标准、扩展、命令 ACL 实验

实验拓扑：



实验目的：

学会配置标准 ACL、扩展 ACL、命名 ACL 以及限制虚拟终端的访问。

实验内容：

在路由器上配置单域的 OSPF

配置 ACL 访问控制列表

实验步骤：

### 配置 ACL 访问控制列表

在完成 OSPF 实验的基础上，修改拓扑结构如下图。其中的虚线只是表示 R1 和 R2 是对等的，它们的功能和配置都基本相同。

修改相应的接口 IP，用 (no) network 命令修改 OSPF 发布的网段信息，进一步熟悉 OSPF 配置，语法如下。以 R1 为例：

```
R1(config)#router ospf 1
R1(config-router)#no network 192.168.1.0 0.0.0.255 area 0
R1(config-router)#network 192.168.4.0 0.0.0.255 area 0
```

配置完 OSPF 之后，检查连通性，各台主机可以相互 ping 通，可以运行 telnet 程序。

### 1. 标准 ACL 的配置

#### (1) 禁止单一的主机

实现目标：禁止网络上某台主机的访问

实现方法：在各台路由器的 f0 口的出方向上绑定 ACL

在路由器 3 上作如下配置：

```
R3(config)#access-list 1 deny host 192.168.4.2 log
R3(config)#access-list 1 deny host 192.168.5.2 log
R3(config)#access-list 1 permit any
R3(config)#int e0
R3(config-if)#ip access-group 1 out
```

检验配置：在路由器 3 上：show run

show access-list 1 //显示访问控制列表的所有条目

show ip int e0 //显示端口上绑定 ACL 的情况

- A. 分别从 Host1 和 Host2 上 ping 192.168.6.2, 结果返回“目标网段不可达”的信息；  
B. 在路由器 1 和 2 上，用扩展 ping 命令，源地址用 e0 的 IP 地址，同样 ping 192.168.6.2，结果是 ping 通的。

- C. 从 Host3 上分别 ping Host1 和 Host2，观察返回信息，思考为什么会有这样的结果。

也可以在 R1 和 R2 上配置标准 ACL：

R1(config)#access-list 1 deny host 192.168.5.2 log //添加 log 关键字是为了记录

R1(config)#access-list 1 deny host 192.168.6.2 log //更详细的信息

R1(config)#access-list 1 permit any

R1(config)#int e0 //注意根据具体的路由器接口而定，譬如有的是 f0 口

R1(config-if)#ip access-group 1 out

R2(config)#access-list 1 deny host 192.168.4.2 log

R2(config)#access-list 1 deny host 192.168.6.2 log

R2(config)#access-list 1 permit any

R2(config)#int e0

R2(config-if)#ip access-group 1 out

重复三个检查步骤，注意返回信息。

最后，在各个路由器上取消 e0 口上的 ACL 绑定：

R1(config)#int e0

R1(config-if)#no ip access-list 1 out

## (2) 禁止整个以太网段

**实现目标：**禁止某个以太网段对 Host1、2、3 的访问。

**实现方法：**分别在两台路由器的 f0 口的出方向上绑定 ACL。

以 RTC 配置为例：

R3(config)#access-list 2 deny 192.168.4.0 0.0.0.255

R3(config)#access-list 2 deny 192.168.5.0 0.0.0.255

R3(config)#access-list 2 permit any

R3(config)#int f0

R3(config-if)#ip access-group 2 out

检验配置：在路由器 C 上：show run

show access-list 2

show ip int e0

- a. 从 Host1、Host2 上 ping 192.168.6.2，结果返回“目标网段不可达”的信息；  
b. 在路由器 A 和 B 上用扩展 ping 命令，源地址用 e0 的 IP，同样 ping 192.168.3.2，结果是不可到达。（与之前的禁止单一主机不同，为什么？）  
c. 在路由器 A 和 B 上用扩展 ping 命令，源地址用 lo0 的 IP，同样再 ping 192.168.3.2，结果是 ping 通的。

同样可以在路由器 1 与 2 上作相同的配置，并检验配置。

最后，在各个路由器上取消 e0 口上的 ACL 绑定：

(config)#int e0

(config-if)#no ip access-list 2 out

## 2. 扩展 ACL 的配置

**实现目标：**禁止主机 telnet 另一主机，但是仍能 ping 通。

**实现方法：**分别在路由器的 f0 口的入方向上绑定 ACL。

以禁止 Host1 远程登陆 Host2 为例：

从 R2 telnet 上 R1，对 R1 进行配置：

R2(config)#telnet 192.168.2.1

R1(config)#access-list 101 deny tcp host 192.168.4.2 host 192.168.5.2 eq telnet

```
R1(config)#access-list 101 permit ip any any
R1(config)#int e0
R1(config-if)#ip access-group 101 in
```

#### 检验配置:

在各路由器上: show access-list 101  
show run  
show ip int f0

从 Host1 上无法 telnet 192.168.5.2, 但可以 ping 通到 192.168.5.2。

同样方法, 可以禁止 Host2 到 Host3, Host3 到 Host1 的 telnet, 请学员参照一下配置完成。

最后在路由器上都取消 e0 口上的 ACL 绑定:

```
Router(config)#int e0
Router(config-if)#no ip access-group 101 in
```

注: eq telnet 可以放置在目的地址或目的掩码后, 也可一放置在源地址或源掩码后。前者表示对目的端口进行匹配, 而后者表示对远端口进行匹配。如下:

```
access-list 101 deny tcp host 192.168.4.2 eq telnet host 192.168.5.2
```

这里表示对主机 192.168.4.2 发出的源端口为 23 的 telnet 数据包进行限制。

### 3. 命名 ACL 的配置 (选做)

命名 ACL 可以实现标准 ACL 或者扩展 ACL, 下面以标准 ACL 为例。

实现目标和实现方法与第 1 步中的标准 ACL 相同, 只是将标准 access-list number 改为命名 ACL。

在路由器 C 上作如下配置:

```
R3(config)#ip access-list standard denyhost
R3(config standard-nacl)#deny host 192.168.4.2 log
R3(config standard-nacl)#deny host 192.168.5.2 log
R3(config standard-nacl)#permit any
R3(config)#int e0
R3(config-if)#ip access-group denyhost out
```

检验配置: 在路由器 3 上: show run

```
show access-list denyhost //现实访问控制列表的所有条目
```

```
show ip int e0 //显示端口上绑定 ACL 的情况
```

从 Host1 和 Host2 都无法 ping 通 Host3。

为了从 Host1 能 ping 通 Host3, 需要把其中一条 deny 语句删除掉, 对于命名 ACL, 可以删除单个条目。

```
RTC(config)#ip access-list standard denyhost
```

```
RTC(config standard-nacl)#no deny host 192.168.4.2 log
```

检查从 Host1 是否可以 ping 通 Host3。

在 R1 与 R2 可以做类似的配置, 体会命名 ACL 特殊作用。

### 4. 限制虚拟终端的访问

在做这个实验前大家先尝试在主机上 telnet 到路由器 R1, 必须先要在路由器上进行以下配置:

```
R1(config)#enable password cisco
R1(config)#line vty 0 4
R1(config-line)#login
R1(config-line)#password cisco
```

确保主机能 telnet 上路由器 1。

在 R3 上进行以下配置:

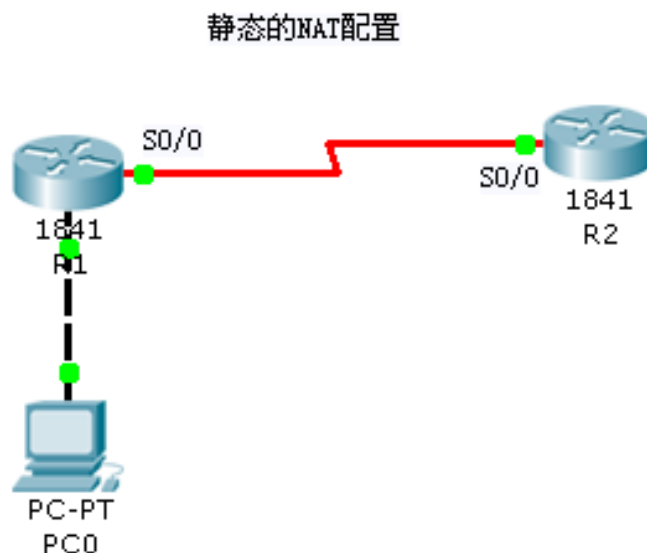
```
R3(config)#access-list 1 deny host 192.168.6.2
R3(config)#access-list 1 permit any
R3(config)#line vty 0 4
R3(config-line)#access-class 1 in
```

尝试从 Host3 telnet R3, 结果是失败的, 证明绑定在虚拟线路上的 ACL 起作用了。

## 实验 23. 静态 NAT 、动态 NAT、PAT 的配置

### 1. 静态 NAT 配置

实验拓扑：



实验目的：

测试内网主机能否成功 ping 通 R2 路由器。

实验步骤：

R1 的配置如下：

```

interface FastEthernet0/1
 ip address 10.1.1.1 255.0.0.0
 ip nat inside
 duplex auto
 speed auto
!
interface Serial0/0
 ip address 192.168.1.1 255.255.255.0
 ip nat outside
!
interface Vlan1
 no ip address
 shutdown
!
ip nat inside source static 10.1.1.2 192.168.1.2
ip classless
!
  
```

R2 的配置如下：

```

interface Serial0/0
 ip address 192.168.1.2 255.255.255.0
 clock rate 64000
  
```

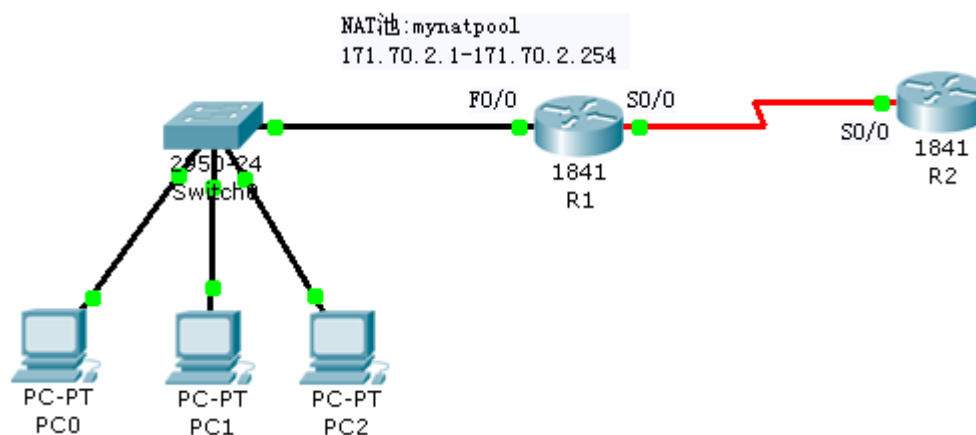
查看地址翻译表

Router#show ip nat translations

| Pro | Inside global | Inside local | Outside local | Outside global |
|-----|---------------|--------------|---------------|----------------|
| --- | 192.168.1.2   | 10.1.1.2     | ---           | ---            |

## 2. 动态 NAT 配置

实验拓扑:



实验目的:

测试内网主机能否成功 ping 通 R2 路由器，并且使用 debug 命令查看数据发送和接受的 IP 转换过程。

实验步骤:

- 1、首先定义 NAT 池。例如：定义一个被内部本地主机使用的叫做 mynatpool 的全局地址池：  
R1(config)#ip nat pool mynatpool 171.70.2.1 171.70.2.254 netmask 255.255.255.0
- 2、为了允许那些本地主机用这个地址池，使用一个访问控制列表来匹配需要转换的源地址。下面是使用 access-list 24 来决定是否使用 mynatpool 转换 IP 源地址：  
R1(config)#access-list 24 permit 10.1.1.0 0.0.0.255  
R1(config)#ip nat inside source list 24 pool mynatpool
- 3、最后配置 NAT 路由器，外部和内部的接口

R1 的配置如下:

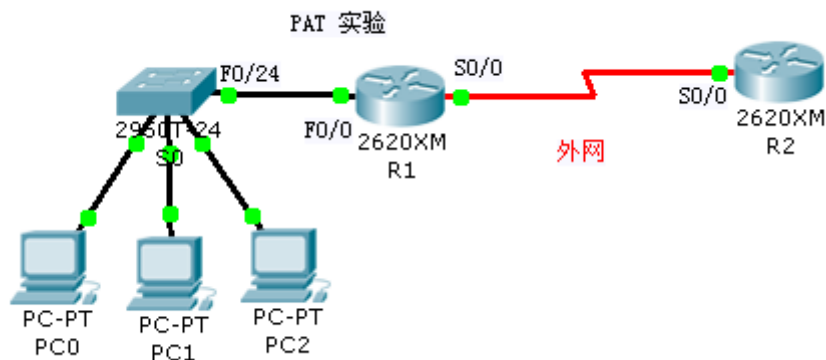
```
interface FastEthernet0/0
 ip address 10.1.1.1 255.255.255.0
 ip nat inside
 duplex auto
 speed auto
!
interface Serial0/0
 ip address 171.70.1.1 255.255.255.0
 ip nat outside
!
ip nat pool mynatpool 171.70.2.1 171.70.2.254 netmask 255.255.255.0
ip nat inside source list 24 pool mynatpool

!
access-list 24 permit 10.1.1.0 0.0.0.255
```

R2 配置如下:

```
interface Serial0/0/0
 ip address 171.70.1.2 255.255.255.0
 clock rate 64000
!
```

## 实验拓扑：



## 实验目的：

进行 NAT 网络试验的条件：

- 1、先决定网络的结构，它是由哪些物理的网络组件构成。（本试验中进行复用动态地址转换《PAT》用到了 3 个计算机、1 个 SWITCH、1 台充当地址转换的路由器和 1 个外部网络路由器。）
- 2、这些网络组件由什么介质进行连接。（计算机与交换机用直通线、交换机与路由器用直通线、路由器与路由器用 DCE 线缆配置。）注意：配置过程中，DCE 端要配置“时钟速率”。

## 实验步骤：

### PAT 试验过程：

1、我们先来配置每台计算机的默认网关、IP 地址和子网掩码。

PC0 网关：192.168.1.254 IP：192.168.1.1 子网：255.255.255.0

PC1 网关：192.168.1.254 IP：192.168.1.2 子网：255.255.255.0

PC2 网关：192.168.1.254 IP：192.168.1.3 子网：255.255.255.0

（在这个试验中交换机只起到组建一个物理网络的作用，所以不需要配置。）

2、在 R1 中先配置路由器名字和三个口令，设置好 R1 的 F0/0 和 S0/0 口的 IP 地址并激活。返回到全局配置模式依次写入：

```
access-list 1 permit 192.168.1.0 0.0.0.255 /定义转发的网段
```

```
ip nat inside source list 1 interface s0/0 overload /将符合的每个内部本地 IP 地址，转换为对应的全局端口的地址。
```

3、int f0/0 /进入 f0/0 端口，这个端口与内部网络相连。

```
ip nat inside /指定连接内部网络的内部端口
```

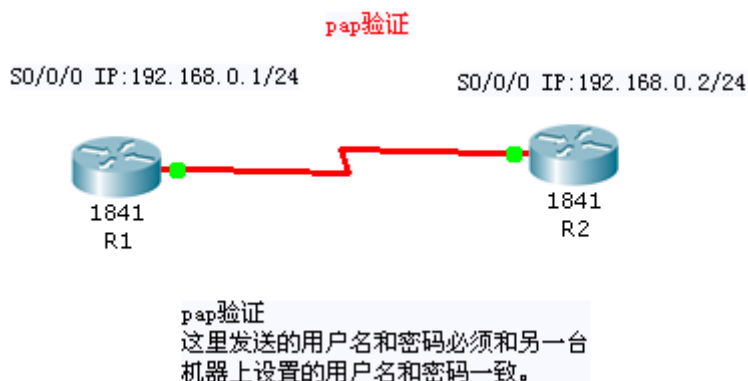
```
int s0/0 /进入 s0/0 端口，这个端口与外部网络相连。
```

```
ip nat outside /指定连接外部网络的外部端口
```

完成试验步骤。

## 实验 24. PPP 的配置

实验拓扑：

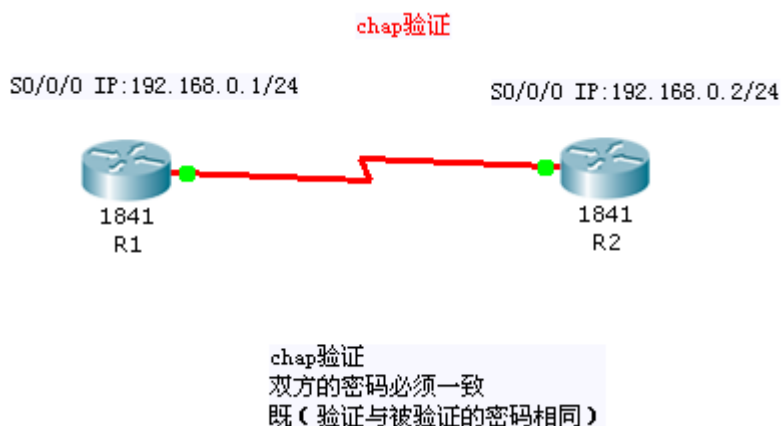


实验步骤：

R1:  
 username aaa password 0 ccc  
 interface Serial0/0/0  
 ip address 192.168.0.1 255.255.255.0  
 no shutdown  
 encapsulation ppp  
 ppp authentication pap  
 ppp pap sent-username bbb password 0 ddd  
 clock rate 64000  
 !

R2:  
 username bbb password 0 ddd  
 interface Serial0/0/0  
 ip address 192.168.0.2 255.255.255.0  
 no shutdown  
 encapsulation ppp  
 ppp authentication pap  
 ppp pap sent-username aaa password 0 ccc  
 !

实验拓扑：



实验步骤：

R1:  
 username aaa password 0 ccc  
 interface Serial0/0/0  
 ip address 192.168.0.1 255.255.255.0  
 no shutdown  
 encapsulation ppp  
 ppp authentication chap  
 clock rate 64000  
 !

R2:  
 username bbb password 0 ccc  
 interface Serial0/0/0  
 ip address 192.168.0.2 255.255.255.0  
 no shutdown  
 encapsulation ppp  
 ppp authentication chap  
 !

## 实验 25. 配置帧中继

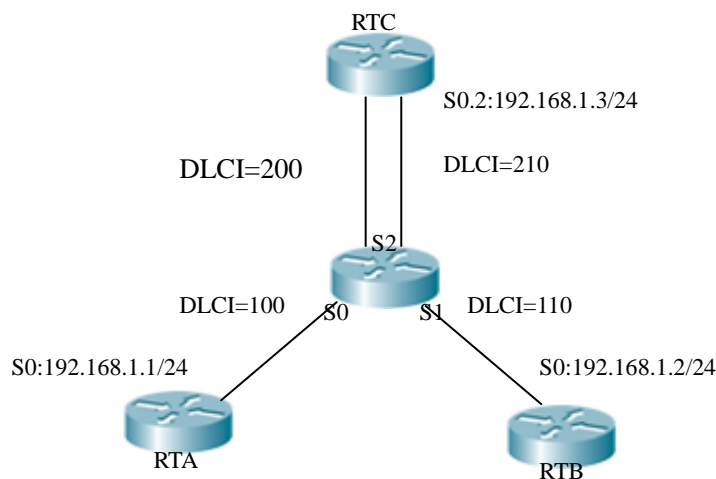
### 实验目的：

了解帧中继技术的原理，掌握 DLCI 和 LMI 的意义和作用；  
掌握帧中继的配置，掌握用路由器模拟帧中继交换机的方法；  
了解子接口的意义，熟悉子接口的配置，并学会在子接口间建立链路；  
掌握各种帧中继的验证和 troubleshooting 命令。

### 实验内容：

#### 帧中继基本配置

#### 1. 实验拓扑



虽然没有 FR 交换机，但我们可以用路由器模拟成帧中继交换机。实验时，用一台 1700 模拟帧中继交换机，并配置其它路由器到 1700 的帧中继连接，分别实现两个路由器的互联。

#### 2. 配置 FR

##### (1) 配置帧中继交换机

##### 要点：

首先要把路由器 1700 设成帧中继交换机，在端口上需要配置其封装为 frame-relay  
在各个端口设置帧中继交换表，并设置端口类型为 DCE。  
注意帧中继交换机的端口不需要配置 ip 地址

```
Switch(config) # frame-relay switching           //模拟成帧中继交换机
Switch(config) # int s0                         //进入 s0 端口
Switch(config-if) # encapsulation frame-relay //配置端口封装为帧中继，默认类型是 cisco
Switch(config-if) # frame-relay intf-type dce   //设置端口类型为 DCE
Switch(config-if) # frame-relay lmi-type cisco  //配置 lmi 类型为 cisco
Switch(config-if) # clock rate 56000
Switch(config-if) # frame-relay route 100 interface s2 200 //建立一条交换记录
Switch(config-if) # no shut
Switch(Config) # int s1                         //进入 s1 端口
Switch(config-if) # encapsulation frame-relay
Switch(config-if) # frame-relay intf-type dce
Switch(config-if) # frame-relay lmi-type cisco
Switch(config-if) # clock rate 56000
Switch(config-if) # frame-relay route 110 interface s2 210 //建立一条交换记录
Switch(config-if) # no shut
Switch(config-if) # exit

Switch(Config) # int s2                         //进入 s1 端口
Switch(config-if) # encapsulation frame-relay
Switch(config-if) # frame-relay intf-type dce
Switch(config-if) # frame-relay lmi-type cisco
Switch(config-if) # clock rate 56000
```



```
Switch(config-if) # frame-relay route 200 interface s0 100
Switch(config-if) # frame-relay route 210 interface s0 110
Switch(config-if) # no shut
Switch(config-if) # exit
```

## (2) 配置用户路由器

要点:

在端口上使用 “ip address <ip> <netmask>” 命令指定 ip 地址

端口上配置封装为 frame-relay

由于 lmi 类型可以通过 LMI 信令自动发现，所以在路由器上可以不进行配置

A 路由器(图中左边为 A 路由器): (1700/1600)

```
RTA(Config) # int s0 //进入 s0 端口
RTA(config-if) # encapsulation frame-relay //设置帧中继封装
RTA(config-if)# ip address 192.168.1.1 255.255.255.0 //指定端口 ip
RTA (config-if)#no shut
```

B、C 路由器与 A 的配置类似

路由器 A、B、C 采用 Inverse-ARP 协议会自动生成帧中继映射表，而不用静态指定。

在对端路由器不支持 inverse arp 的情况下，我们可以采用静态指定 DLCI 和静态生成映射表，则可采用下列命令：

```
RTA(config-if) # frame-relay map ip <下一跳地址> <dli-number>
```

## (3) 验证帧中继

测试连通性：在用户路由器（A/B/C）上使用 ping 命令

查看虚电路：show frame-relay pvc (在帧中继交换机上)

查看映射表：show frame-relay map (在用户路由器上)

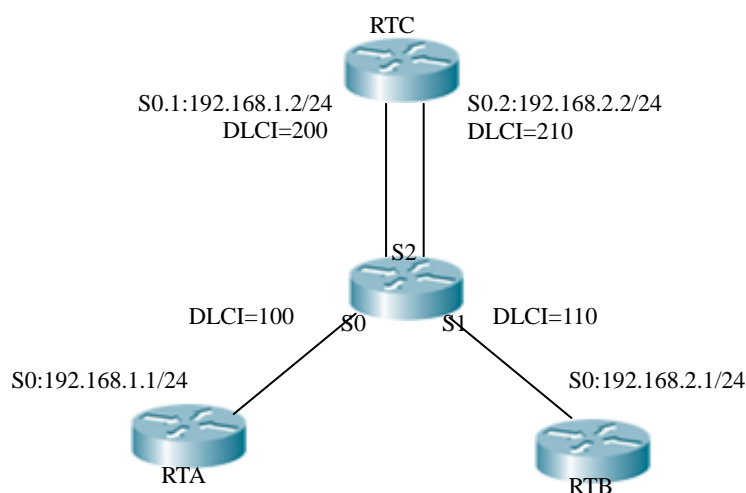
查看 LMI 信息：show frame-relay lmi (在用户路由器上)

查看端口封装：show int s0

查看交换表（只在 FR 交换机上用）：show frame-relay route

使用子接口

### 1. 实验拓扑



目标：在以上步骤的配置下，RTA 与 RTB 由于水平分割的限制而不能交换路由信息，在 RTC 物理接口上使用子接口，可以解决这个问题。

## 2. 观察水平分割问题

修改用户路由器配置之前，观察水平分割对路由更新信息的影响。

在用户路由器 A、B 上配置 Loopback 口，模拟两个网段地址，并启动 RIP 协议：

```
RTA (config) #int l0
RTA(config-if)#ip add 192.168.3.1 255.255.255.0
RTA(config-if)#exit
RTA(config)#router rip
RTA(config-route)#network 192.168.1.0
RTA(config-route)#network 192.168.3.0
RTB(config)#int l0
RTB(config-if)#ip add 192.168.4.1 255.255.255.0
RTB(config-if)#exit
RTB(config)#router rip
RTB(config-route)#network 192.168.1.0
RTB(config-route)#network 192.168.4.0
```

在 A、B 上 show ip route，观察路由表是否有所有网段的路由条目。

## 3. 子接口配置

### (1) 帧中继交换机

与第一步中的交换机一样，不需要作配置上的改动。

### (2) 在用户路由器 C 上启用子接口

要点：

由于要配置子接口，物理端口要用 no ip address 去掉 ip；

进入子接口时需要指明点对点模式，在子接口只需要配置 IP 地址以及 DLCI 号就可以了，帧中继映射表通过 LMI 动态生成。

为查看帧中继 lmi 的工作过程，可以使用 debug frame-relay lmi 命令

## C 路由器

### 物理端口

```
RTC(config) #int s0
RTC(config-if) # no ip address //去掉 IP，否则子接口不起作用
RTC(config-if) # encaps frame-relay //帧中继封装
RTC(config-if) # frame-relay lmi-type cisco //配置 lmi 类型，注意 lmi 类型要匹配
RTC(config-if) # no shut
RTC(config-if) # exit
```

### 子接口 1

```
RTC(config) # int s0.1 point-to-point //进入 s0.1 子接口，注意需要指明是 point-to-point
RTC(config-subif) # ip address 192.168.1.2 255.255.255.0 //设置子接口 ip 地址
RTC(config-subif) # frame-relay interface-dlci 200 //设置本地 dlci 号 RTC(config-subif) #exit
```

### 子接口 2

```
RTC(config) # int s0.1 point-to-point
RTC(config-subif) # ip address 192.168.2.2 255.255.255.0
RTC(config-subif) #frame-relay interface-dlci 210
RTC(config-subif) #exit
```

修改路由器 RTB 的 S0 口的 IP 地址如拓扑图所示。

### (3) 验证配置

查看路由表：show ip route （检查是否有全部网段的信息）

查看虚电路：show frame-relay pvc (在帧中继交换机上)

查看映射表：show frame-relay map （在用户路由器上）

查看端口封装：show int s0

查看 lmi 信息：show frame-relay lmi

查看帧中继 lmi 的工作过程：debug frame-relay lmi