

# CCNA 实验手册

## 目 录

实验一 认识设备端口及连接.....	2
实验二 修改路由器的名称及密码操作.....	6
实验三 基本的操作命令.....	9
实验四 CDP 命令操作.....	11
实验五 备份路由器的 IOS.....	15
实验六 路由器密码的恢复.....	18
实验七 配置 TELNET 远程登陆.....	20
实验八 静态路由.....	22
实验九 RIP 动态路由的配置.....	25
实验十 EIGRP 路由协议.....	30
实验十一 EIGRP 非等价带宽的负载均衡.....	34
实验十二 配置单区域的 OSPF 协议.....	38
实验十三 使用扩展的 ACL 封杀 PING 命令.....	42
实验十四 使用 ACL 禁止 TELNET 应用.....	44
实验十五 NAT 网络地址转换.....	46
实验十六 交换机的基础配置.....	53
实验十七 交换机密码恢复.....	54
实验十八 交换机端口安全.....	57
实验十九 单臂路由.....	58
实验二十 PAP 认证实验.....	61
实验二十一 CHAP 认证实验.....	63
实验二十二 帧中继实验.....	65
实验二十三 用 SDM 管理路由器.....	68

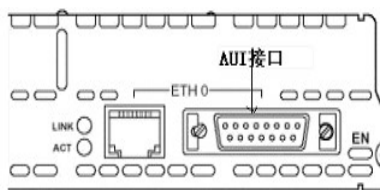
## 实验一 认识设备端口及连接

### 路由器接口

#### 局域网接口

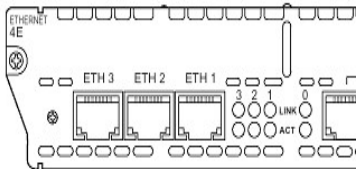
##### (1) AUI 端口

AUI 端口它就是用来与粗同轴电缆连接的接口，它是一种“D”型 15 针接口，这在令牌环网或总线型网络中是一种比较常见的端口之一。路由器可通过粗同轴电缆收发器实现与 10Base-5 网络的连接。但更多的则是借助于外接的收发转发器（AUI-to-RJ-45），实现与 10Base-T 以太网络的连接。当然，也可借助于其他类型的收发转发器实现与细同轴电缆（10Base-2）或光缆（10Base-F）的连接。

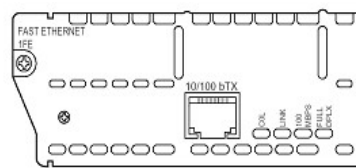


##### (2) RJ-45 端口

RJ-45 端口是我们最常见的端口了，它是我们常见的双绞线以太网端口。因为在快速以太网中也主要采用双绞线作为传输介质，所以根据端口的通信速率不同 RJ-45 端口又可分为 10Base-T 网 RJ-45 端口和 100Base-TX 网 RJ-45 端口两类。其中，10Base-T 网的 RJ-45 端口在路由器中通常是标识为“ETH”，而 100Base-TX 网的 RJ-45 端口则通常标识为“10/100bTX”。

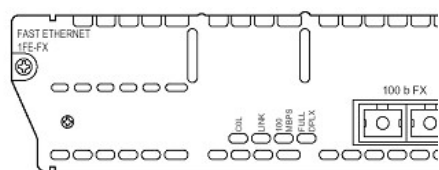


如图 2 所示为 10Base-T 网 RJ-45 端口，而图 3 所示的为 10/100Base-TX 网 RJ-45 端口。其实这两种 RJ-45 端口仅就端口本身而言是完全一样的，但端口中对应的网络电路结构是不同的，所以也不能随便接。



##### (3) SC 端口

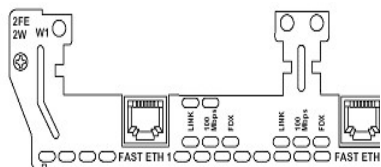
SC 端口也就是我们常说的光纤端口，它是用于与光纤的连接。光纤端口通常是不直接用光纤连接至工作站，而是通过光纤连接到快速以太网或千兆以太网等具有光纤端口的交换机。这种端口一般在高档路由器才具有，都以“100b FX”标注，如图 4 所示。



## 广域网接口

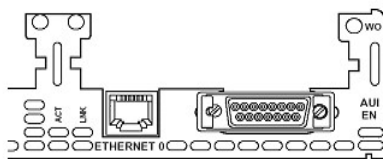
### (1) RJ-45 端口

利用 RJ-45 端口也可以建立广域网与局域网 VLAN（虚拟局域网）之间，以及与远程网络或 Internet 的连接。如果使用路由器为不同 VLAN 提供路由时，可以直接利用双绞线连接至不同的 VLAN 端口。但要注意这里的 RJ-45 端口所连接的网络一般就不太可有是 10Base-T 这种了，一般都是 100Mbps 快速以太网以上。如果必须通过光纤连接至远程网络，或连接的是其他类型的端口时，则需要借助于收发转发器才能实现彼此之间的连接。如图 5 所示为快速以太网（Fast Ethernet）端口。



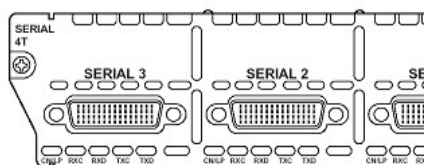
### (2) AUI 端口

AUI 端口我们在局域网中也讲过，它是用于与粗同轴电缆连接的网络接口，其实 AUI 端口也被常用于与广域网的连接，但是这种接口类型在广域网应用得比较少。在 Cisco 2600 系列路由器上，提供了 AUI 与 RJ-45 两个广域网连接端口（如图 6 所示），用户可以根据自己的需要选择适当的类型。



### (3) 高速同步串口

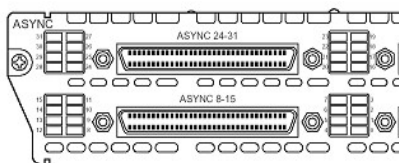
在路由器的广域网连接中，应用最多的端口还要算“高速同步串口”（SERIAL）了，如图 7 所示。



这种端口主要是用于连接目前应用非常广泛的 DDN、帧中继（Frame Relay）、X.25、PSTN（模拟电话线路）等网络连接模式。在企业网之间有时也通过 DDN 或 X.25 等广域网连接技术进行专线连接。这种同步端口一般要求速率非常高，因为一般来说通过这种端口所连接的网路的两端都要求实时同步。

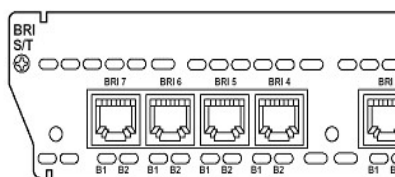
### (4) 异步串口

异步串口（ASYNC）主要是应用于 Modem 或 Modem 池的连接，如图 8 所示。它主要用于实现远程计算机通过公用电话网拨入网络。这种异步端口相对于上面介绍的同步端口来说在速率上要求就松许多，因为它并不要求网路的两端保持实时同步，只要求能连续即可，主要是因为这种接口所连接的通信方式速率较低。



### (5) ISDN BRI 端口

因 ISDN 这种互联网接入方式连接速度上有它独特的一面，所以在当时 ISDN 刚兴起时在互联网的连接方式上还得到了充分的应用。ISDN BRI 端口用于 ISDN 线路通过路由器实现与 Internet 或其他远程网络的连接，可实现 128Kbps 的通信速率。ISDN 有两种速率连接端口，一种是 ISDN BRI（基本速率接口）；另一种是 ISDN PRI（基群速率接口）。ISDN BRI 端口是采用 RJ-45 标准，与 ISDN NT1 的连接使用 RJ-45-to-RJ-45 直通线。如图 9 所示的 BRI 为 ISDN BRI 端口。

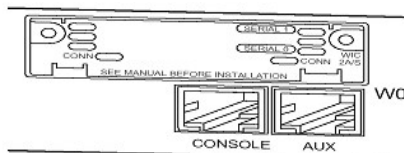


## 路由器配置接口

路由器的配置端口有两个，分别是“Console”和“AUX”，“Console”通常是用来进行路由器的基本配置时通过专用连线与计算机连用的，而“AUX”是用于路由器的远程配置连接用的。

### (1) Console 端口

Console 端口使用配置专用连线直接连接至计算机的串口，利用终端仿真程序（如 Windows 下的“超级终端”）进行路由器本地配置。路由器的 Console 端口多为 RJ-45 端口。如下图 10 所示就包含了一个 Console 配置端口。



### (2) AUX 端口

AUX 端口为异步端口，主要用于远程配置，也可用于拨号连接，还可通过收发器与 MODEM 进行连接。AUX 端口与 Console 端口通常同时提供，因为它们各自的用途不一样。接口图示仍参见上述图 10。

## 路由器的硬件连接

从上面的介绍或知，路由器的接口类型非常多，它们各自用于不同的网络连接，如果不能明白各自端口的作用，就很可能进行错误的连接，导致网络连接不正确，网络不通。下面我们通过对路由器的几种网络连接形式来进一步理解各种端口的连接应用环境。

路由器的硬件连接因端口类型，也主要分与局域网设备之间的连接、与广域网设备之间的连接以及与配置设备之间的连接三类。

### 路由器与局域网接入设备之间的连接

局域网设备主要是指集线器与交换机，交换机通常使用的端口只有 RJ-45 和 SC，而集线器使用的端口则通常为 AUI、BNC 和 RJ-45。下面，我们简单介绍一下路由器和集线设备各

种端口之间如何进行连接。

### (1) RJ-45-to-RJ-45

这种连接方式就是路由器所连接的两端都是 RJ-45 接口的，如果路由器和集线设备均提供 RJ-45 端口，那么，可以使用双绞线将集线设备和路由器的两个端口连接在一起。需要注意的是，与集线设备之间的连接不同，路由器和集线设备之间的连接不使用交叉线，而是使用直通线，也就是说，跳线两端的线序完全相同，但也不是说只要线序相同就行，但最好不要采用一一对应法。再一个要注意的是集线器设备之间的级联通常是通过级联端口进行的，而路由器与集线器或交换机之间的互联是通过普通端口进行的。

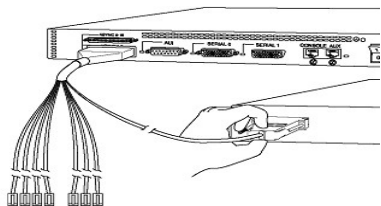
另外，路由器和集线设备端口通信速率应当尽量匹配，否则，宁可使集线设备的端口速率高于路由器的速率，并且最好将路由器直接连接至交换机。

## 路由器与 Internet 接入设备的连接

路由器的主要应用互联网的连接，路由器与互联网接入设备的连接情况主要有以下几种：

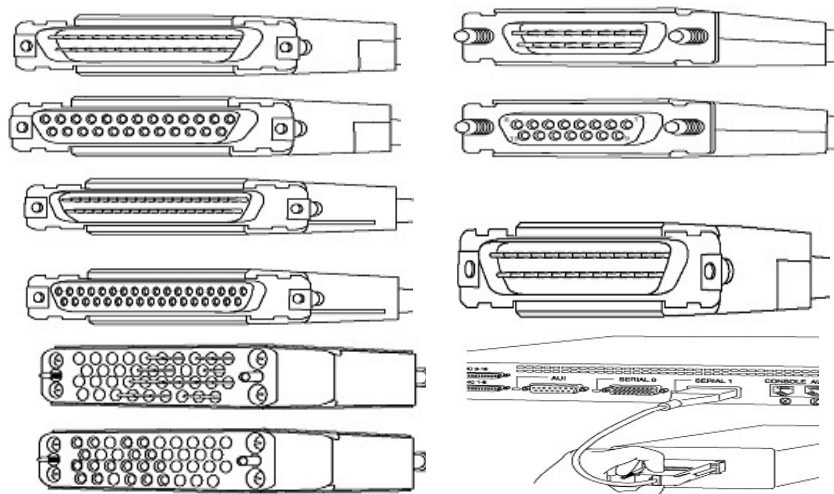
### (1) 通过异步串行口连接

异步串口主要是用来与 Modem 设连接，用于实现远程计算机通过公用电话网拨入局域网。除此之外，也可用于连接其他终端。当路由器通过电缆与 Modem 连接时，必须使用 AYSNC-to-DB25 或 AYSNC-to-DB9 适配器来连接。路由器与 Modem 或终端的连接如图 12 所示。



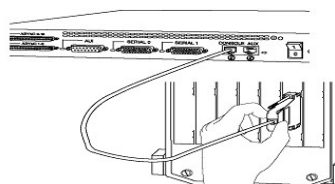
### (2) 同步串行口

在路由器中所能支持的同步串行端口类型比较多，如 Cisco 系统就可以支持 5 种不同类型的同步串行端口，分别是：EIA/TIA-232 接口、EIA/TIA-449 接口、V.35 接口、X.21 串行电缆总成和 EIA-530 接口，所对应的适配器图示分别如图 13、图 14、图 15、图 16、图 17 所示。但是在这里要注意的一点就是，因为一般来说适配器连线的两端是采用不同的外形（一般称带插针之类推适配器头一端称之为“公头”，而带有孔的适配器一端通常称之为“母头”，注意“EIA-530”接口两端都是一样的接口类型），这主要是考虑到连接的紧密。图中的“公头”为 DTE（数据终端设备，Data Terminal Equipment）连接适配器，下方“母头”为 DCE（数据通信设备，Data Communications Equipment）连接适配器。如图 18 所示为同步串行口与 Internet 接入设备连接的示意图，在连接时只需要对应看一下连接用线与设备端接口类型就可以知道正确选择了。



### 配置设备链接

当使用计算机配置路由器时，必须使用翻转线将路由器的 Console 口与计算机的串口/并口连接在一起，这种连接线一般来说需要特制，根据计算机端所使用的是串口还是并口，选择制作 RJ-45-to-DB-9 或 RJ-45-to-DB-25 转换用适配器，如图 21 所示。



## 实验二 修改路由器的名称及密码操作

### 实验要求：

1. 设置路由器名为：wanhe
2. 设置特权模式下的 password 为 ccna, secret 为 ccnp, vty 线路密码为 ccie
3. 要求所有的明文密码都加密

### 实验过程：

#### 1. 设置基本明文的密码

Router> **enable**

注：从用户模式进入特权模式

Router # **configure terminal**

注：从特权模式进入全局配置模式

Router (config)#**hostname wanhe**

注：路由器名称改为 wanhe

wanhe(config)# **enable password ccna**

注：特权密码加密且 password 为明文加密



```
wanhe(config)#exit
```

注：由全局配置视图进入特权视图

```
wanhe#exit
```

注：从特权模式退到上级级别的用户模式

## 2. 验证结果

```
wanhe con0 is now available
Press RETURN to get started.
wanhe>
wanhe>enable
Password:
wanhe#
```

注：IOS 的编写是基于 UNIX 系统的，所以输入密码的时候不会显示位数，当输入正确的

密码后，可以进入特权模式

## 3. 查看路由器内存中的所有配置

```
wanhe#sh run
Building configuration...
Current configuration : 857 bytes
省略
!
```

enable password **ccna**-----特权密码加密且 password 为明文加密，即通过 show run 命令可看到明文密码

```
memory-size iomem 5
no aaa new-model
ip subnet-zero
省略
```

## 4. 设置加密密码

```
wanhe#conf t
wanhe(config)# enable secret ccnp
```

注：设置加密的特权密码，secret 为加密密码，即通过 show run 命令看不到明文密码 ccnp

## 5. 查看加密密码

```
wanhe#sh run
Building configuration...
省略
```

```
no service password-encryption
hostname wanhe
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$rRML$6L/XFgA9gJylQtQ0byLp..
```

注：表示路由器的加密程度和级别；Secret 加密与 password 加密相比，安全级别较高

```
enable password ccna
!
memory-size iomem 5
no aaa new-model
ip subnet-zero
```

## 6.再次验证结果

```
wanhe#exit
wanhe con0 is now availe
Press RETURN to get started.
wanhe>enable
Password:
```

注：现在特权密码有 2 个，一个是明文密码，一个是加密密码，路由器优先选择加密程度

高的 secret 密码

## 7.设置 VTY 密码

```
wanhe(config)# line vty 0 4-----加密 0 到 4 这五条线路
wanhe(config-line)#password ccie
wanhe(config-line)#login
wanhe(config-line)#end-----从接口模式下直接退到特权模式
wanhe#sh run
Building configuration...
省略

line con 0
  transport preferred all
  transport output all
line aux 0
  transport preferred all
  transport output all
line vty 0 4
  password ccie
```



注：设置成功，但是现在的密码是明文的，并且没有 secret 在这个线路中使用

```
login
End
```

### 8. 给所有的明文密码加密

```
wanhe(config)#service password-encryption
```

### 9. 查看加密效果

```
wanhe#sh run
Building configuration...
省略
```

```
line con 0
transport preferred all
transport output all
line aux 0
transport preferred all
transport output all
line vty 0 4
password 7 1306141B0E
```

注：将 password 密码进行了简单加密

```
login
transport preferred all
transport input all
transport output all
```

## 实验三 基本的操作命令

### 实验要求：

1. 熟练掌握各种基本操作

### 实验过程：

#### 1、配置路由器本地时间：

```
wanhe#clock set 12:12:12 30 Nov 2008
wanhe#show clock
12:12:30.204 UTC Sun Nov 30 2008
```

#### 2、启用光标跟随：

```
wanhe (config) #line console 0
wanhe (config-line) #logging synchronous
wanhe (config) #end
wanhe#
```

解释：所谓光标跟随，是指当我们在输入命令的时候，不会被一些日志信息或 debug 命令产生的调试命令所中断，默认是关闭的。

### 3、为路由器设置标语信息和描述信息：

```
wanhe (config) #banner motd #  
Enter TEXT message .End with the character”#”  
Hello! Welcome to wanhe  
#  
wanhe (config) #end  
wanhe#exit
```

wanhe con0 is now available

Press RETURN to get started

Hello! Welcome to wanhe

wanhe>

注意：应谨慎选择登录旗标中使用的词语，诸如“欢迎”等词语可能隐含着访问不受限制的意思，从而让黑客找到发起攻击的借口。

### 4、为接口配置描述信息

```
wanhe (config-if) #description ISP  
wanhe (config-if) #end  
wanhe#show run int e0  
Interface Ethernet0  
    description ISP  
    no ip address  
    shutdown  
end
```

### 5、快速恢复接口到出厂设置：

```
wanhe#show run int s0  
Interface Serial0  
    ip address 10.0.0.1 255.255.255.252  
  
wanhe (config) #default interface serial 0  
  
Building configuration...  
Interface Serial0 set to default configuration  
  
wanhe (config) #end  
wanhe#show run int s0  
!
```

Interface Serial0  
no ip address

查看路由器版本	Router#show version
查看路由器 <b>flash</b>	Router#show flash:
查看历史命令记录	Router#show history
查看接口物理相关信息	Router#show interfaces s0/0
查看接口协议相关信息	Router#show ip int s0/0
查看接口简要信息	Router#show ip int brief
查看路由器启动配置文件	Router#show running-config
查看路由器运行配置文件	Router#show running-config
查看当前登录的所有用户	Router#show users
查看路由器 <b>ARP</b> 表	Router#show arp

## 实验四 CDP 命令操作

### 实验要求:

1. 设置 1 号线路的路由器名称为 wanhe1、 2 号的线路 wanhe2、 3 号线路为 wanhe3
2. 配置路由器，以保证 3 层连通性
3. 使用 CDP 命令观察路由器的连接



### 实验过程:

#### 1. wanhe1 的配置

Router>enable

Password:

Router#conf t

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

Router(config)#hostname wanhe1

## 2. 查看路由器端口状态

wanhe1#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
Serial0/0	unassigned	YES	unset	<b>administratively down</b>	<b>down</b>
Serial0/1	unassigned	YES	unset	administratively down	down

注：status 和 protocol 分别表示物理端口状态 数据链路层状态。默认情况下，路由

器的端口为“管理员性关闭”

## 3. 设置路由器的一层连通性

wanhe1#configure terminal

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

wanhe1(config)#interface serial 0

wanhe1(config-if)#**no shutdown**-----把端口手工打开

wanhe1(config-if)#end

等待约 30 秒时间后

wanhe1#show ip interface brief

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

注：现在的端口状态变成了物理 down

虽然我们再端口下使用了 no shutdown 的命令，但是端口状态依然还是 down，现实中原因可能有端口问题，线缆问题等物理问题。我们这个时候为了排错需要，把线路切换到 2 号线路。

Router>enable

Router#show ip interface brief

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

注：连接到 wanhe1 的 serial 0 端口为“管理员性关闭”，也就是说这个端口还没有电压载

入，所以对端的状态就是物理性 down

Router#conf t

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

Router(config)#hostname wanhe2

wanhe2(config)#interface serial 0

wanhe2(config-if)#no shutdown

wanhe2(config-if)#end

## 等待约 30 秒的网络延迟后

wanhe2#show ip interface brief

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

注：双方都手工打开了端口，这个时候状态为 **up**。这个时候数据链路层为 **down**，原因是

线路中的 DCE 端口没有配置时钟频率

## 4. 设置路由器的二层连通性

## 查看某一端口的线缆连接情况

wanhe2#show controllers serial 0

省略

line state: up

cable type : V.11 (X.21) **DCE** cable, received clockrate 2015232

base0 registers=0x3C000000, base1 registers=0x3C002000

mxt\_ds=0x62AD6220, rx ring entries=78, tx ring entries=128

rxring=0x5D18580, rxr shadow=0x62619EF8, rx\_head=40

txring=0x5D18820, txr shadow=0x6261A2CC, tx\_head=43, tx\_tail=43, tx\_count=0

注：这个端口连接的是 DCE 端，需要在这个端口上配置时钟频率

省略

wanhe2(config)#int s0

wanhe2(config-if)#**clock rate 64000**

注：设置端口的时钟频率为 64000 bit/s（1.544M 的速率与 64000bit/s 的时钟频率最匹配）

wanhe2#show ip interface brief

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

注：数据链路层 **UP**

## 5. 设置路由器的三层连通性

wanhe1(config)#int s0

wanhe1(config-if)#ip address **192.168.12.1 255.255.255.0**

注：配置 IP 地址和子网掩码

wanhe1#show ip interface brief

Interface	IP-Address	OK?	Method	Status	Protocol
Serial0	<b>192.168.12.1</b>	YES	manual	up	up
Serial1	unassigned	YES	unset	administratively down	down

注：查看到在 serial0 端口配置的 IP 地址为 192.168.12.1

```
wanhe2(config)#int serial 0
wanhe2(config-if)#ip address 192.168.12.2 255.255.255.0
wanhe2#ping 192.168.12.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
!!!!----- PING 成功，不成功则为“.....”
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/70/108 ms
```

## 6. CDP 的相关命令

```
wanhe2#show cdp
Global CDP information:
    Sending CDP packets every 60 seconds
    Sending a holdtime value of 180 seconds
    Sending CDPv2 advertisements is enabled
```

注：路由器每 60 秒发送一次 CDP 信息 抑制时间为 180 秒，也就是说经过 180 的时间依

然没有接受到对方发来的 CDP 信息，则把对方从 CDP 邻居表中删除

```
wanhe2#show cdp neighbors
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
wanhe1	Ser 0	122	R	2500	Ser 0

wanhe1-----对端路由器的名称  
 Ser 0-----连接到对端的本地接口名称  
 R-----对端为路由器  
 2500-----对端路由器的型号为 2500 系列  
 Ser 0-----连接到本地路由器的对端接口名称

```
wanhe2#show cdp entry *
```

\*代表所有与本地路由器连接的情况，也可以查看和对端某一具体路由器的连接情况

```
Device ID: wanhe1
```

```
Entry address(es):
```

IP address: **192.168.12.1**-----对端的 IP 地址为 192.168.12.1

```
Platform: Cisco 2500, Capabilities: Router Switch IGMP
```

```
Interface: Serial0, Port ID (outgoing port): Serial0
```

```
Holdtime : 179 sec
```

注意：要使用 CDP 信息，至少要保证二层连通性，也就是说，你的数据链路层至少要 UP

## 实验五 备份路由器的 IOS

### 实验要求:

1. 配置路由器的以太网口的 IP 地址为 192.168.2.1
2. 在 PC 上开启 TFTP SERVER,修改 IOS 的路径保存到 d 盘根目录 2500 文件夹
3. 把路由器的 IOS 导入到 PC 上



### 实验过程:

#### 1. wanhe 上的配置

```
wanhe#conf t
wanhe(config)#interface ethernet 0
wanhe(config-if)#ip address 192.168.2.1 255.255.255.0
wanhe(config-if)#no shutdown
```

#### 2. 配置 PC 端的 IP 地址为 192.168.2.2，子网掩码为 255.255.255.0 (过程省略)

#### 3. 验证 3 层连通性

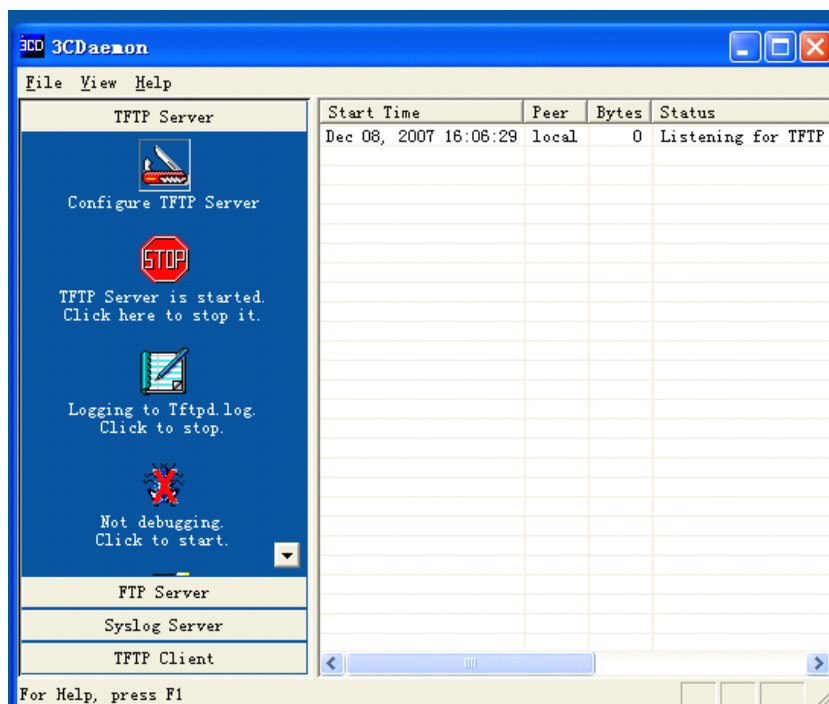
```
wanhe#ping 192.168.2.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 40/70/108 ms
```

#### 3. 在 PC 端开启 TFTP 软件，在桌面上点击以下图标

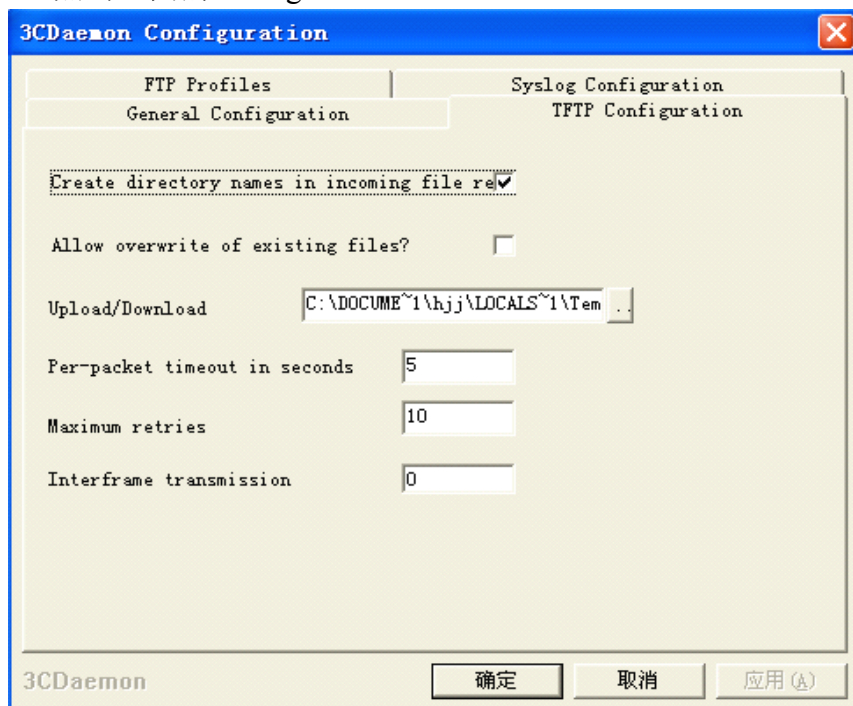


打开后出现下列界面





点击上面的 configure TFTP server



修改 upload/download 的目录为 D:/IOS

#### 4. 查看路由器的 IOS 名称

wanhe#show version

省略

Terminal-Server uptime is 3 hours, 13 minutes

System returned to ROM by power-on

System image file is "**flash:c2500-i-l.121-10.bin**"-----路由器的 IOS 名称

省略

#### 5. 开始备份 IOS

wanher#**copy flash: tftp:**

注：flash: -----代表源地址，来自 flash。 Tftp: -----代表目的地址，去往 TFTP server

中间有空格，请注意。

Source filename []? **c2500-i-l.121-10.bin**

注：源文件名称，用刚才 show Version 中查看到的路由器 IOS

Address or name of remote host []? **192.168.2.2**

注：对端 TFTP server 的 IP 地址。

Destination filename [c2500-i-l.121-10.bin]?

注：默认的路由器名称，建议不要去更改，此处回车即可。

!!  
!!!!!!!!!!!!!!!!!!!!!!!!!!!! 省略

约过 10 分钟后，IOS 备份成功，在 PC 的 D 盘的 IOS 文件夹是否有备份好的 IOS 文件

## 实验六 路由器密码的恢复

### 实验要求：

1. 在路由器上将密码设置为 wanhe 后，保存配置；
2. 重新启动路由器将密码破解，并保证里面的配置其他配置依然正常

### 实验过程：

#### 1. 以下是设置密码过程

在 wanhe 上的配置：

router(config)#hostname wanhe

wanhe(config)#enable secret wanhe

jaince(config)#exit

wanhe#**copy running-config startup-config** -----保存配置文件

wanhe#**reload**-----重新启动

#### 2. 以下是密码恢复过程

2500 series

重新启动后马上不断地按键盘的 CTRL 和 BREAK 键。如果是使用了笔记本，则需要按 FN 键，直到进入以下的 ROMMON 模式

> **o/r 0x2142**

>----- ROMMON 模式的提示符号

o/r-----2500 系列修改配置寄存器值的指令

0x2142----把配置寄存器值修改为 2142

> **i**-----保存配置并重新启动

启动完成后，现在的路由器中没有任何配置文件被载入

```
router#copy startup-config running-config
```

注：把配置文件加载路由器的 RAM 中

```
wanhe#configure terminal
```

```
wanhe(config)#no enable secret
```

注：删除密码，对于 cisco 路由器，删除密码并不需要原来的密码

```
wanhe(config)#exit
```

### 3. 检查配置寄存器值

```
wanhe#show version
```

X.25 software, Version 3.0.0.

1 Ethernet/IEEE 802.3 interface(s)

2 Serial network interface(s)

32K bytes of non-volatile configuration memory.

16384K bytes of processor board System flash (Read ONLY)

Configuration register is **0x2142**-----现在的配置寄存器值为 2142

检查密码是否存在？

注意：如果是在实际工程环境中，完成上述工作后，还需要做两件事情。

1. 修改完密码后，需要保存配置，否则之前的密码还是放在配置文件中；
2. 把配置寄存器的值修改为 2102，因为如果是 2142 的话，下次路由器的配置文件不会被加载。

在正常的 IOS 中修改配置寄存器的值如下：

```
wanhe(config)#configure-register 0x2102
```

## 2600 series and the later

重新启动后马上不断地按键盘的 CTRL 和 BREAK 键。如果是使用了笔记本，则需要按 FN 键，直到进入以下的 ROMMON 模式

```
> confreg 0x2142
```

>----- ROMMON 模式的提示符号

**confreg** -----2600 以及以上系列修改配置寄存器值的指令

0x2142----把配置寄存器值修改为 2142

```
> reset-----保存配置并重新启动
```

启动完成后，现在的路由器中没有任何配置文件被载入

```
router#copy startup-config running-config
```

注：把配置文件加载路由器的 RAM 中

```
wanhe#configure terminal
```

```
wanhe(config)#no enable secret
```

注：删除密码，对于 cisco 路由器，删除密码并不需要原来的密码

```
wanhe(config)#exit
```

#### 4. 检查配置寄存器值

```
wanhe#show version
```

X.25 software, Version 3.0.0.  
1 Ethernet/IEEE 802.3 interface(s)  
2 Serial network interface(s)  
32K bytes of non-volatile configuration memory.  
16384K bytes of processor board System flash (Read ONLY)  
Configuration register is **0x2142**-----现在的配置寄存器值为 2142  
检查密码是否存在？

注意：如果是在实际工程环境中，完成上述工作后，还需要做两件事情。

3. 修改完密码后，需要保存配置，否则之前的密码还是放在配置文件中；
4. 把配置寄存器的值修改为 2102，因为如果是 2142 的话，下次路由器的配置文件不会被加载。

在正常的 IOS 中修改配置寄存器的值如下：

```
wanhe(config)#configure-register 0x2102
```

注：不同型号的路由器，密码恢复时的参数可能不同，详情请具体参考路由器配置手册。

## 实验七 配置 TELNET 远程登陆

### 实验要求：

1. 在 wanhe2 上设置特权密码和 VTY 密码；
2. 从 wanhe1 上 TELNET 到 wanhe2；
3. 在 wanhe2 上查看已经 TELNET 到本设备上的其它用户。



### 实验过程：

#### 1. 设置 wanhe2 的特权密码和线路密码

```
wanhe2(config)#enable secret wanhe-----设置加密的特权密码  
wanhe2(config)#line vty 0 ?  
<1-15> Last Line number
```

注：本地路由器最多可以创建 0 到 15 条线路，也就是说可以允许同时 16 个人登陆到这台路

由器上

```

wanhe2(config)#line vty 0 4
wanhe2(config-line)#password cisco -----设置线路密码
wanhe2(config-line)#exit
wanhe2(config)#interface serial 0
wanhe2(config-if)#ip address 192.168.12.2 255.255.255.0
wanhe2(config-if)#clock rate 64000
wanhe2(config-if)#no shutdown

```

## 2. 设置 wanhe1 的端口

```

wanhe1(config)#interface serial 0
wanhe1(config-if)#ip address 192.168.12.1 255.255.255.0
wanhe1(config-if)#no shutdown

```

## 3. 验证

```

wanhe1#telnet 192.168.12.2
Trying 192.168.12.2 ... Open
User Access Verification
Password: -----输入线路密码 cisco
wanhe2>enable
Password: -----输入特权密码 wanhe
wanhe2#
查看当前正在使用的用户
wanhe2#show users

```

Line	User	Host(s)	Idle	Location
0 con 0		idle	00:00:22	
* 3 vty 0		idle	00:00:00	192.168.12.1

\*-----表示当前正在使用的线路  
 0-----表示第一个 VTY 线路  
 192.168.12.1-----表示登陆端的 IP 地址

Interface	User	Mode	Idle	Peer Addr
-----------	------	------	------	-----------

## 4. login 的使用

```

wanhe2#show running-config
Building configuration...
line con 0
  exec-timeout 0 0
  transport preferred all
  transport output all
line aux 0
  transport preferred all
  transport output all
line vty 0 4
  password cisco
  login

```

注：我们在对 VTY 线路操作的时候，并没有输入 login，但是却发现系统自动加上了这条

命令

!

End

wanhe2#conf t

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

wanhe2(config)#line vty 0 4

wanhe2(config-line)#no login-----删去 login，以研究它的作用

wanhe2(config-line)#end

wanhe2#quit-----退出 telnet，也可以用 exit

[Connection to 192.168.12.2 closed by foreign host]

wanhe1#telnet 192.168.12.2

Trying 192.168.12.2 ... Open

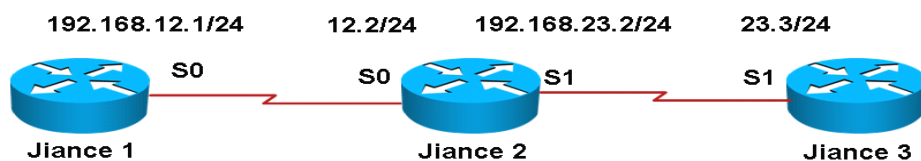
**wanhe2>**-----我们发现，wanhe1 竟然直接登陆到 wanhe2 上，而没有进行线路密码的检测！

由此得出结论，login 的作用是对线路密码进行检测，如果去掉，则导致直接登陆到路由器上，而丧失了安全性。对于高版本的 IOS,login 这条命令可以自动加上，但是对于低版本的 IOS 却没有这个功能，所以，为了网络的安全性，我们最好手工输入 login，以确保网络的安全性。

## 实验八 静态路由

### 实验要求：

1. 设置路由器的基本参数，如图所示；配置完成后应保证在 wanhe2 上可以 ping 通 wanhe1 和 wanhe3。
2. 在路由器 wanhe1 配置静态路由（使用本地出站接口）；
3. 在 wanhe3 上配置默认路由（使用邻居路由器的下一跳地址）；



### 实验过程：

1. 检验路由器之间的三层连通性

wanhe2#ping 192.168.12.1

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 32/55/80 ms

```
wanhe2#ping 192.168.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3,timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/61/120 ms
wanhe2#
```

## 2. 在 wanhe1 上配置静态路由

```
wanhe1#ping 192.168.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3,timeout is 2 seconds:
.....
```

注：Ping 不成功，原因是路由表中没有到达 192.168.23.0 网段的信息

```
wanhe1#show ip route
Gateway of last resort is not set
C    192.168.12.0/24 is directly connected, Serial0
```

注：C 代表的是直连网段，而我们需要配置和学习的是非直连网段

```
wanhe1(config)#ip route 192.168.23.0 255.255.255.0 192.168.12.2
```

注：上面所加的 ip 地址分别为目的网段、子网掩码、下一跳地址

```
wanhe1#sh ip route
Gateway of last resort is not set
C    192.168.12.0/24 is directly connected, Serial0
S    192.168.23.0/24 [1/0] via 192.168.1.2
```

注：S 代表静态路由、管理距离值为 1、度量值为 0。

```
wanhe1#ping 192.168.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3,timeout is 2 seconds:
.....
```

问题分析：虽然现在路由表中有路由条目了，但是 ping 却没有成功，那是因为我们的 PING 命令是基于 ICMP 协议，ping 到对方的时候，我们使用的是 ECHO REQUEST 报文，对方收到后，我们使用的是 ECHO RELAY 来回应，那么现在只有 wanhe1 上路由条目，而 wanhe3 上却没有回来的路径，所以 ping 会失败。

```
wanhe3#sh ip route
Gateway of last resort is not set
C    192.168.23.0/24 is directly connected, Serial1
```

```
wanhe1#ping 192.168.23.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.2,timeout is 2 seconds:
!!!!
```



```
wanhe2#sh ip route
```

```
Gateway of last resort is not set
```

```
C    192.168.12.0/24 is directly connected, Serial0
```

```
C    192.168.23.0/24 is directly connected, Serial1
```

问题分析：这个时候我们发现在 wanhe2 上的路由条目中有 2 个直连网段的条目，当我们从 wanhe1 ping 到 wanhe2 上的时候，wanhe2 上可以利用直连路由到达 192.168.12.1 这个接口 IP 地址。从而说明，路由器需要学习的是非直连网段的路由条目。为了让 wanhe1 能 ping 到 wanhe3，我们还需要以下的设置。

```
wanhe3(config)#ip route 192.168.12.0 255.255.255.0 serial 1
```

注：除了可以用下一跳地址，还可以用出站接口的名称。

```
wanhe3(config)#end
```

```
wanhe3#sh ip route
```

```
Gateway of last resort is not set
```

```
S    192.168.12.0/24 is directly connected, Serial1
```

注：使用本地出站接口的静态路由，它就好像是直连在路由器的接口上的，所以它的管理距

离值为 0

```
C    192.168.23.0/24 is directly connected, Serial1
```

```
wanhe3#ping 192.168.12.1
```

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

```
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
```

```
!!!!
```

还有一种默认路由的配置方式

```
wanhe3(config)#ip route 0.0.0.0 0.0.0.0 192.168.23.2
```

注：8 个 0 代表了所有的网段所有主机

```
wanhe3(config)#end
```

```
wanhe3#sh ip route
```

```
Gateway of last resort is 192.168.23.2 to network 0.0.0.0
```

注：最后的求助网关为 192.168.23.2 这个地址，也就是说，路由表接受了一个路由条目后，

如果在路由表中查找不到，就通过这个地址转发出去。

```
S    192.168.12.0/24 is directly connected, Serial1
```

```
C    192.168.23.0/24 is directly connected, Serial1
```

```
S*   0.0.0.0/0 [1/0] via 192.168.23.2
```

注：默认路由都是以\*号出现的

小结：在配置静态路由时，两种不同的配置方法所得到的路由度量值是不一样的，  
配置为出站接口，度量值为 0

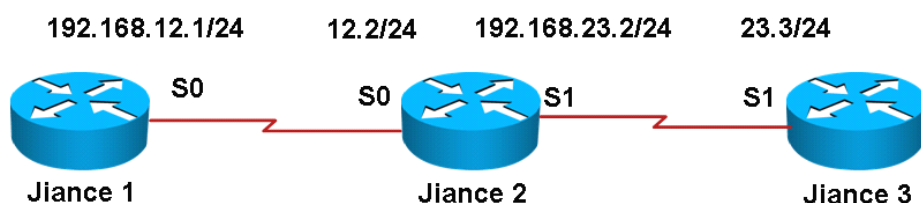
配置为下一跳地址，度量值为 1

## 实验九 RIP 动态路由的配置

阶段一：配置基本的 RIP

实验要求：

- 1.在三台路由器上启用 RIP 协议；
- 2.查看路由表并验证各网段之间的连通性.



实验过程：

### 1.wanhe1 上的配置

```
wanhe1(config)#router rip
wanhe1(config-router)#network 192.168.12.0
```

注：启用 RIP 协议、把路由器所有 IP 地址匹配 192.168.12.1—192.168.12.255 的接口参与到

RIP 协议的路由选择进程中。

### 2.wanhe2 上的配置

```
wanhe2(config)#router rip
wanhe2(config-router)#network 192.168.12.0
wanhe2(config-router)#network 192.168.23.0
```

### 3. wanhe3 上的配置

```
wanhe3(config)#router rip
wanhe3(config-router)#network 192.168.23.0
```

### 4.查看路由协议

```
wanhe1#show ip protocol
Routing Protocol is "rip"
  Sending updates every 30 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 1, receive any version
```

```

Interface          Send  Recv  Triggered RIP  Key-chain
Serial 0           1      1 2
Automatic network summarization is in effect
Maximum path: 4
Routing for Networks:
  192.168.12.0
Routing Information Sources:
  Gateway          Distance      Last Update
Distance: (default is 120)
    
```

注：RIP 每 30S 发送路由表的拷贝给它的邻居。180S 时间后还接收不到邻居宣告的路由条

目，则认为邻居 possibly down 状态。180S 时间之内如果出现物理问题，则宣告这

个端口上所有接收到的路由条目为 possibly down 状态。标注为 possibly down 状态

的路由条目，再经过 60S 的时间后，将完全从路由表中删除。240=180+60

### 5.查看路由表

wanhe1#sh ip route

Gateway of last resort is not set

C 192.168.12.0/24 is directly connected, Serial0

**R 192.168.23.0/24 [120/1] via 192.168.1.2, 00:00:15, Serial0**

注：RIP 协议的标号为 R IP 的管理距离值为 120。RIP 计算度量值的方法是跳数，从 R1

到 192.168.23.0 网段需要经过几台路由器，也就是几跳，所以跳数为 1。

### 6.测试连通性

wanhe1#ping 192.168.23.3

Type escape sequence to abort.

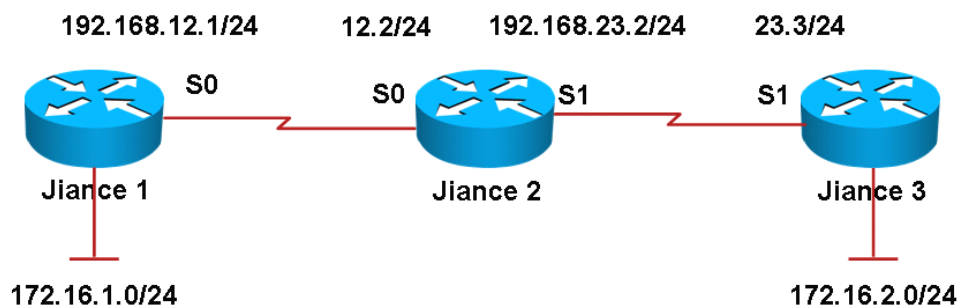
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:

!!!!

## 阶段二：解决 RIP 中子网不连续的现象

### 实验要求：

- 1.wanhe1 上添加一个逻辑环回口，IP 地址为 172.16.1.1/24
- 2.wanhe3 上添加一个逻辑环回口，IP 地址为 172.16.2.1/24
- 3.在 RIP 进程中，把两个环回口添加到 RIP 路由进程中



### 实验过程:

#### 1.wanhe1 添加网段并宣告

```
wanhe1(config)#interface loopback 0
wanhe1(config-if)#ip add 172.16.1.1 255.255.255.0
wanhe1(config-if)#router rip
wanhe1(config-router)#network 172.16.1.0
```

注：创建一个逻辑的环回口，该接口在路由器中永远处于 UP 的状态，非常稳定。接口编号

任意

#### 2.wanhe3 加入路由选择进程

```
wanhe3(config)#interface loopback 0
wanhe3(config-if)#ip add 172.16.2.1 255.255.255.0
wanhe3(config-if)#router rip
wanhe3(config-router)#network 172.16.2.0
```

#### 3.问题现象

在 wanhe2 上发现如下的现象：

```
wanhe2#ping 172.16.2.1
```

!U!!

注：U 也是代表不可达，现在的丢包率为 50%！做实验时，可能出现全部 ping 通的现象，

那是由于 CEF 起作用，你可以换成 ping 172.16.1.1 具体内容将会在 CCNP 课程中介绍。

#### 4.查看 wanhe2 的路由表

```
wanhe2#sh ip route
```

```
R    172.16.0.0/16 [120/1] via 192.168.23.3, 00:00:15, Serial1
      [120/1] via 192.168.12.1, 00:00:09, Serial0
```

```
C    192.168.12.0/24 is directly connected, Serial0
```

```
C    192.168.23.0/24 is directly connected, Serial1
```

注：172.16.0.0 一个网段对应着两个出站接口，所以数据出去的时候会轮流选择两个出站接

口，最后只有 50%的成功率。

问题分析：RIP 现在使用的是默认版本 v1，它是一个有类路由选择协议，所以两端虽然宣告的是 24 位子网掩码，但是实际宣告的是 16 位的子网掩码。

### 5. 开启 debug 信息动态观察路由更新情况

```
wanhe2#debug ip rip
```

```
RIP protocol debugging is on
```

```
wanhe2#
```

```
*Mar  1 00:42:23.095: RIP: sending v1 update to 255.255.255.255 via Serial0/0  
(192.168.1.2)
```

```
*Mar  1 00:42:23.099: RIP: build update entries
```

```
*Mar  1 00:42:23.099:   network 192.168.23.0 metric 1
```

```
*Mar  1 00:42:23.639: RIP: sending v1 update to 255.255.255.255 via Serial0/1  
(192.168.2.2)
```

```
*Mar  1 00:42:23.639: RIP: build update entries
```

```
*Mar  1 00:42:23.643:   network 192.168.12.0 metric 1
```

```
*Mar  1 00:42:24.559: RIP: received v1 update from 192.168.1.1 on Serial0/0
```

```
*Mar  1 00:42:24.563:           172.16.0.0 in 1 hops
```

注：来自 wanhe1 的环回口网段。

```
*Mar  1 00:42:40.247: RIP: received v1 update from 192.168.2.3 on Serial0/1
```

```
*Mar  1 00:42:40.251:           172.16.0.0 in 1 hops
```

注：来自 wanhe3 的环回口网段。

```
wanhe2#u all
```

注：最快速关闭 DEBUG 的方法，它的全写是 undebug all

问题分析：RIP 2 版本是无类路由选择协议，把默认版本升级到 RIP 2，是否可以解决这样的想象？

### 6. 修改 rip 版本

```
wanhe1(config)#router rip
```

```
wanhe1(config-router)#version 2
```

```
wanhe1#clear ip route *
```

注：修改位 2 版本。RIP 的抑制时间为 180S，为了快速收敛，我们强制清除路由表。

```
wanhe2(config)#router rip
```

```
wanhe2(config-router)#version 2
```

```
wanhe2#clear ip route *
```

```
wanhe3(config)#router rip
wanhe3(config-router)#version 2
wanhe3#clear ip route *
```

```
wanhe2#ping 172.16.2.1
!U!!
```

注：我们依然还是 PING 不通，所以我们到 2 端去 DEBUG 查看

```
wanhe1#
*Mar  1 00:40:02.111: RIP: sending v2 update to 224.0.0.9 via Loopback0 (172.16.1.1)
*Mar  1 00:40:02.115: RIP: build update entries
*Mar  1 00:40:02.119:    192.168.12.0/24 via 0.0.0.0, metric 1, tag 0
*Mar  1 00:40:02.123:    192.168.23.0/24 via 0.0.0.0, metric 2, tag 0
*Mar  1 00:40:02.127: RIP: sending v2 update to 224.0.0.9 via Serial0 (192.168.1.1)
*Mar  1 00:40:02.127: RIP: build update entries
*Mar  1 00:40:02.131:    172.16.0.0/16 via 0.0.0.0, metric 1, tag 0
*Mar  1 00:40:02.143: RIP: ignored v2 packet from 172.16.1.1 (sourced from one of our
addresses)
```

注：现在的确是使用无类路由选择协议来发送路由更新，但是它携带的确是经过汇总后的

#### 16 位子网掩码

```
wanhe1#u all
All possible debugging has been turned off
```

```
jinace3#
00:57:11: RIP: received v2 update from 192.168.2.2 on Serial1
00:57:11:    192.168.12.0/24 via 0.0.0.0 in 1 hops
00:57:31: RIP: sending v2 update to 224.0.0.9 via Loopback0 (172.16.2.1)
00:57:31: RIP: build update entries
00:57:31:    192.168.12.0/24 via 0.0.0.0, metric 2, tag 0
00:57:31:    192.168.23.0/24 via 0.0.0.0, metric 1, tag 0
00:57:31: RIP: sending v2 update to 224.0.0.9 via Serial1 (192.168.2.3)
00:57:31: RIP: build update entries
00:57:31:    172.16.0.0/16 via 0.0.0.0, metric 1, tag 0
00:57:31: RIP: ignored v2 packet from 172.16.2.1 (sourced from one of our addresses)
```

注： wanhe3 上也是由于汇总的问题才宣告了 16 位的路由条目。

问题分析：两边同时宣告 172.16.0.0/16 的网段，所以 wanhe2 依然还是一个路由条目对应 2 个出站接口的现象。要解决这样的问题，还需要我们关闭 RIP 中自动汇总的特性。

## 7.关闭自动汇总

```
wanhe1(config-router)#no auto-summary
```

```
wanhe1#clear ip route *
```

```
wanhe2(config-router)#no auto-summary
```

```
wanhe2#clear ip route *
```

```
wanhe3(config-router)#no auto-summary
```

```
wanhe3#clear ip route *
```

```
wanhe2#sh ip route
```

```
172.16.0.0/16 is variably subnetted, 3 subnets, 2 masks
```

```
R      172.16.1.0/24 [120/1] via 192.168.12.1, 00:00:07, Serial0
```

```
R      172.16.2.0/24 [120/1] via 192.168.23.3, 00:00:00, Serial1
```

```
C      192.168.12.0/24 is directly connected, Serial0
```

```
C      192.168.23.0/24 is directly connected, Serial1
```

```
wanhe2#ping 172.16.1.1
```

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

```
Sending 5, 100-byte ICMP Echos to 172.16.1.1,timeout is 2 seconds:
```

```
!!!!
```

```
wanhe2#ping 172.16.2.1
```

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

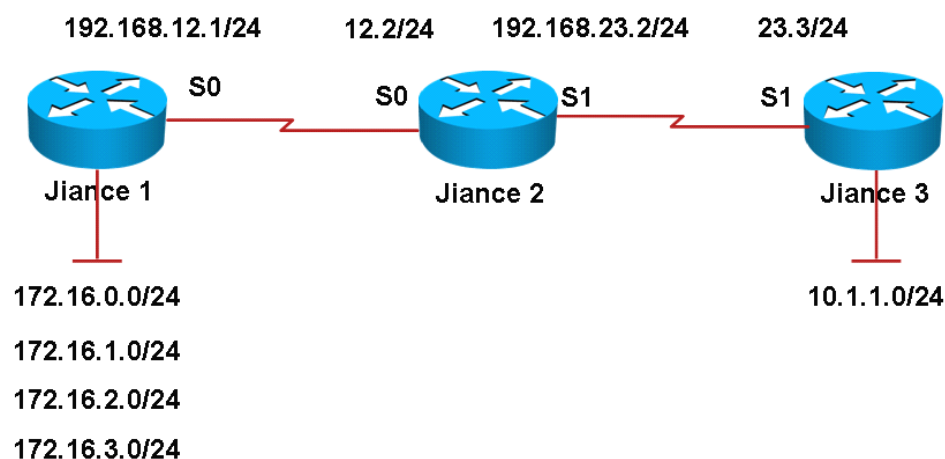
```
!!!!
```

## 实验十 EIGRP 路由协议

阶段一：基本的 EIGRP 配置

实验要求：

1. 完成路由器基本参数配置；
2. 在 wanhe1 上创建 4 个环回口；在 wanhe3 上创建 1 个环回口；IP 地址如图所示；
3. 所有路由器参与 EIGRP，AS 号为 100。





**实验过程:****1.启动 EIGRP 进程**

```
wanhe1(config)#router eigrp 100
wanhe1(config-router)#network 192.168.12.0
wanhe1(config-router)#network 172.16.0.0
wanhe1(config-router)#network 172.16.1.0
wanhe1(config-router)#network 172.16.2.0
wanhe1(config-router)#network 172.16.3.0
```

```
wanhe2(config)#router eigrp 100
wanhe2(config-router)#network 192.168.12.0
wanhe2(config-router)#network 192.168.23.0
```

```
wanhe3(config)#router eigrp 100
wanhe3(config-router)#network 192.168.23.0
wanhe3(config-router)#network 10.0.0.0
```

**2.查看 wanhe2 的路由表**

```
wanhe2#show ip route
Gateway of last resort is not set
D    172.16.0.0/16 [90/2297856] via 192.168.12.1, 00:01:11, Serial0
D    10.0.0.0/8 [90/2297856] via 192.168.23.3, 00:01:11, Serial1
C    192.168.12.0/24 is directly connected, Serial0
C    192.168.23.0/24 is directly connected, Serial1
```

注：EIGRP 的代号为 D，来源与它的核心算法：DUAL。EIGRP 的自治系统内部管理距离值

为 90。度量值的计算参数为：带宽，延迟，可靠，负载，MTU。但是默认情况下只有带

宽和延迟。

**3. 查看 wanhe2 上运行的动态路由选择协议**

```
wanhe2#show ip protocols
Routing Protocol is "eigrp 100"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
```

注：计算 EIGRP 度量值默认的参数是：K1(带宽)，K3（延迟）

```
EIGRP maximum hopcount 100
EIGRP maximum metric variance 1
Redistributing: eigrp 100
Automatic network summarization is in effect
```

Automatic address summarization:

192.168.23.0/24 for Serial0

192.168.12.0/24 for Serial1

Maximum path: 4

Routing for Networks:

**192.168.12.0**

**192.168.23.0**

Routing Information Sources:

Gateway	Distance	Last Update
192.168.12.1	90	00:03:53
192.168.23.3	90	00:03:53

Distance: internal 90 external 170

注：在 wanhe2 上宣告的两个网段

#### 4. 查看 EIGRP 的邻居表

wanhe2#show ip eigrp neighbors

IP-EIGRP neighbors for process 100

H	Address	Interface	Hold Uptime	SRTT (sec)	RTO	Q	Seq Type	Cnt Num
1	192.168.23.3	Se1	12 00:23:42	24	200	0	4	
0	192.168.12.1	Se0	11 00:26:44	688	4128	0	3	

注：H 代表建立 EIGRP 邻居关系的先后顺序。EIGRP 的 HELLO 时间为 5 秒，HOLD 时间

为 15 秒。

#### 5. 观察 EIGRP 的报文更新地址

wanhe3#debug ip packet

IP packet debugging is on

wanhe3#

01:18:51: IP: s=192.168.23.3 (local), d=**224.0.0.10** (Serial1), len 60, sending broad/multicast

01:18:52: IP: s=10.0.0.1 (local), d=224.0.0.10 (Loopback0), len 60, sending broad/multicast

01:18:52: IP: s=10.0.0.1 (Loopback0), d=224.0.0.10, len 60, rcvd 2

01:18:53: IP: s=192.168.23.2 (Serial1), d=224.0.0.10, len 60, rcvd 2

注：EIGRP 使用组播地址 224.0.0.10 来参与 EIGRP 进程

wanhe3#u all

注：速关闭 debug 信息

**阶段二：EIGRP 路由协议****实验要求：**

1. 设置 EIGRP 路由协议
2. 关闭自动汇总（解决路由非连续子网的路由可达性问题）
3. 使用手工汇总以达到精确汇总的目的

**实验过程：****1. 查看当前的路由表**

```
wanhe2#show ip route
```

```
Gateway of last resort is not set
```

```
D    172.16.0.0/16 [90/2297856] via 192.168.12.1, 00:14:43, Serial0
```

```
D    10.0.0.0/8 [90/2297856] via 192.168.23.3, 00:14:43, Serial1
```

```
C    192.168.12.0/24 is directly connected, Serial0
```

```
C    192.168.23.0/24 is directly connected, Serial1
```

注：从 wanhe1 上宣告的 4 个网段，出来的时候被自动汇总成它的主类网络

**2. 在 wanhe1 上关闭自动汇总**

```
wanhe1(config)#router eigrp 100
```

```
wanhe1(config-router)#no auto-summary
```

注：关闭自动汇总，这样可以了解到明细条目

**3. 观察关闭自动汇总后 wanhe2 路由表**

```
wanhe2#sh 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 - ODRP - periodic downloaded static route

```
Gateway of last resort is not set
```

```
172.16.0.0/24 is subnetted, 4 subnets
```

```
D    172.16.0.0 [90/2297856] via 192.168.12.1, 00:01:00, Serial0
```

```
D    172.16.1.0 [90/2297856] via 192.168.12.1, 00:01:00, Serial0
```

```
D    172.16.2.0 [90/2297856] via 192.168.12.1, 00:01:00, Serial0
```

```
D    172.16.3.0 [90/2297856] via 192.168.12.1, 00:01:00, Serial0
```

```
D    10.0.0.0/8 [90/2297856] via 192.168.23.3, 00:17:55, Serial1
```

```
C    192.168.12.0/24 is directly connected, Serial0
```

```
C    192.168.23.0/24 is directly connected, Serial1
```

注：明细网段的条目就在对端邻居显示出来了

问题分析：但是大部分的时候，我们需要精细的汇总条目，这个时候，我们使用手工汇总的方式。

#### 4.在 wanhe1 上设置手工汇总

```
wanhe1(config)#interface serial 0
```

```
wanhe1(config-if)#ip summary-address eigrp 100 172.16.0.0 255.255.252.0
```

注：EIGRP 的手工汇总是在接口模式下使用。需要精确最总条目的网络号和子网掩码。

#### 5. 观察开启自动汇总后的 wanhe2 上路由表

```
wanhe2#show ip route
```

```
172.16.0.0/22 is subnetted, 1 subnets
```

```
D    172.16.0.0 [90/2297856] via 192.168.12.1, 00:01:23, Serial0
```

```
D    10.0.0.0/8 [90/2297856] via 192.168.23.3, 00:21:15, Serial1
```

```
C    192.168.12.0/24 is directly connected, Serial0
```

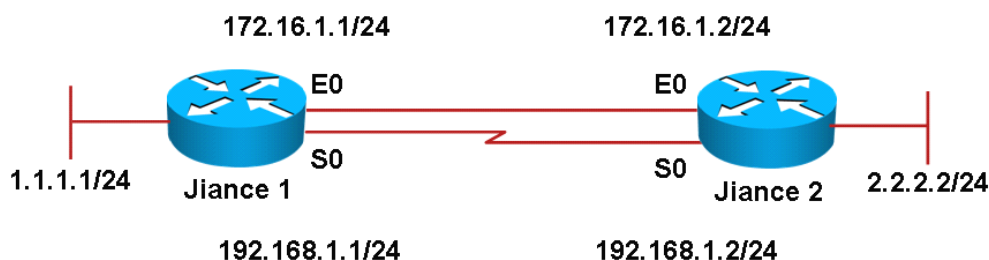
```
C    192.168.23.0/24 is directly connected, Serial1
```

注：现在的路由表中，4 个明细路由条目被手工汇总为子网掩码为 22 的路由条目

## 实验十一 EIGRP 非等价带宽的负载均衡

### 实验要求：

- 1.路由器名为 wanhe1 和 wanhe2；
- 2.两路由器用双绞线（10M）和串行线连接(1.544M)相连接；
- 3.使用命令实现非等价负载均衡。
- 4.验证非等价负载均衡



### 实验过程：

注：请先关闭路由的快速转发功能，以免影响实验结果

#### 1. 路由器 wanhe1 的配置

```
wanhe1(config)#no ip cef
```

```
wanhe1(config)#interface ethernet 0
```

```
wanhe1(config-if)#ip address 172.16.1.1 255.255.255.0
```

```
wanhe1(config-if)#exit
```

```
wanhe1(config)#interface serial 0
wanhe1(config-if)#ip address 192.168.1.1 255.255.255.0
wanhe1(config-if)#no shutdown
wanhe1(config-if)#exit
wanhe1(config)#interface loopback 0
wanhe1(config-if)#ip address 1.1.1.1 255.255.255.0
```

```
wanhe1(config)#router eigrp 100
wanhe1(config-router)#network 192.168.1.0
wanhe1(config-router)#network 172.16.1.0
wanhe1(config-router)#network 1.1.1.0
wanhe1(config-router)#no auto-summary
```

```
wanhe2(config)#no ip cef
wanhe2(config-if)#interface serial 0
wanhe2(config-if)#ip address 192.168.1.2 255.255.255.0
wanhe2(config-if)#clock rate 64000
wanhe2(config-if)#no shutdown
wanhe2(config-if)#exit
wanhe2(config)#interface ethernet 0
wanhe2(config-if)#ip address 172.16.1.2 255.255.255.0
wanhe2(config-if)#exit
wanhe2(config)#interface loopback 0
wanhe2(config-if)#ip address 2.2.2.2 255.255.255.0
```

```
wanhe2(config)#router eigrp 100
wanhe2(config-router)#network 192.168.1.0
wanhe2(config-router)#network 172.16.1.0
wanhe2(config-router)#network 2.2.2.0
wanhe2(config-router)#no auto-summary
```

## 2. 查看路由表

```
wanhe1#sh ip route
```

04:24:23: %SYS-5-CONFIG\_I: Configured from console by consoleute

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 - ODRP - periodic downloaded static route

Gateway of last resort is not set

1.0.0.0/24 is subnetted, 1 subnets

C 1.1.1.0 is directly connected, Loopback0

2.0.0.0/24 is subnetted, 1 subnets

D **2.2.2.0 [90/409600] via 172.16.1.2, 00:00:16, Ethernet0**

```
172.16.0.0/24 is subnetted, 1 subnets
C      172.16.1.0 is directly connected, Ethernet0
C      192.168.1.0/24 is directly connected, Serial0
```

注：到达目的网段只能经过以太网线路，路由器认为走这条路径最优

### 3. 查看拓扑表

```
wanhe1#show ip eigrp topology
IP-EIGRP Topology Table for AS(100)/ID(1.1.1.1)
Codes: P - Passive, A - Active, U - Update, Q - Query, R - Reply,
       r - reply Status, s - sia Status
P 1.1.1.0/24, 1 successors, FD is 128256
     via Connected, Loopback0
P 2.2.2.0/24, 1 successors, FD is 409600
     via 172.16.1.2 (409600/128256), Ethernet0
     via 192.168.1.2 (2297856/128256), Serial0
P 192.168.1.0/24, 1 successors, FD is 2169856
     via Connected, Serial0
P 172.16.1.0/24, 1 successors, FD is 281600
     via Connected, Ethernet0
```

注：查看 EIGRP 的拓扑表。**409600** 这个参数指的是可行距离，就是本地路由器到达目的网

段的度量值。**128256** 这个参数指的是通告距离，就是邻居路由器到达目的网段的度量值。

问题分析：从上可以看出，经过以太网线路的度量值约为 41 万，而经过 serial 线的度量值约为 229 万，根据度量值越小越优先的原则，路由器会选取 41 万度量值的这个条目加入到路由表中。在 EIGRP 的路由表中，为了可以进行非等价的负载均衡，我们必须要把 serial 线这个路径加入到路由表中。

在 EIGRP 协议中，有 variance（因子值）这个参数，它的含义是把最优路径乘以这个因子值，以让次优路径可以加入到路由选择表中。现在的最优路径为 41 万，次优路径为 229 万，那么  $41 \text{ 万} \times 5 = 205 \text{ 万}$ ，度量值依然还是小于 229 万的，那么 serial 线这条路径还是加入不到路由表中，而  $41 \text{ 万} \times 6 = 246 \text{ 万}$ ，正好大于 229 万，那么我们就选取 6 作为我们的因子值。

### 4. 设置 eigrp 的因子值

```
wanhe1(config)#router eigrp 100
wanhe1(config-router)#variance 6
wanhe1(config-router)#end
wanhe1#clear ip route *
wanhe1#show ip route
Gateway of last resort is not set
1.0.0.0/24 is subnetted, 1 subnets
```

- C            1.1.1.0 is directly connected, Loopback0  
              2.0.0.0/24 is subnetted, 1 subnets
- D            2.2.2.0 [90/409600] via 172.16.1.2, 00:00:01, Ethernet0**  
              **[90/2297856] via 192.168.1.2, 00:00:01, Serial0**

注： 现在一个路由标目对应着 2 个出站接口，就可以进行非等价负载均衡了

- 172.16.0.0/24 is subnetted, 1 subnets
- C            172.16.1.0 is directly connected, Ethernet0
- C            192.168.1.0/24 is directly connected, Serial0
- wanhe2(config)#router eigrp 100  
wanhe2(config-router)#variance 6  
wanhe2(config-router)#end  
wanhe2#clear ip route \*  
wanhe2#sh ip route  
Gateway of last resort is not set  
              1.0.0.0/24 is subnetted, 1 subnets
- D            1.1.1.0 [90/409600] via 172.16.1.1, 00:00:01, Ethernet0  
              [90/2297856] via 192.168.1.1, 00:00:01, Serial0  
              2.0.0.0/24 is subnetted, 1 subnets
- C            2.2.2.0 is directly connected, Loopback0  
              172.16.0.0/24 is subnetted, 1 subnets
- C            172.16.1.0 is directly connected, Ethernet0
- C            192.168.1.0/24 is directly connected, Serial0

## 5. 验证效果

wanhe1#**traceroute** -----进行路由跟踪  
Protocol [ip]:  
**Target IP address: 2.2.2.2**-----目的地址  
**Source address: 1.1.1.1**-----源地址  
Numeric display [n]:  
Timeout in seconds [3]:  
**Probe count [3]: 20**-----一共发送 20 个包  
Minimum Time to Live [1]:  
Maximum Time to Live [30]:  
Port Number [33434]:  
Loose, Strict, Record, Timestamp, Verbose[none]:  
Type escape sequence to abort.  
Tracing the route to 2.2.2.2  
  1 172.16.1.2 4 msec \*    4 msec \*    4 msec \*  
    192.168.1.2 8 msec  
  172.16.1.2 4 msec \*    4 msec \*    12 msec \*  
    192.168.1.2 8 msec

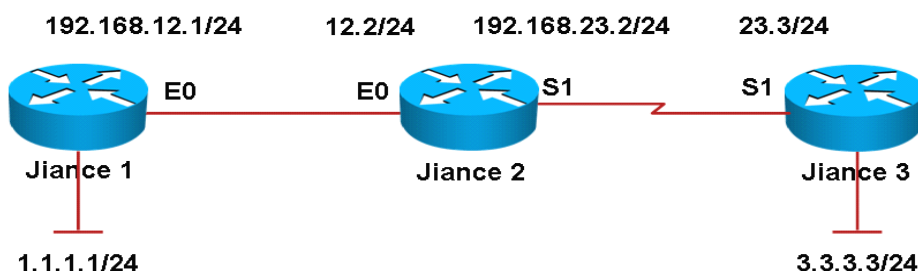
注： 以太网线走了 6 个包，其中“\*”也算一个包。Serial 线走了 1 个包。



## 实验十二 配置单区域的 OSPF 协议

### 实验要求:

1. 路由器名为 wanhe1, wanhe2 和 wanhe3;
2. 配置 IP 地址如图所示, wanhe1 的环回口 IP 地址为 1.1.1.1, 255.255.255.0, wanhe3 为 3.3.3.3, 子网掩码为 255.255.255.0。
3. 三台路由器属于同一个 OSPF 区域 0;
4. 将三台路由器相连, 并在路由器上启用 OSPF 路由协议;
5. 用扩展 PING 命令, 测试用 1.1.1.1 到 3.3.3.3 的连通性。



### 实验过程:

#### 1. 设置路由选择协议

```
wanhe1(config)#router ospf 1
wanhe1(config-router)#network 192.168.12.0 0.0.0.255 area 0
wanhe1(config-router)#network 1.1.1.1 0.0.0.0 area 0
```

注: 利用反掩码技术来匹配某一接口, 如果匹配到了, 那么就把这个接口的网段参与到 OSPF 的进程中

```
wanhe2(config)#router ospf 1
wanhe2(config-router)#network 192.168.0.0 0.0.255.255 area 0
```

注: 虽然只写入了一个条目, 但是它匹配到了 192.168.12.2 和 192.168.23.2 这两个接口, 所以它们都会参与到 OSPF 进程中

```
wanhe3(config)#router ospf 1
wanhe3(config-router)#network 3.3.3.0 0.0.0.255 area 0
wanhe3(config-router)#network 192.168.23.0 0.0.0.255 area 0
```

#### 2. 查看路由表

```
wanhe1#show ip route
1.0.0.0/24 is subnetted, 1 subnets
C      1.1.1.0 is directly connected, Loopback0
```

3.0.0.0/32 is subnetted, 1 subnets

```
O      3.3.3.3 [110/75] via 192.168.12.2, 00:00:59, Ethernet0
C      192.168.12.0/24 is directly connected, Ethernet0
O      192.168.23.0/24 [110/74] via 192.168.12.2, 00:00:59, Ethernet0
```

注：OSPF 的代号为 O。管理距离值为 110。度量值的计算方法是  $\text{cost} = 10 \times \frac{\text{带宽}}{\text{接口速度}}$  的 8 次方除以带宽。

### 3.观察 OSPF 邻居关系的建立情况

```
wanhe1#debug ip ospf adj
wanhe1(config)#int e0
wanhe1(config-if)#shutdown
wanhe1(config-if)#
```

注：开启 OSPF 建立邻居关系的 debug 信息。关闭端口！以验证邻居建立情况

```
02:00:59: %LINK-5-CHANGED: Interface Ethernet0, changed state to administratively down
02:01:00: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0, changed state to down
```

```
wanhe1(config-if)#no shutdown
wanhe1(config-if)#
02:01:05: OSPF: Interface Ethernet0 going Up
02:01:05: OSPF: Build router LSA for area 0, router ID 1.1.1.1, seq 0x80000005
02:01:07: %LINK-3-UPDOWN: Interface Ethernet0, changed state to up
02:01:08: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0, changed state to up
02:01:12: OSPF: 2 Way Communication to 192.168.23.2 on Ethernet0, state 2WAY
02:01:12: OSPF: Backup seen Event before WAIT timer on Ethernet0
02:01:12: OSPF: DR/BDR election on Ethernet0
02:01:12: OSPF: Elect BDR 1.1.1.1
02:01:12: OSPF: Elect DR 192.168.23.2
02:01:12: OSPF: Elect BDR 1.1.1.1
02:01:12: OSPF: Elect DR 192.168.23.2
02:01:12:      DR: 192.168.23.2 (Id)    BDR: 1.1.1.1 (Id)
02:01:12: OSPF: Send DBD to 192.168.23.2 on Ethernet0 seq 0x1513 opt 0x42 flag 0x7 len 32
02:01:15: OSPF: Rcv DBD from 192.168.23.2 on Ethernet0 seq 0x169C opt 0x42 flag 0x7 len 32 mtu 1500 state EXSTART
02:01:15: OSPF: NBR Negotiation Done. We are the SLAVE
02:01:15: OSPF: Send DBD to 192.168.23.2 on Ethernet0 seq 0x169C opt 0x42 flag 0x2 len 112
02:01:15: OSPF: Rcv DBD from 192.168.23.2 on Ethernet0 seq 0x169D opt 0x42 flag 0x3 len 92 mtu 1500 state EXCHANGE
02:01:15: OSPF: Send DBD to 192.168.23.2 on Ethernet0 seq 0x169D opt 0x42 flag 0x0 len
```

32

```

02:01:15: OSPF: Database request to 192.168.23.2
02:01:15: OSPF: sent LS REQ packet to 192.168.12.2, length 12
02:01:15: OSPF: Rcv DBD from 192.168.23.2 on Ethernet0 seq 0x169E opt 0x42 flag 0x1
len 32  mtu 1500 state EXCHANGE
02:01:15: OSPF: Exchange Done with 192.168.23.2 on Ethernet0
02:01:15: OSPF: Send DBD to 192.168.23.2 on Ethernet0 seq 0x169E opt 0x42 flag 0x0 len
32
02:01:15: OSPF: Synchronized with 192.168.23.2 on Ethernet0, state FULL
02:01:15: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.23.2 on Ethernet0 from LOADING
to FULL, Loading Done
02:01:15: OSPF: Build router LSA for area 0, router ID 1.1.1.1, seq 0x80000006
wanhe1(config-if)#end
wanhe1#u all

```

问题分析: OSPF 的整个邻居关系应该经历以下完整的过程

DOWN---INIT---2 WAY---EXSTART---EXCHANGE---LOADING---FULL

你能回答出每个过程的具体细节和交换的报文吗?

### 3.查看 ospf 邻居表

```

wanhe1#show ip ospf neighbor
Neighbor ID      Pri   State   Dead Time   Address      Interface
192.168.23.2     1     FULL/DR  00:00:39  192.168.12.2 Ethernet0

```

注: OSPF 的 HELLO 时间为 10 秒, HOLD 时间为 40 秒。可以显示对端邻居的接口 IP 地

址, 可以查看 OSPF 的邻居状态, 稳定时为 FULL。

```

wanhe2#show ip ospf neighbor
Neighbor ID      Pri   State   Dead Time   Address      Interface
1.1.1.1          1     FULL/BDR 00:00:31  192.168.12.1 Ethernet0
3.3.3.3          1     FULL/   -   00:00:38  192.168.23.3 Serial1

```

注: -代表了 wanhe2 和 wanhe3 上却没有 DR 和 BDR 的选举

问题分析: 为什么在 wanhe1 和 wanhe2 上有 DR 和 BDR 的选举过程, 而 wanhe2 和 wanhe3 上却没有? 我们要了解, 对于 DR 和 BDR 的选举, 是有条件的, 那就是必须是要在多路访问的链路上进行, wanhe1 和 wanhe2 是用以太网线链接, 以太网的链路是星型结构, 属于多路访问类型, 可以参与 DR 和 BDR 的选举, 但是 wanhe2 和 wanhe3 上使用的是 serial 线缆, 它是属于点到点的专线结构, 所以是没有必要参与 DR 和 BDR 的选举的。

### 4.查看参与 OSPF 进程的以太网口状态

```

wanhe1#show ip ospf interface ethernet 0
Ethernet0 is up, line protocol is up
Internet Address 192.168.12.1/24, Area 0
Process ID 1, Router ID 1.1.1.1, Network Type BROADCAST, Cost: 10-----

```

以太网类型为多路访问 (*broadcast*)

Transmit Delay is 1 sec, State DROTHER, Priority 0  
Designated Router (ID) 192.168.23.2, Interface address 192.168.12.2  
No backup designated router on this network  
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5  
Hello due in 00:00:07  
Index 1/1, flood queue length 0  
Next 0x0(0)/0x0(0)  
Last flood scan length is 1, maximum is 1  
Last flood scan time is 0 msec, maximum is 8 msec  
Neighbor Count is 1, Adjacent neighbor count is 1  
Adjacent with neighbor 192.168.23.2 (Designated Router)  
Suppress hello for 0 neighbor(s)

wanhe2#show ip ospf interface serial 1

Serial1 is up, line protocol is up

Internet Address 192.168.23.2/24, Area 0

Process ID 1, Router ID 192.168.23.2, Network Type **POINT\_TO\_POINT**, Cost:

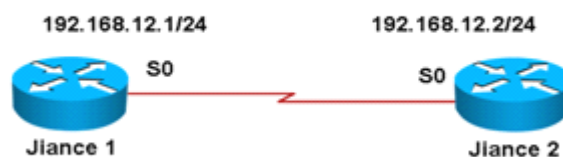
64-----Serial 线缆为点到点类型的

Transmit Delay is 1 sec, State POINT\_TO\_POINT,  
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5  
Hello due in 00:00:00  
Index 2/2, flood queue length 0  
Next 0x0(0)/0x0(0)  
Last flood scan length is 1, maximum is 2  
Last flood scan time is 0 msec, maximum is 0 msec  
Neighbor Count is 1, Adjacent neighbor count is 1  
Adjacent with neighbor 3.3.3.3  
Suppress hello for 0 neighbor(s)

## 实验十三 使用扩展的 ACL 封杀 PING 命令

实验要求:

1. 路由器名为:wanhe1、wanhe2，并配置相关的 IP 地址，以保证 3 层连通性
2. 做扩展的访问控制列表，禁止 wanhe1 PING 到 wanhe2



## 实验过程:

### 1.配置路由器的基本参数

```
wanhe1(config)#interface serial 0
wanhe1(config-if)#ip address 192.168.12.1 255.255.255.0
wanhe1(config-if)#no shutdown
```

```
wanhe2(config)#interface serial 0
wanhe2(config-if)#clock rate 64000
wanhe2(config-if)#ip address 192.168.12.2 255.255.255.0
wanhe2(config-if)#no shutdown
```

### 2. 验证 3 层连通性

```
wanhe1#ping 192.168.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!
Success rate is 0 percent (0/5)
```

### 3.创建 ACL

```
wanhe1(config)#access-list 100 deny icmp 192.168.12.1 0.0.0.0 192.168.12.2 0.0.0.0
wanhe1(config)#access-list 100 permit ip any any
```

注：扩展 ACL 的编号范围为 100 到 199。我们需要拒绝的是 PING 命令，它是属于 ICMP

协议，所以我们需要拒绝 ICMP。用反掩码绝对匹配一个源地址。用反掩码绝对匹配

一个目的地址。ACL 有隐含拒绝的条目，它会拒绝所有的数据流量，为了防止数据流

被误拒绝，我们可以加一条放行所有数据流量的条目。

### 4.检查 ACL

```
wanhe1#show ip access-lists
Extended IP access list 100
    10 deny icmp host 192.168.12.1 host 192.168.12.2
    20 permit ip any any
```

注：ACL 以编号 10 开始，然后以 10 为递增。

### 5.应用 ACL 到接口

```
wanhe1(config)#int s0
wanhe1(config-if)#ip access-group 100 out
```

注：对出去的数据流量进行检测

### 6.验证效果

```
wanhe1#ping 192.168.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!-----实验失败！现在依然可以 PING 通！
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/37/52 ms
```

问题分析：对于 ACL 的放置位置，我们有以下的原则：扩展 ACL 放置在靠近源的位置，标准 ACL 放置在靠近目的位置。那按照上述的原则，我们创建一个扩展的 ACL，并放置在源端，并没有错误。

## 7. 排错

```
wanhe1#show ip access-lists
Extended IP access list 100
 10 deny icmp host 192.168.12.1 host 192.168.12.2
 20 permit ip any any (15 matches) -----Permit 语句匹配到 15 个数据包
```

问题分析：对于 ACL，有个非常重要的特性，他不能过滤本地数据流！也就是说，对于 wanhe1 上发送的数据，设置在 wanhe1 接口上的 ACL 并不能对它进行过滤。为了能对数据流进行过滤，我们需要把 ACL 设置在对端的 wanhe2 上

## 8. 在 wanhe2 上设置并应用 ACL

```
wanhe2(config)#access-l 100 deny icmp host 192.168.12.1 host 192.168.12.2
wanhe2(config)#access-l 100 permit icmp any any

wanhe2(config)#interface serial 0
wanhe2(config-if)#ip access-group 100 in
```

注:对于 wanhe2 来说，数据是进到 s0 接口中，所以我们需要对进来的数据进行检测，那么

我们就用 in

## 9. 检测效果

```
wanhe1#ping 192.168.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
U.U.U-----实验成功，ICMP 包被拒绝
Success rate is 0 percent (0/5)
```

```
wanhe2#show ip access-lists
Extended IP access list 100
 10 deny icmp host 192.168.12.1 host 192.168.12.2 (11 matches) ----
```

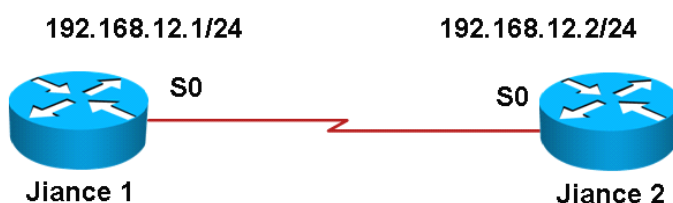
注：有相关的数据流被拒绝

```
 20 permit icmp any any
```

## 实验十四 使用 ACL 禁止 TELNET 应用

### 实验要求:

- 1.路由器名为:wanhe1、wanhe2;
- 2.wanhe1 的 S0 端口与 wanhe2 的 S0 端口相连, IP 地址如图所示,在 wanhe2 上设置特权密码为 wanhe, 线路密码为 cisco;
- 3.从 wanhe1 使用 PING 命令测试到 wanhe2 的连通性, 结果可达, 但却不可以 TELNET 到 wanhe2。



### 实验过程:

有两种方法可以实现这样的操作

#### 方法一:使用扩展 ACL

##### 1.检测基本配置

```
wanhe2(config)#enable secret wanhe
wanhe2(config)#line vty 0 4
wanhe2(config-line)#password cisco
wanhe2(config-line)#login
```

```
wanhe1#telnet 192.168.12.2
Trying 192.168.12.2 ... Open
User Access Verification
Password:
```

**wanhe2>**-----验证成功!

##### 2.创建 ACL

```
wanhe2(config)#access-list 101 deny tcp host 192.168.12.1 any eq 23
```

注: telnet 属于 TCP 协议。代表绝对匹配一个主机。telnet 协议的端口号是 23。

```
wanhe2(config)#access-list 101 permit ip any any
wanhe2(config-if)#ip access-group 101 in
```

### 3.检验效果

```
wanhe1#telnet 192.168.12.2
Trying 192.168.12.2 ...
% Destination unreachable; gateway or host down

wanhe1#ping 192.168.12.2
Type escape sequence to abort.
Sending5, 100-byteICMP Echos to 192.168.12.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/156/364 ms
```

注：我们只拒绝了 TELNET 协议，但是 ICMP 协议我们是放行的。

### 方法二:使用标准 ACL

#### 1. 删除先前的配置

```
wanhe2(config)#int s0
wanhe2(config-if)#no ip access-group 101 in

wanhe1#telnet 192.168.12.2
Trying 192.168.12.2 ... Open
User Access Verification
Password:
Password:
wanhe2> -----wanhe2 上的配置被删除了
```

#### 2.创建 ACL

```
wanhe2(config)#access-list 1 deny host 192.168.12.1
wanhe2(config)#access-list 1 permit any
```

注：标准的 ACL 编号范围是 1 到 99，和 1300 到 1999。标准的 ACL 只能检测源地址。

#### 3.应用 ACL

```
wanhe2(config)#line vty 0 4
wanhe2(config-line)# access-class 1 in
```

注：在 VTY 线路中应用 ACL。

#### 4.验证效果

```
wanhe1#telnet 192.168.12.2
Trying 192.168.12.2 ...
% Connection refused by remote host
```

问题分析：设置了 2 种 ACL，但是得到的效果却不一样。如果是使用了扩展的 ACL，那么它的提示是“% Destination unreachable; gateway or host down”，说明 23 号端口根本不可达。如果是使用了标准的 ACL 放置在 VTY 线路中，则提示“% Connection refused by remote host”，说明的确是到达了 23 号端口，只不过被拒绝了。在实际使用中最好使用扩展的 ACL，减少 23 号端口的负担！



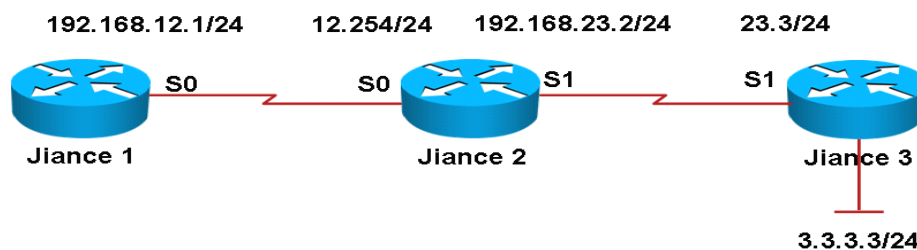
## 实验十五 NAT 网络地址转换

### 实验要求:

1. 在 wanhe1、2、3 上完成基本的配置，根据拓扑；
2. 在 wanhe2 上完成 NAT 的基本配置；
3. 理解静态 NAT，动态 NAT 和 PAT 的原理。

192.168.12.3/24 (辅助地址)

192.168.12.2/24 (辅助地址)



### 实验过程:

#### 1、配置每个设备的名称和接口的 ip 地址

```
wanhe1(config)#interface serial 0
```

```
wanhe1(config-if)#ip address 192.168.12.1 255.255.255.0
```

```
wanhe1(config-if)#ip address 192.168.12.2 255.255.255.0 secondary
```

注: 配置辅助 ip 地址。

```
wanhe1(config-if)#ip address 192.168.12.3 255.255.255.0 secondary
```

```
wanhe1(config-if)#no sh
```

```
wanhe2(config)#interface serial 0
```

```
wanhe2(config-if)#ip address 192.168.12.254 255.255.255.0
```

```
wanhe2(config-if)#no shutdown
```

```
wanhe2(config-if)#clock rate 64000
```

```
wanhe2(config)#interface serial 1
```

```
wanhe2(config-if)#ip add 192.168.23.2 255.255.255.0
```

```
wanhe2(config-if)#no shutdown
```

```
wanhe2(config-if)#clock rate 64000
```

```
wanhe2(config-if)#exit
```

```
wanhe3(config)#interface serial 1
```

```
wanhe3(config-if)#ip add 192.168.23.3 255.255.255.0
```

```
wanhe3(config-if)#no shutdown
```

```
wanhe3(config-if)#exit
```

#### 2、在 wanhe2 上完成静态 NAT 的配置。

```
wanhe2(config)#ip nat inside source static 192.168.12.1 192.168.23.4
```

注：Inside 关键字指定内部源本地 ip 地址转换成内部全局 ip 地址。当数据由内向外是转换

是源地址。回应时转换的是目标地址。静态 NAT 转换一对一。内部局部地址：在内部

网络使用的地址。内部全局地址：用来代替一个或多个本地地址的，对外的，向 NIC

注册过的地址。

```
wanhe2(config)#interface serial 0
wanhe2(config-if)#ip nat inside-----指定了 s0 接口在内部。
wanhe2(config-if)#int s 1
wanhe2(config-if)#ip nat out-----指定 s1 接口在外部。
wanhe2(config-if)#end
wanhe2#debug ip nat
IP NAT debugging is on
wanhe1#ping 192.168.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
....
Success rate is 0 percent (0/5)
```

注：用本地地址 192.168.12.1 Ping 192.168.23.3,结果没有 ping 通，为什么？

wanhe2#show ip nat translations

注：查看 wanhe2 上是否有地址转换的 NAT 表。

```
wanhe1(config)#ip route 192.168.23.0 255.255.255.0 serial 0
wanhe1(config)#end
```

注：为 jiace1 上加上去往 wanhe3 的路由。

```
wanhe1#ping 192.168.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
!!!!
```

注：可以 ping 通说明加上了路由可以让数据发出去也能回来。

Success rate is 100 percent (5/5), round-trip min/avg/max = 68/74/88 ms

```
wanhe1#ping-----使用扩展 ping。
Protocol [ip]:
Target IP address: 192.168.23.3
Repeat count [5]: 50-----发送 50 数据包。
```

Datagram size [100]:

Timeout in seconds [2]:

**Extended commands [n]:** -----这里不使用扩展的命令，直接回车。表示使用主 ip 地址 192.168.12.1 来 ping 192.168.23.3。

Sweep range of sizes [n]:

Type escape sequence to abort.

Sending 50, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:

!!!!!!!!!!!!!!!!!!!!

Terminal-Server#2-----快速切换到 wanhe2 上。来查看具体的转换过程。

[Resuming connection 2 to wanhe2 ... ]

00:18:28: NAT\*: s=192.168.12.1->192.168.23.4, d=192.168.23.3 [38]

00:18:28: NAT\*: s=192.168.23.3, d=192.168.23.4->192.168.12.1 [38] 省略...

注：第一个条目是将源地址进行转化。那第二个条目是将目的地址进行转化。

wanhe2#show ip nat translations

Pro Inside global    Inside local    Outside local    Outside global

--- 192.168.23.4      192.168.12.1            ---            ---

注：建立了 NAT 表，当有流量符合这个匹配规则时就会两个地址进行转换。

## 2、在 wanhe2 上完成动态 NAT 的配置。

wanhe2(config)#no ip nat inside source static 192.168.12.1 192.168.23.4-----将原来的静态 NAT 的条目删除。

若不能删除请执行命令：clear ip nat tran \*

wanhe2(config)#access-list 1 permit 192.168.12.0 0.0.0.255

注：通过使用用户访问控制列表来定义本地地址池。

wanhe2(config)#ip nat pool wanhe 192.168.34.1 192.168.34.2 p 24

注：通过使用用户访问控制列表来定义本地地址池。

wanhe2(config)#ip nat inside source list 1 pool wanhe

注：定义公有地址池，命名为 wanhe。地址的范围是 192.168.34.1 到 192.168.34.2，子网掩码

用前缀表示 24。也可以使用关键字 network+具体的网段。

## 3、用 192.168.12.1 ping 192.168.23.3

wanhe1#ping 192.168.23.3

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:

.....

Success rate is 0 percent (0/5)

注：当数据包到达 wanhe2 时会将 192.168.12.1 转化成 192.168.34.1，这时 wanhe3 会收到这

个数据包，因为目的地址没变。但 wanhe3 给 wanhe1 回应时，将以 192.168.34.1 为目的

地址，这是在 wanhe3 上没有相关的路由条目。

```
wanhe3(config)#ip route 192.168.34.0 255.255.255.0 s 1
```

注：在 wanhe3 上配置去往 wanhe1 上公有地址的路由。

```
wanhe1#ping
Protocol [ip]:
Target IP address: 192.168.23.3
Repeat count [5]: 50
Sending 50, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
!!!!!!!!!!!!!! -----Ping 通说明路由添加正确。
Terminal-Server#2
[Resuming connection 2 to wanhe2 ... ]
01:16:55: NAT*: s=192.168.12.1->192.168.34.1, d=192.168.23.3 [134]
01:16:55: NAT*: s=192.168.23.3, d=192.168.34.1->192.168.12.1 [134]
省略...
wanhe2#show ip nat tr
Pro Inside global    Inside local    Outside local    Outside global
--- 192.168.34.1     192.168.12.1      ---              ---
```

#### 4、用 **192.168.12.2** ping 192.168.23.3

```
wanhe1#ping
Protocol [ip]:
Target IP address: 192.168.23.3
Repeat count [5]: 20
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 192.168.12.2
Sending 20, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
Packet sent with a source address of 192.168.12.2
!!!!!!!!!!!!
```

注：要求使用扩展的命令。可以选择详细的参数。比如这里可以使用辅助的 ip 地址。使用

wanhe1 接口 s0 的辅助 ip 地址来作为源地址。

```
Terminal-Server#2
[Resuming connection 2 to wanhe2 ... ]
```

01:35:02: NAT\*: **s=192.168.12.2->192.168.34.2, d=192.168.23.3** [22]  
 01:35:02: NAT\*: s=192.168.23.3, d=192.168.34.2->192.168.12.2 [22]  
 省略...

注：源地址 192.168.12.2 转换成 192.168.34.2。很明显调用了第 2 个公有地址。

```
wanhe2#show ip nat tr
Pro Inside global   Inside local   Outside local   Outside global
--- 192.168.34.1    192.168.12.1   ---             ---
--- 192.168.34.2    192.168.12.3   ---             ---
```

### 5、用 **192.168.12.3** ping 192.168.23.3

```
wanhe1#ping
Protocol [ip]:
Target IP address: 192.168.23.3
Repeat count [5]: 20
Extended commands [n]: y
Source address or interface: 192.168.12.3-----用第 3 个私有地址来 ping 192.168.23.3。
Sending 20, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
Packet sent with a source address of 192.168.12.3
U.U.U.U.U.-----结果不能 ping 通到目的。
Terminal-Server>2
[Resuming connection 2 to wanhe2 ... ]
```

**00:22:02: NAT: translation failed (A), dropping packet s=192.168.12.3 d=192.168.23.3**

00:22:02: NAT: translation failed (A), dropping packet s=192.168.12.3 d=192.168.23.3  
 省略...

注：从调试的信息中可以查找出不能 ping 通的原因。是因为地址转换的失败而丢包。

wanhe2#**show ip nat tr**--通过显示 NAT 表也可以发现没有 192.168.12.3 的条目。

```
Pro Inside global   Inside local   Outside local   Outside global
--- 192.168.34.1    192.168.12.1   ---             ---
--- 192.168.34.2    192.168.12.2   ---             ---
```

解决的方法： 1、清除 NAT 表中的条目，将公有地址池中的公有地址释放出来。  
 2、将 NAT 超时时间改小，让被转换的目标地址能在短时间内得到释放。

请大家自己研究

```
wanhe2#clear ip nat tr *
wanhe2#show ip nat tr
```

```
Terminal-Server>1
[Resuming connection 1 to r4 ... ]
wanhe1#ping
Protocol [ip]:
```

```

Target IP address: 192.168.23.3
Extended commands [n]: y
Source address or interface: 192.168.12.3
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
Packet sent with a source address of 192.168.12.3
!!!!-----又可以 ping 通对端。
Terminal-Server>2
[Resuming connection 2 to wanhe2 ... ]

00:46:21: NAT: s=192.168.12.3->192.168.34.2, d=192.168.23.3 [55]
00:46:21: NAT*: s=192.168.23.3, d=192.168.34.2->192.168.12.3 [55]
省略...

```

注：调试所显示的转换过程。

```

wanhe2#sh ip nat tr
Pro Inside global    Inside local    Outside local    Outside global
--- 192.168.34.2      192.168.12.3      ---              ---

```

注：NAT 表中有了转换的条目。

## 6、配置 PAT

```

wanhe2(config)#no ip nat pool wanhe 192.168.34.1 192.168.34.2 prefix-length 24
wanhe2(config)#ip nat pool wanhe 192.168.34.1 192.168.34.1 prefix-length 24
wanhe2(config)#no ip nat inside source list 1 pool wanhe
wanhe2(config)#ip nat inside source list 1 pool wanhe overload

```

## 7、在 wanhe1 用 192.168.12.1 上 ping 192.168.23.3

```

wanhe1#ping 192.168.23.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 68/71/80 ms
Terminal-Server>2
[Resuming connection 2 to wanhe2 ... ]
01:01:55: NAT*: s=192.168.12.1->192.168.34.1, d=192.168.23.3 [74]
01:01:55: NAT*: s=192.168.23.3, d=192.168.34.1->192.168.12.1 [74]
省略...
wanhe2#sh ip nat tr
Pro    Inside global    Inside local    Outside local    Outside global
Icmp 192.168.34.1:6 192.168.12.1:6 192.168.23.3:6    192.168.23.3:6

```

注：由于发送的 ping 包，所以显示转换的是 icmp 协议。随机产生端口号 6。

```

wanhe2#
01:02:55: NAT: expiring 192.168.34.1 (192.168.12.1) icmp 6 (6)

```

注：约 1 分钟的时间释放地址转换的空间。

wanhe2# sh ip nat translations----查找 NAT，表中没有任何的转换条目。  
wanhe2#

## 8、在 wanhe1 用 192.168.12.2 上 ping 192.168.23.3

```
wanhe1#ping
Protocol [ip]:
Target IP address: 192.168.23.3
Extended commands [n]: y
Source address or interface: 192.168.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.23.3, timeout is 2 seconds:
Packet sent with a source address of 192.168.12.2
!!!!
Terminal-Server>2
[Resuming connection 2 to r5 ... ]
01:03:37: NAT: s=192.168.12.2->192.168.34.1, d=192.168.23.3 [75]
01:03:37: NAT*: s=192.168.23.3, d=192.168.34.1->192.168.12.2 [75]
省略...
wanhe2#sh ip nat tr
Pro Inside global      Inside local      Outside local      Outside global
icmp 192.168.34.1:7    192.168.12.2:7    192.168.23.3:7    192.168.23.3:7
```

注：端口号已改为 7。

```
wanhe2#
01:04:37: NAT: expiring 192.168.34.1 (192.168.12.2) icmp 7 (7)
```

## 实验十六 交换机的基础配置

### 实验要求：

- 1.主机名更改为 wanhe
- 2.完成交换机 IP 地址配置，IP 地址为 202.119.249.250 255.255.255.0，网关为 202.119.249.2
- 3.MAC 地址绑定,mac 地址为 0010.7a60.1884
- 4.配置 telnet 时用到的用户密码和 vty 密码。分别为 ccna 和 ccnp。

### 实验过程：（以 2950 交换机为例）

#### 1.完成交换机重命名

```
Switch>enable
Switch#configure terminal
Switch(config)#hostname wanhe
```

#### 2.设置 TELNET 时必要的密码

```
wanhe(config)#enable password ccna
wanhe(config)#line vty 0 4
wanhe(config-if)#password ccnp
wanhe(config-if)#exit
```

注：设置交换机的特权密码。设置交换机的虚拟登陆时的密码。

### 3.设置管理 VLAN 的 IP 地址

```
wanhe(config)#interface vlan 1
wanhe(config-if)#ip address 202.119.249.250 255.255.255.0
wanhe(config-if)#no shutdown
wanhe(config-if)#exit
wanhe(config)#ip default-gateway 202.119.249.2
```

注：进入接口 vlan1。配置管理 ip 地址。置交换机的网关。

### 4.完成 MAC 地址的绑定

```
wanhe(config)#mac-address-table static 0010.7a61.1884 vlan 1 interface fastethernet
```

0/5

注：将 MAC 地址加入到地址表中

### 5.测试实验效果

将 PC 终端设置 ip 地址 202.119.249.251，然后与交换机的 fa0/8 相连，并打开“开始”菜单——“运行”——“cmd”，接着按照下面给出的 DOS 所显示的信息来验证实验。

telnet 登陆后，分别输入 vty 密码和特权密码。然后 show mac-address-table 显示 mac 地址表。

## 实验十七 交换机密码恢复

### 实验要求

- 1、掌握交换机的密码恢复原理
- 2、区分交换机和路由器密码恢复的不同

### 实验步骤

#### 1、设置交换机加密并保存配置

```
Switch>en
Switch#conf t
Switch(config)#enable password 123
Switch(config)#line console 0
Switch(config-line)#password 123
Switch(config-line)#login
Switch(config-line)#end
```



注：进入线路中控制台。设置控制台密码。

## 2. 保存配置

Switch#**copy running-config startup-config**

Destination filename [startup-config]?

Building configuration...

[OK]

Switch#**reload**

注：保存配置。起交换机。这次软重启，也可以拔掉交换机电源，然后再接上电源。

当交换机重起时，按住交换机前面板的 **Mode** 键不放，显示以下的信息。

## 3. 修改启动文件名

Base ethernet MAC Address: 00:0c:30:51:56:00

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...

省略...

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

switch: **load\_helper**

## 4. 查看配置文件

switch: **dir flash:**-----显示 flash 中的文件。

Directory of flash:/

```

 2    -rwx   1193    <date>          config.text
 3 -rwx 2958970    <date>  c2950-i6q4l2-mz.121-14.EA1a.bin
 4    drwx   2304    <date>          html
 79   -rwx    5     <date>          private-config.text
2482688 bytes avai (5258752 bytes used)
```

注：config.text 是交换机的启动配置文件，和路由器的 startup-config 类似。

## 5. 修改配置文件名

switch: **rename flash:config.text flash:config.old**

注：将启动配置文件改名，这样交换机启动时就读不到 config.text 了，从而没有了密码。

## 6.重启交换机

switch: **boot**-----引导系统，这时不再按住 Mode 键了。

Loading

```
"flash:/c2950-i6q4l2-mz.121-14.EA1a.bin"...#####
#####
#####
省略...
```

## 7.修改密码

Switch>en

Switch#show flash

Directory of flash:/

```
2  -rwx  1193   Mar 01 1993 00:01:20  config.old
```

注：启动文件名被修改所以不会被加载。

```
3 -rwx 2958970 Mar 01 1993 00:04:42 c2950-i6q4l2-mz.121-14.EA1a.bin
```

```
4  drwx          2304   Mar 01 1993 00:05:47  html
```

```
79  -rwx 5 Mar 01 1993 00:01:20  private-config.text
```

Switch#rename **flash:config.old flash:config.text**

注：将启动的文件名改回到正常的文件名。

Switch#**copy flash:config.text running-config**

Destination filename [running-config]?

1193 bytes copied in 1.312 secs (909 bytes/sec)

注：将配置文件加载到 RAM 中。

Switch#conf t

Switch(config)#**no enable password**-----删除密码。

Switch(config)#line con 0

Switch(config-line)#**no password**-----删除控制台的密码。

Switch(config-line)#**no login**-----不要求密码验证。

Switch(config-line)#end

## 8. 保存配置文件

Switch#copy running-config startup-config

Destination filename [startup-config]?

Building configuration...

[OK]

注：将密码删除后的配置文件保存。

Switch#**exit**

注：退出用户模式，进行验证是否还要求密码验证。或将交换机重新启动检查是否要求

密码验证。

Switch con0 is now available  
Press RETURN to get started.

Switch>**en**

Switch#

注：很明显没要求密码验证。密码已被删除。

\

## 实验十八 交换机端口安全

### 实验要求：

1. 启用端口的安全措施
2. 限制 fa0/23 口最大允许访问量为 1
3. 采取的安全措施为保护，限制或关闭
4. 常用的交换机命令

### 实验过程：

#### 1. 启用端口的安全措施

wanhe(config)#**interface fastethernet 0/23**  
wanhe(config-if)#**switchport mode access**

注：将此接口定义为主机端口，交换机的连接类型有接入模式和中继模式。

wanhe(config-if)#**switchport port-security**

注：启动交换机的端口安全特性。

wanhe(config-if)#**switchport port-security mac-address aaaa.aaaa.aaaa**

注：将 mac 地址 aaaa.aaaa.aaaa 绑定在接口 0/23 上。

wanhe(config-if)#**switchport port-security maximum 1**

注：设置可以安全访问的用户有 1 个。

wanhe(config-if)#**switchport port-security violation shutdown**

注：设置当接口上的访问违反了安全特性时，所采用的惩罚措施。

惩罚措施有保护、限制和关闭。关闭：当新的计算机接入时，如果该接口的

MAC 条目超过了最大数目，则该接口将会被关闭，则这个新的计算机和原来的计算

机都无法接入，需要管理员使用“no shutdown”命令从新打开。

```
wanhe(config-if)#exit
```

实验检测：用一根直通线将 PC 和交换机的 0/23 口相连，查看 0/23 接口上的指示灯的变化情况。如果由橙色经过大约 50 秒的时间变为绿色后再立即关闭，说明实验成功，并在交换机上有如下的显示：

```
00:19:19: %LINK-3-UPDOWN: Interface FastEthernet0/23, changed state to up
```

```
00:19:20: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/23, changed state to up
```

```
00:19:52: %PM-4-ERR_DISABLE: psecure-violation error detected on Fa0/23, putting Fa0/23 in err-disable state
```

```
00:19:52: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred, caused by MAC address 0000.b420.0742 on port FastEthernet0/23.
```

```
00:19:53: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/23, changed state to down
```

```
00:19:54: %LINK-3-UPDOWN: Interface FastEthernet0/23, changed state to down-----  
说明执行了 shutdown 的惩罚措施。
```

```
wanhe#show interfaces fastethernet 0/23
```

```
FastEthernet0/23 is down, line protocol is down (err-disabled)
```

```
Hardware is Fast Ethernet, address is 000c.3051.5617 (bia 000c.3051.5617)
```

注：表示是出现错误时，关闭了接口。

## 2、选择其他端口测试另外两种惩罚措施的现象

```
wanhe(config)#interface fastethernet 0/22
```

注：选择另一个端口进行测试。

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

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

```
wanhe(config-if)#switchport port-security mac-address 15.15.15
```

```
wanhe(config-if)#switchport port-security mac-address 14.14.14
```

注：将安全地址 0015.0015.0015 和 0014.0014.0014 绑定到端口上。

```
wanhe(config-if)#switchport port-security maximum 2
```

注：定义端口最多可以让两个用户访问。

wanhe(config-if)#**switchport port-security violation restrict**

注：设置当接口上的访问违反了安全特性时，所采用的惩罚措施。

惩罚措施有保护、限制和关闭。限制：当新的当新的计算机接入时，如果该接口的

MAC 条目超过了最大数目，则这个新的计算机可以接入，然而交换机将要发送警告

信息。保护：当新的计算机接入时，如果该接口的 MAC 条目超过了最大数目， 则

这个新的计算机将无法接入，而原来的计算机不受影响。

将 PC 连接到交换机的 fa0/22 端口上，查看现象。

02:54:23: %PORT\_SECURITY-2-PSECURE\_VIOLATION: Security violation occurred, caused by MAC address 0000.b420.0742 on port FastEthernet0/21.

注：交换机不停的报错，原因是接入的地址超过了最大的数目。

## 实验十九 单臂路由

实验要求：

1. 将交换机划分 2 个 VLAN
2. 将交换机 fa0/1,fa0/2,fa0/4 划入 VLAN2， fa0/3,fa0/5 到 fa0/8 划入 VLAN3， fa0/10 配置成 Trunk
3. 将路由器的 Fa0/0 口与交换机 fa0/10 口连接，配置中继路由

注：本实验可以使用 Packet Tracer 来完成本实验。



实验过程：

1. 交换机创建 vlan

```
wanhe1#configure terminal
wanhe1(config)#vlan 2
wanhe1(config-vlan)#name cisco
```

```
wanhe1(config-vlan)#vlan 3
wanhe1(config-vlan)#name wanhe
```

## 2.把相应的端口划分到 VLAN 中

```
wanhe1(config)#interface range fastEthernet 0/1 -2 , fastEthernet 0/4
```

注：使用关键字可以同时多个接口做相同的设置。用短横线来表示范围。用逗号将不

连续的接口隔开。

```
wanhe1(config-if-range)#switchport mode access
wanhe1(config-if-range)#switchport access vlan 2
```

注：定义接口的连接方式为 *access*。将这些接口加入到 *vlan2* 中。

```
wanhe 1(config-if-range)#interface range fastEthernet 0/3 , fastEthernet 0/5 -8
wanhe1(config-if-range)#switchport mode access
wanhe1(config-if-range)#switchport access vlan 3
```

## 3.查看 VLAN 信息

```
wanhe1#show vlan brief
```

VLAN Name	Status	Ports
1 default	active	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
<b>2 cisco</b>	<b>active</b>	<b>Fa0/1, Fa0/2, Fa0/4</b>
<b>3 wanhe</b>	<b>active</b>	<b>Fa0/3, Fa0/5, Fa0/6, Fa0/7 Fa0/8</b>

注：Fa0/1, Fa0/2, Fa0/4 已加入到 vlan 2 中。Fa0/3, Fa0/5, Fa0/6, Fa0/7

Fa0/8 加入到 vlan 3 中。

## 4. 设置 trunk 端口

```
wanhe1(config)#interface fastEthernet0/10
wanhe1(config-if)#switchport mode trunk
```

注：定义接口 fa0/10 的连接类型为 *trunk*。

5.将做实验用的 PC 上配置 ip 地址 192.168.2.3 255.255.255.0。网关为 192.168.2.1。然后

将 PC 和路由器模拟的 PC1 分别与交换机的 vlan2 中的接口相连。

## 6.配置 2600 路由器

```
Router>enable
Router#conf t
Router(config)#hostname jiacenj

wanhenj(config)#interface fastethernet 0/0
wanhenj(config-if)#no ip address
wanhenj(config-if)#no shutdown
wanhenj(config-if)#exit
```

注：禁止物理接口的 ip 地址。

```
wanhenj(config)#interface fastethernet 0/0.1
wanhenj(config-subif)#encapsulation dot1q 2
wanhenj(config-subif)#ip address 192.168.2.1 255.255.255.0
wanhenj(config-subif)#exit
```

注：定义子接口 fa0/0.1。将子接口封装成 802.1Q 的标准，并和 vlan 2 关联。配置 vlan 2

的网关地址。

```
wanhenj(config)#interface fastethernet 0/0.2
wanhenj(config-subif)#encapsulation dot1q 3
wanhenj(config-subif)#ip address 192.168.3.1 255.255.255.0
wanhenj(config-subif)#exit
```

注：定义子接口 fa0/0.2。将子接口封装成 802.1Q 的标准。和 vlan 3 关联。配置 vlan 3 的

网关地址。

## 7、用终端 PC ping 路由器上另外一个网段的网关地址，查看是否联通

```
C:\Documents and Settings\Administrator>ping 192.168.3.1
```

```
Pinging 192.168.2.2 with 32 bytes of data:
```

```
Reply from 192.168.2.2: bytes=32 time=2ms TTL=254
```

注：可以 ping 通到对端。

```
Reply from 192.168.2.2: bytes=32 time=2ms TTL=254
```

```
Reply from 192.168.2.2: bytes=32 time=2ms TTL=254
```

## 实验二十 PAP 认证实验

### 实验要求:

1. 为路由器指定唯一主机名;
2. 列出认证路由器时所使用的远端主机名称和口令;
3. WAN 接口上完成 PPP 协议的封装;
4. wanhe1 为服务端, wanhe2 为客户端, 客户端主动向服务端发出认证请求, 密码设置为 ccna.



### 实验过程:

#### 1、配置 wanhe1 的服务段设置

```
Router(config)#hostname wanhe1
```

注：配置唯一主机名。

```
wanhe1(config)#username wanhe password ccna
```

注：列出本地用户名和口令列表。

```
wanhe1(config)#interface serial 0
wanhe1(config-if)#ip address 192.168.12.1 255.255.255.0
wanhe1(config-if)#encapsulation ppp
```

注：启动 PPP 封装协议。默认情况下是 HDLC 进行封装。

```
wanhe1(config-if)#ppp authentication pap
wanhe1(config-if)#no shutdown
```

注：设置成服务器端并启用了 PAP 身份验证协议。

#### 2、配置 Jiance2 的客户端配置

```
Router(config)#hostname wanhe2
wanhe2(config)#interface serial 0
wanhe2(config-if)#ip address 192.168.12.2 255.255.255.0
wanhe2(config-if)#clock rate 64000
wanhe2(config-if)#encapsulation ppp
wanhe2(config-if)#no shutdown
```

问题分析：这时如果不设置被验证方所要放送的用户名和列表，将会不停的报错显示如下：



```
00:13:02: %LINK-3-UPDOWN: Interface Serial0, changed state to up
00:13:10: %LINK-3-UPDOWN: Interface Serial0, changed state to down
00:13:20: %LINK-3-UPDOWN: Interface Serial0, changed state to up
00:13:27: %LINK-3-UPDOWN: Interface Serial0, changed state to down
```

### 3.client 端发送用户名和密码

```
wanhe2(config-if)#ppp pap sent-username adsf password asdf
```

注：设置被验证方发送的用户名。设置被验证方发送的口令。当用户名和口令中的任意一个

和验证方的本地用户列表不同时，不再报错但同样无法通信。

```
wanhe2(config-if)#end
wanhe2#ping 192.168.12.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
.....-----信息不一致时，无法通信。
Success rate is 0 percent (0/5)
```

### 4. 设置正确的用户名和密码

```
wanhe2(config-if)#ppp pap sent-username wanhe password ccna
```

注：发送用于验证的用户名和口令，向服务器发起认证。注意：所发送的用户名不一定必须

是 hostname。

### 5.测试连通性

```
wanhe2#ping 192.168.12.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
!!!!-----说明验证通过。
```

## 实验二十一 CHAP 认证实验

### 实验要求：

- 1.为路由器指定唯一主机名
- 2.列出认证路由器时所使用的远端主机名称和口令，密码为 ccna.
- 3.WAN 接口上完成 PPP 协议的封装和 CHAP 认证的配置



### 实验过程:

#### 1、配置 wanhe1

```
Router(config)#hostname wanhe1
wanhe1(config)#username wanhe2 password ccna
```

注：用于验证对端发送过来的用户名和口令，用户名必须是对端的 hostname，而口令在两端必须要一样。

```
wanhe1(config)#interface serial 0
wanhe1(config-if)#ip address 192.168.12.1 255.255.255.0
wanhe1(config-if)#encapsulation ppp
wanhe1(config-if)#ppp authentication chap
wanhe1(config-if)#no shutdown
```

注：启用 CHAP 认证协议。CHAP 是双向验证协议。并使用 hostname 作为用户名去被验证，用本地用户列表来验证对端。

#### 2、配置 wanhe2

```
Router(config)#hostname wanhe2
wanhe2(config)#interface serial 0
wanhe2(config-if)#ip address 192.168.12.2 255.255.255.0
wanhe2(config-if)#clock rate 64000
wanhe2(config-if)#encapsulation ppp
wanhe2(config-if)#ppp authentication chap
wanhe2(config-if)#no shutdown
wanhe2(config-if)#exit
```

注：在对端需要配置相同的，CHAP 是双向认证。

```
00:55:48: %LINK-3-UPDOWN: Interface Serial0, changed state to down
00:55:50: %LINK-3-UPDOWN: Interface Serial0, changed state to up
00:55:58: %LINK-3-UPDOWN: Interface Serial0, changed state to down
00:56:00: %LINK-3-UPDOWN: Interface Serial0, changed state to up
```

注：由于没有配置用于验证发送过来的本地用户名和列表，导致了不停的报错。

### 3.设置 wanhe2 上的用户名和密码

```
wanhe2(config)#username wanhe1 password ccnp
```

注：wanhe2 口令 ccnp,配置了本地用户列表不再报错。

### 4. 测试连接效果

```
wanhe2#ping 192.168.12.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:

.....~~-----~~ 口令不一致，无法建立连接。

在 wanhe1 上 debug ppp authentication

```
*Mar  1 00:02:45.251: Se0/0 CHAP: O CHALLENGE id 18 len 28 from "wanhe1"
*Mar  1 00:02:45.255: Se0/0 CHAP: I CHALLENGE id 16 len 28 from "wanhe2"
*Mar  1 00:02:45.271: Se0/0 CHAP: Using hostname from unknown source
*Mar  1 00:02:45.275: Se0/0 CHAP: Using password from AAA
*Mar  1 00:02:45.275: Se0/0 CHAP: O RESPONSE id 16 len 28 from "wanhe1"
*Mar  1 00:02:45.391: Se0/0 CHAP: I RESPONSE id 18 len 28 from "wanhe2"
*Mar  1 00:02:45.395: Se0/0 CHAP: I FAILURE id 16 len 25 msg is "Authentication failed"
*Mar  1 00:02:45.407: Se0/0 PPP: Sent CHAP LOGIN Request
```

### 5.设置正确的用户名和密码

```
wanhe2(config)#username wanhe1 password ccna
```

注：置和 wanhe1 的本地用户列表相同的口令。

### 6.测试连接效果

```
wanhe2#ping 192.168.12.1
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:

!!!!!

### 7.在 wanhe2 上使用 debug ppp authentication 查看 chap 的认证过程

```
wanhe2#debug ppp authentication
```

```
wanhe2#conf t
```

```
wanhe2(config)#interface serial 0
```

```
wanhe2(config-if)#shutdown
```

```
wanhe2(config-if)#no shutdown
```

```
*Dec  1 21:09:44.943: %LINK-3-UPDOWN: Interface Serial0, changed state to up
*Dec  1 21:09:44.951: Se0 PPP: Authorization required
*Dec  1 21:09:45.527: Se0 PPP: Sent CHAP LOGIN Request
*Dec  1 21:09:45.531: Se0 CHAP: Using hostname from unknown source
```

```
*Dec 1 21:09:45.531: Se0 CHAP: Using password from AAA
*Dec 1 21:09:45.531: Se0 CHAP: O RESPONSE id 215 len 23 from "wanhe2"
*Dec 1 21:09:45.779: Se0 CDPCP: Received AAA AUTHOR Response PASS
*Dec 1 21:09:46.759: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0,
changed state to up
```

4、在验证都通过的情况下将任何一边的口令随便设置一个不要和 ccna 一样，然后 ping 对端会出现什么情况，为什么会出现这种现象，怎么解决？

```
wanhe1(config)#no username wanhe2
wanhe1(config)#username wanhe2 password ccnp
wanhe1(config)#end
```

注：口令为 ccnp 不和 ccna 一样。

```
wanhe1#ping 192.168.12.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.2, timeout is 2 seconds:
```

!!!!-----两边的口令不一样也可以 ping 通。

Success rate is 100 percent (5/5), round-trip min/avg/max = 48/144/244 ms

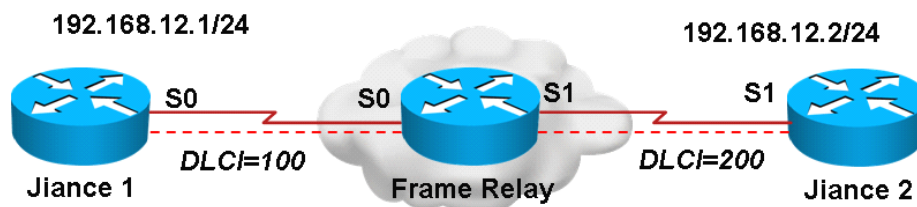
问题分析：开始的时候两边的口令不一样无法验证通过，当验证通过后，再将口令改为不一致。同样可以 ping 通。

因为当验证通过后会一直保存已经建立好的连接。解决的方法是将接口关闭后然后再启动。

## 实验二十二 帧中继实验

### 实验要求：

- 1、使用 cisco 路由器模拟帧中继交换机
- 2、完成帧中继实验要求，具体参数图中已给出



### 实验过程：

- 1、配置中间的帧中继交换机

```
Fr-sw(config)#frame-relay switching
```

注：启用 router 的帧中继功能。

```
Fr-sw(config)#interface serial 0
Fr-sw(config-if)#encapsulation frame-relay
```

注：封装帧中继协议。默认情况下是用 cisco 来封装。

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

注：指定 s0 口为 dce 端。

```
Fr-sw(config-if)#clock rate 64000
Fr-sw(config-if)#frame-relay route 100 interface serial 1 200
```

注：建立桥接，源 DLCI 号 100 经过 s1 口到目的地 DLCI 号 200 线路。

```
Fr-sw(config-if)#no shutdown
Fr-sw(config-if)#exit
```

```
Fr-sw(config)#interface serial 1
Fr-sw(config-if)#encapsulation frame-relay
Fr-sw(config-if)#frame-relay intf-type dce
Fr-sw(config-if)#clock rate 64000
Fr-sw(config-if)#frame-relay route 200 interface serial 0 100
Fr-sw(config-if)#no shutdown
```

## 2、配置 wanhe1

```
wanhe1(config)#interface serial 0
wanhe1(config-if)#ip address 192.168.12.1 255.255.255.0
wanhe1(config-if)#encapsulation frame-relay
wanhe1(config-if)#no shutdown
```

## 3、配置 wanhe2

```
wanhe2(config)#interface serial 1
wanhe2(config-if)#ip address 192.168.12.2 255.255.255.0
wanhe2(config-if)#encapsulation frame-relay
wanhe2(config-if)#no shutdown
```

## 4、验证实验

```
wanhe2#ping 192.168.12.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.12.1, timeout is 2 seconds:
!!!!
```

```
wanhe2#show frame-relay map
Serial1 (up): ip 192.168.12.1 dlci 200(0xC8,0x3080), dynamic,
broadcast, status defined, active
```

注：本地的 dlci=200 与对端 192.168.12.1 通过逆向 arp 功能自动学习得到动态 map。

当发送数据给 192.168.12.1 时，就从 200 虚电路发送出去。

Fr-sw#**show frame-relay route**

Input Intf	Input Dlc	Output Intf	Output Dlc	Status
Serial0	100	Serial1	200	active
<b>Serial1</b>	<b>200</b>	<b>Serial0</b>	<b>100</b>	<b>active</b>

注：通过命令可以查看在帧中继交换机上虚电路交换的过程。从接口 s1 的 200 虚电路交换

到 s0 的 100 的虚电路。

wanher1#show frame-relay pvc

PVC Statistics for interface Serial0 (Frame Relay DTE)

	Active	Inactive	Deleted	Static
Local	1	0	0	0
Switched	0	0	0	0
Unused	0	0	0	0

**DLCI = 100, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0**

注：路由器的虚电路 100 在 s0 上，这样从交换 s0 过来的数据就会发送给路由器的 s0 上。

input pkts 8	output pkts 7	in bytes 622
out bytes 588	dropped pkts 0	in pkts dropped 0
out pkts dropped 0	out bytes dropped 0	
in FECN pkts 0	in BECN pkts 0	out FECN pkts 0
out BECN pkts 0	in DE pkts 0	out DE pkts 0
out bcast pkts 2	out bcast bytes 68	
5 minute input rate 0 bits/sec, 0 packets/sec		
5 minute output rate 0 bits/sec, 0 packets/sec		
pvc create time 00:35:28, last time pvc status changed 00:33:18		

注：也可以在 wanhe1 和 wanhe2 上分别禁用逆向 ARP 查询，手动配置 DLCI 号与 IP 地址的

映射

wanhe1 的配置

wanhe1(config-if)#no frame-relay inverse-arp

wanhe1(config-if)#frame-relay map ip 192.168.12.2 100 broadcast

wanhe2 的配置

wanhe2(config-if)#no frame-relay inverse-arp

wanhe2(config-if)#frame-relay map ip 192.168.12.1 200 broadcast

```
wanhe1#show frame-relay map  
Serial0 (up): ip 192.168.12.2 dlci 100(0x64,0x1840), static,  
                broadcast,  
                CISCO, status defined, active
```

## 实验二十三 用 **SDM** 管理路由器

### 实验要求:

1. 设置R1使其能够通过SDM进行管理

注意：本实验需要使用2600以上路由器，2500路由器不支持，可以使用Dynamips模拟器



### 实验步骤:

- 1: 配置基本IP地址
- 2: 在R1上进行设置，使其能够通过SDM连接

```
R1(config)#username stsd privilege 15 secret cisco
```

创建15级用户

```
R1(config)#ip http server
```

打开R1的http服务器

```
R1(config)#ip http authentication local
```

将http认证设置为使用本地认证数据库

- 3: 通过SDM连接R1

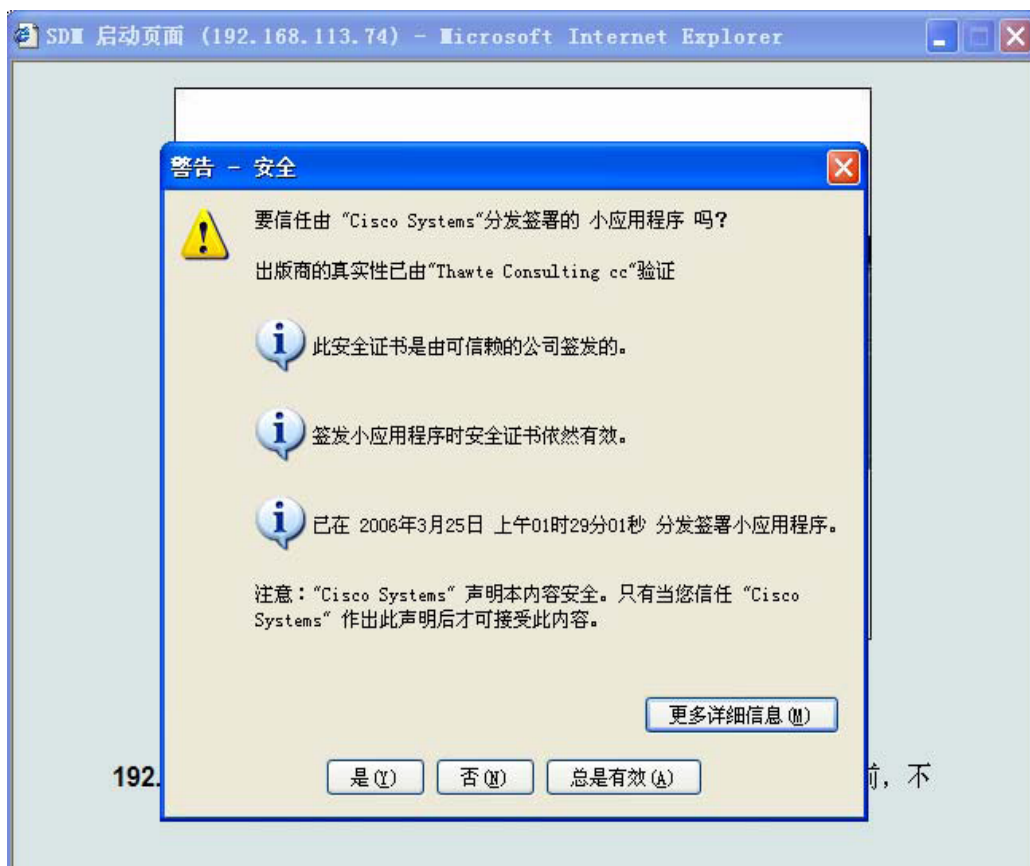


在地址栏中输入 R1 的 IP 地址。



输入在 R1 中创建的 15 级的用户名与密码





同意 java 安全警告



再输入一次用户名密码



连接完成，现在就可以通过 SDM 对 R1 进行管理了。