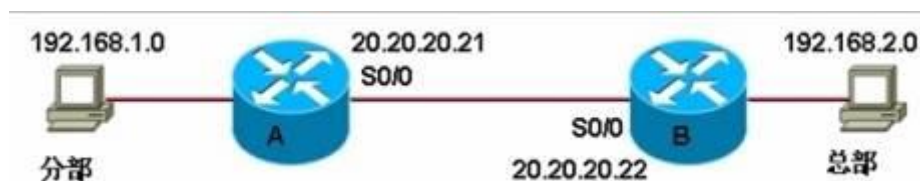


## 01-IPSec VPN 的基本配置



环境：接口地址都已经配置完，路由也配置了，双方可以互相通信。

密钥认证的算法 2 种：md5 和 sha1

加密算法 2 种：des 和 3des

IPsec 传输模式 3 种：

AH 验证参数：ah-md5-hmac(md5 验证)、ah-sha-hmac(sha1 验证)

ESP 加密参数：esp-des(des 加密)、esp-3des(3des 加密)、esp-null（不对数据进行加密）

ESP 验证参数：esp-md5-hmac(md5 验证)、esp-sha-hmac(采用 sha1 验证)

### 1 启用 IKE 协商

路由器 A

```
routerA(config)#crypto isakmp policy 1 //建立 IKE 协商策略（1 是策略编号 1-1000，号越小，优先级越高）
```

```
routerA(config-isakmp)#hash md5 //选用 md5 密钥认证的算法
```

```
routerA(config-isakmp)#authentication pre-share //告诉路由使用预先共享的密钥
```

```
routerA(config)#crypto isakmp key 12345 address 20.20.20.22 //12345 是设置的共享密钥,20.20.20.22 是对端的 IP
```

路由器 B

```
routerB(config)#crypto isakmp policy 1
```

```
routerB(config-isakmp)#hash md5
```

```
routerB(config-isakmp)#authentication pre-share
```

```
routerB(config)#crypto isakmp key 12345 address 20.20.20.21 //路由 B 和 A 的配置除了这里的对端 IP 地址，其它都要一样的）。
```

### 2 配置 IPSec 相关参数

路由器 A

```
routerA(config)#crypto ipsec transform-set test ah-md5-hmac esp-des
```

//test 传输模式的名称。ah-md5-hmac esp-des 表示传输模式中采用的验证参数和加密参数。

```
routerA(config)#access-list 101 permit ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255
```

//定义哪些地址的报文加密或是不加密。

路由器 B

```
routerB(config)#crypto ipsec transform-set test ah-md5-hmac esp-des
```

```
routerB(config)#access-list 101 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
```

//路由器 B 和 A 的配置除了这里的源和目的 IP 地址变了，其它都一样。

### 3 设置 crypto map （目的把 IKE 的协商信息和 IPSec 的参数，整合到一起，起一个名字）

路由器 A

```
routerA(config)#crypto map testmap 1 ipsec-isakmp
```

//testmap:给 crypto map 起的名字。1: 优先级。ipsec-isakmp:表示此 IPSec 链接采用 IKE 自动协商

```
routerA(config-crypto-map)#set peer 20.20.20.22 //指定此 VPN 链路，对端的 IP 地址
```

```
routerA(config-crypto-map)#set transform-set test //IPSec 传输模式的名称。
```

```
routerA(config-crypto-map)#match address 101 //上面定义的 ACL 列表号
```

路由器 B

---

```
routerB(config)#crypto map testmap 1 ipsec-isakmp
routerB(config-crypto-map)#set peer 20.20.20.21 （和 A 路由的配置只有这里的对端 IP 不一样）
routerB(config-crypto-map)#set transform-set test
routerB(config-crypto-map)#match address 101
```

#### 4 把 **crypto map** 的名字应用到端口

```
routerA(config)#inter s0/0 (进入应用 VPN 的接口)
routerA(config-if)#crypto map testmap （testmap: crypto map 的名字）
```

B 路由器和 A 设置完全一样。

#### 5 查看 VPN 的配置

```
Router#show crypto ipsec sa //查看安全联盟（SA）
router#show crypto map //显示 crypto map 内的所有配置
router#show crypto isakmp policy //查看优先级
```

## 02-IPSec VPN 的基本配置

### PIX 配置专题 六个基本命令

PIX 是 CISCO 公司开发的防火墙系列设备，主要起到策略过滤，隔离内外网，根据用户实际需求设置 DMZ（停火区）。它和一般硬件防火墙一样具有转发数据包速度快，可设定的规则种类多，配置灵活的特点。配置 PIX 防火墙有六个基本命令：nameif, interface, ip address, nat, global, route。

我们先掌握这六个基本命令，然后再学习更高级的配置语句。

#### 1、nameif 配置防火墙接口的名字，并指定安全级别：

```
Pix525(config)#nameif ethernet0 outside security0 //命名 e0 为 outside,安全级别为 0。该名称在后面使用
Pix525(config)#nameif ethernet1 inside security100 //命名 e1 为 inside，安全级别为 100。安全系数最高
Pix525(config)#nameif dmz security50 //设置 DMZ 接口为停火区，安全级别 50。安全系数居中。
```

在该配置中，e0 被命名为外部接口（outside），安全级别是 0；以 e1 被命名为内部接口（inside），安全级别是 100。安全级别取值范围为 1 到 99，数字越大安全级别越高。

#### 2、配置以太网口参数（interface）：

```
Pix525(config)#interface ethernet0 auto //设置 e0 为 AUTO 模式，auto 选项表明系统网卡速度工作模式等为自动适应，这样该接口会自动在 10M/100M，单工/半双工/全双工直接切换。
```

```
Pix525(config)#interface ethernet1 100 full //强制设置以太接口 1 为 100Mbit/s 全双工通信。
```

```
Pix525(config)#shutdown //关闭端口
```

小提示：在节假日需要关闭停火区的服务器的服务时可以在 PIX 设备上使用 interface dmz 100full shutdown,这样 DMZ 区会关闭对外服务。

#### 3、配置内外网卡的 IP 地址（ip address）

```
Pix525(config)#ip address outside 61.144.51.42 255.255.255.248 //设置外网接口为 61.144.51.42，子网掩码为 255.255.255.248
```

```
Pix525(config)#ip address inside 192.168.0.1 255.255.255.0 //设置内网接口 IP 地址、子网掩码。
```

你可能会问为什么用的是 outside 和 inside 而没有使用 ethernet1, ethernet0 呢？其实这样写是为了方便我们配置，不容易出错误。只要我们通过 nameif 设置了各个接口的安全级别和接口类别，接口类别就代表了相应的端口，也就是说 outside=ethernet0, inside=ethernet1。

#### 4、指定要进行转换的内部地址（nat）

NAT 的作用是将内网的私有 ip 转换为外网的公有 ip，Nat 命令总是与 global 命令一起使用，这是因为 nat 命令可以指定一台主机或一段范围的主机访问外网，访问外网时需要利用 global 所指定的地址池进行对外访问。

nat 命令配置语法：nat (if\_name) nat\_id local\_ip [netmark]

---

其中（if\_name）表示内网接口名字，如 inside，Nat\_id 用来标识全局地址池，使它与其相应的 global 命令相匹配，local\_ip 表示内网被分配的 ip 地址。例如 0.0.0.0 表示内网所有主机可以对外访问。[netmark]表示内网 ip 地址的子网掩码。示例语句如下：

```
Pix525(config)#nat (inside) 1 0.0.0.0 0.0.0.0 //启用 nat,内网的所有主机都可以访问外网，可以用 0 代表 0.0.0.0
```

```
Pix525(config)#nat (inside) 1 172.16.5.0 255.255.0.0 //设置只有 172.16.5.0 这个网段内的主机可以访问外网。
```

## 5、指定外部地址范围（global）

global 命令把内网的 ip 地址翻译成外网的 ip 地址或一段地址范围。

Global 命令的配置语法：global (if\_name) nat\_id ip\_address-ip\_address [netmark global\_mask]

其中（if\_name）表示外网接口名字，如 outside；Nat\_id 用来标识全局地址池，使它与其相应的 nat 命令相匹配；

ip\_address-ip\_address 表示翻译后的单个 ip 地址或一段 ip 地址范围；[netmark global\_mask]表示全局 ip 地址的网络掩码。示例语句如下：

```
Pix525(config)#global (outside) 1 61.144.51.42-61.144.51.48 //设置内网的主机通过 pix 防火墙要访问外网时，pix 防火墙将使用 61.144.51.42-61.144.51.48 这段 ip 地址池为要访问外网的主机分配一个全局 ip 地址。
```

```
Pix525(config)#global (outside) 1 61.144.51.42 //设置内网要访问外网时，pix 防火墙将为访问外网的所有主机统一使用 61.144.51.42 这个单一 ip 地址。
```

```
Pix525(config)#no global (outside) 1 61.144.51.42 //删除 global 中对 61.144.51.42 的宣告，也就是说数据包通过 NAT 向外传送时将不使用该 IP，这个全局表项被删除。
```

## 6、设置指向内网和外网的静态路由（route）

route 命令定义一条静态路由。

route 命令配置语法：route (if\_name) 0 0 gateway\_ip [metric]

其中（if\_name）表示接口名字，例如 inside，outside。Gateway\_ip 表示网关路由器的 ip 地址。[metric]表示到 gateway\_ip 的跳数。通常缺省是 1。示例语句如下：

```
Pix525(config)#route outside 0 0 61.144.51.168 1 //设置一条指向边界路由器（ip 地址 61.144.51.168）的缺省路由。
```

```
Pix525(config)#route inside 10.1.1.0 255.255.255.0 172.16.0.1 1 //设置一条指向内部的路由。
```

```
Pix525(config)#route inside 10.2.0.0 255.255.0.0 172.16.0.1 1 //设置另一条指向内部的路由。
```

**总结：**目前我们已经掌握了设置 PIX 的六大基本命令，通过这六个命令我们已经可以让 PIX 为我们的网络服务了。不过让网络运行还远远不够，我们要有效的利用网络，合理的管理网络，这时候就需要一些高级命令了。

## PIX 防火墙高级配置命令

### 1、配置静态 IP 地址翻译（static）：

如果从外网发起一个会话，会话目的地址是一个内网的 ip 地址，static 就把内部地址翻译成一个指定的全局地址，允许这个会话建立。

static 命令配置语法：static (internal\_if\_name, external\_if\_name) outside\_ip\_address inside\_ip\_address

其中 internal\_if\_name 表示内部网络接口，安全级别较高。如 inside。external\_if\_name 为外部网络接口，安全级别较低，如 outside 等。outside\_ip\_address 为正在访问的较低安全级别的接口上的 ip 地址。inside\_ip\_address 为内部网络的本地 ip 地址。示例语句如下：

```
Pix525(config)#static (inside, outside) 61.144.51.62 192.168.0.8 //ip 地址为 192.168.0.8 的主机，对于通过 pix 防火墙建立的每个会话，都被翻译成 61.144.51.62 这个全局地址，也可以理解成 static 命令创建了内部 ip 地址 192.168.0.8 和外部 ip 地址 61.144.51.62 之间的静态映射。PIX 将把 192.168.0.8 映射为 61.144.51.62 以便 NAT 更好的工作。
```

小提示：使用 static 命令可以让我们为一个特定的内部 ip 地址设置一个永久的全局 ip 地址。这样就能够为具有较低安全级别的指定接口创建一个入口，使它们可以进入到具有较高安全级别的指定接口。

### 2、管道命令（conduit）：

使用 static 命令可以在一个本地 ip 地址和一个全局 ip 地址之间创建了一个静态映射，但从外部到内部接口的连接仍然会被 pix 防火墙的自适应安全算法(ASA)阻挡，conduit 命令用来允许数据流从具有较低安全级别的接口流向具有较高安全级别的接口，例如允许从外部到 DMZ 或内部接口的入方向的会话。对于向内部接口的连接，static 和 conduit 命令将一起使用，来指定会话的建立。说得通俗一点管道命令（conduit）就相当于以往 CISCO 设备的访问控制列表（ACL）。

---

**conduit 命令配置语法:** `conduit permit|deny global_ip port[-port] protocol foreign_ip [netmask]`

其中 `permit|deny` 为允许|拒绝访问, `global_ip` 指的是先前由 `global` 或 `static` 命令定义的全局 ip 地址, 如果 `global_ip` 为 0, 就用 `any` 代替 0; 如果 `global_ip` 是一台主机, 就用 `host` 命令参数。 `port` 指的是服务所作用的端口, 例如 `www` 使用 80, `smtp` 使用 25 等等, 我们可以通过服务名称或端口数字来指定端口。 `protocol` 指的是连接协议, 比如: `TCP`、`UDP`、`ICMP` 等。 `foreign_ip` 表示可访问 `global_ip` 的外部 ip。对于任意主机可以用 `any` 表示。如果 `foreign_ip` 是一台主机, 就用 `host` 命令参数。

示例语句如下:

`Pix525(config)#conduit permit tcp host 192.168.0.8 eq www any` //表示允许任何外部主机对全局地址 192.168.0.8 的这台主机进行 `http` 访问。其中使用 `eq` 和一个端口来允许或拒绝对这个端口的访问。`Eq ftp` 就是指允许或拒绝只对 `ftp` 的访问。

`Pix525(config)#conduit deny tcp any eq ftp host 61.144.51.89` //设置不允许外部主机 61.144.51.89 对任何全局地址进行 `ftp` 访问。

`Pix525(config)#conduit permit icmp any any` //设置允许 `icmp` 消息向内部和外部通过。

`Pix525(config)#static (inside, outside) 61.144.51.62 192.168.0.3`

`Pix525(config)#conduit permit tcp host 61.144.51.62 eq www any`

这两句是将 `static` 和 `conduit` 语句结合而生效的, 192.168.0.3 在内网是一台 `web` 服务器, 现在希望外网的用户能够通过 `pix` 防火墙得到 `web` 服务。所以先做 `static` 静态映射把内部 IP192.168.0.3 转换为全局 IP61.144.51.62, 然后利用 `conduit` 命令允许任何外部主机对全局地址 61.144.51.62 进行 `http` 访问。

小提示: 对于上面的情况不使用 `conduit` 语句设置容许访问规则是不可以的, 因为默认情况下 `PIX` 不容许数据包主动从低安全级别的端口流向高安全级别的端口。

### 3、配置 fixup 协议:

`fixup` 命令作用是启用, 禁止, 改变一个服务或协议通过 `pix` 防火墙, 由 `fixup` 命令指定的端口是 `pix` 防火墙要侦听的服务。示例例子如下:

`Pix525(config)#fixup protocol ftp 21` //启用 `ftp` 协议, 并指定 `ftp` 的端口号为 21

`Pix525(config)#fixup protocol http 80`

`Pix525(config)#fixup protocol http 1080` //为 `http` 协议指定 80 和 1080 两个端口。

`Pix525(config)#no fixup protocol smtp 80` //禁用 `smtp` 协议。

### 4、设置 telnet:

在 `pix5.0` 之前只能从内部网络上的主机通过 `telnet` 访问 `pix`。在 `pix 5.0` 及后续版本中, 可以在所有的接口上启用 `telnet` 到 `pix` 的访问。当从外部接口要 `telnet` 到 `pix` 防火墙时, `telnet` 数据流需要用 `ipsec` 提供保护, 也就是说用户必须配置 `pix` 来建立一条到另外一台 `pix`, 路由器或 `vpn` 客户端的 `ipsec` 隧道。另外就是在 `PIX` 上配置 `SSH`, 然后用 `SSH client` 从外部 `telnet` 到 `PIX` 防火墙。

我们可以使用 `telnet` 语句管理登录 `PIX` 的权限。

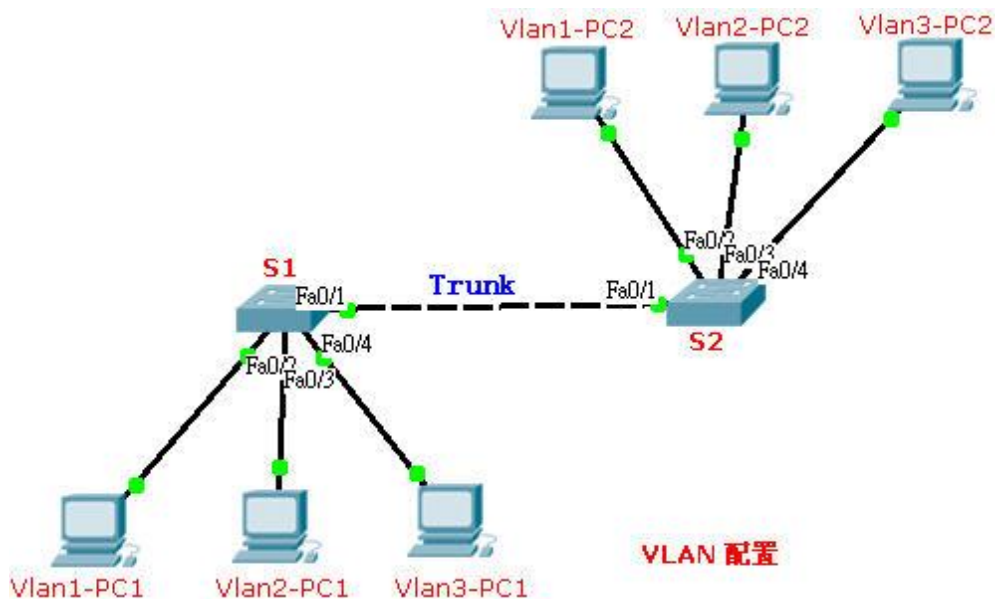
**telnet 配置语法:** `telnet local_ip [netmask]`

`local_ip` 表示被授权通过 `telnet` 访问到 `pix` 的 ip 地址。如果不设此项, `pix` 的配置方式只能由 `console` 进行。也就是说默认情况下只有通过 `console` 口才能配置 `PIX` 防火墙。

小提示: 由于管理 `PIX` 具有一定的危险性, 需要的安全级别非常高, 所以不建议大家开放提供外网 IP 的 `telnet` 管理 `PIX` 的功能。如果实际情况一定要通过外网 IP 管理 `PIX` 则使用 `SSH` 加密手段来完成。

## 03-Vlan 配置

### 一、虚拟局域网 VLAN 配置



#### <1>配置 VTP 服务器

```
S1#vlan database //进入 vlan 配置模式
S1(vlan)#vtp server //设置该交换机为 VTP 服务器模式，考试时选默认是透明模式，但思科是默认为服务器模式。
S1(vlan)#vtp domain domain-name //设置 VTP 管理域名称
S1(vlan)#vtp pruning //启用 VTP 修剪功能
S1(vlan)#exit
S1#show vtp status //显示 VTP 状态信息
```

#### <2>配置 VTP 客户端

```
S2#vlan database
S2(vlan)#vtp client //设置该交换机为 VTP 客户端模式
S2(vlan)#vtp domain domain-name //这里域名必须和上面的 vtp 服务器设置的 domain 名称一样
S2(vlan)#exit
```

#### <3>配置两个交换机之间的 Trunk 端口。<下面命令要分别在 S1\S2 交换机上执行>

```
S(config)#interface f0/1
S(config-if)#switchport mode trunk //配置该端口为 Trunk 中继模式。
S(config-if)#switchport trunk allowed vlan all //设置允许从该接口交换数据的 vlan
S(config-if)#^Z
```

#### <4>在 VTP 服务器上创建 VLAN

```
S1#vlan database
S1(vlan)#vlan 2 name vlan2 //创建一个 Vlan 2 命名为 vlan2
S1(vlan)#vlan 3 name //创建一个 Vlan 3，系统会自动命名为 Vlan003
S1(vlan)#exit
```

#### <4>分配交换机端口到各个 Vlan <下面命令要分别在 S1\S2 交换机上执行>

```
S(config)#interface f0/3 <如果交换机上有多个端口要分到同一个 VLAN 中，可用 interface range f0/3 - 12,15>
S(config-if)#switchport mode access //设置端口为静态 VLAN 访问模式
S(config-if)#switchport access allowed vlan 2 //设置允许从该接口交换数据的 vlan
S(config-if)#^Z
```

<这是将 F0/3 端口分配到 VLAN2 中，另外，用同样的方法把 F0/4 分配到 vlan3 中>

## 二、交换机基本配置---mac 地址配置

```
Switch(config)#mac-address-table aging-time 100 //设置学习的 MAC 地址的超时时间，默认 300s
Switch(config)#mac-address-table permanent 0050.8DCB.FBD1 f0/3 //添加永久地址
Switch(config)#mac-address-table restricted static 0050.8DCB.FBD2 f0/6 f0/7 //加入限制性静态地址，它是在设置永久性地址的基础上，同时限制了源端口，它只允许所 static 配置的接口（f0/6）与指定的端口（f0/7）通信，提高安全性。
Switch(config)#end
Switch#show mac-address-table （查看 MAC 地址表）
Switch#clear mac-address-table dynamic （清除动态学习的 MAC 地址表项）
Switch#clear mac-address-table restricted static （清除配置的限制性 MAC 地址表项）
```

### 三、生成树快速端口(PortFast)配置

STP PortFast 是一个 Catalyst 的一个特性。在 STP 中，只有 forwarding 状态，port 才能发送用户数据。如果一个 port 一开始是没有接 pc，一旦 pc 接上，就会经历 blocking(20s)->listening(15s)->learning(15s)->forwarding 状态的变化。这样从 pc 接上网线，到能发送用户数据，缺省的配置下需要等 50 秒的时间，但如果设置了 portfast，就使得该端口不再应用 STP 算法，一旦该端口物理上能工作，就立即将其置为“转发”状态。在基于 IOS 交换机上，PortFast 只能用于连接到终端设备的接入层交换机端口上。

开启 PortFast 命令：

```
Switch(config)#interface f0/1
Switch(config-if)#spanning-tree portfast
```

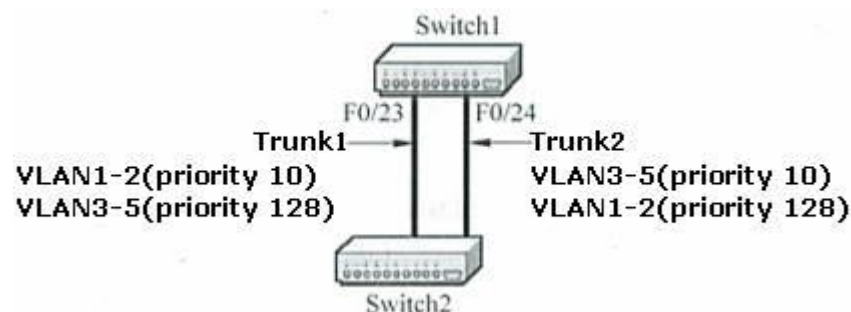
如果把批次开启可以用：

```
Switch(config)#interface range f0/1 - 3
Switch(config-if-range)#spanning-tree portfast //注：只有在确认不会产生环路的端口上开启快速端口。
```

### 四、STP 的负载均衡配置

#### 1、使用 STP 端口权值（优先级）实现负载均衡

当交换机的两个口形成环路时，STP 端口优先级用来决定那个口是转发状态，那个处于阻塞的。可以通过修改 Vlan 对应端口的优先级来决定该 VLAN 的流量走两对 Trunk 链路中那一条。



如上图，我们用端口 F0/23 做 Trunk1，用 f0/24 做 Trunk2。具体配置如下：

#### <1>、配置 VTP、VLAN 及 Trunk（和上面 VLAN 配置过程一样，我们把 S1 设成服务器模式，S2 设为客户端模式）

（配置 vtp----在 S1、S2 上）

```
S1#vlan database //进入 VLAN 配置子模式
```

```
S1(vlan)#vtp server
```

```
S1(vlan)#vtp domain vtpserver //这三步也要在 S2 上执行，只是把第二步的 Vtp server 换成 vtp client
```

（配置 Trunk----在 S1、S2 上）

```
S1(config)#interface f0/23
```

```
S1(config-if)#switchport mode trunk
```

```
S1(config-if)#exit
```

```
S1(config)#interface f0/24
```

```
S1(config-if)#switchport mode trunk //在 S2 上执行同样的这几步操作。
```



（配置 VLAN----只在 S1 上）

```
S1#vlan database
S1(vlan)#vlan 2 name vlan2
S1(vlan)#vlan 3 name vlan3
...      //依次创建 2-5 vlan。
S1(vlan)#exit
```

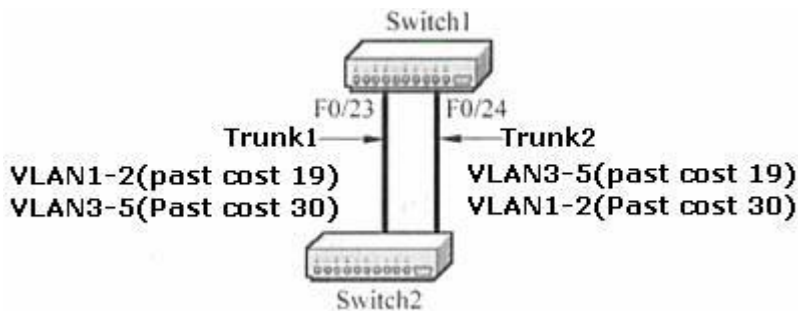
### <2>、配置 STP 优先级----在 vtp 服务器 S1 上配置

```
S1(config)#interface f0/23      //进入 f0/23 端口配置模式，Trunk1
S1(config-if)#spanning-tree vlan 1 port-priority 10  //将 vlan 1 的端口优先级设为 10（值越小，优先级越高！）
S1(config-if)#spanning-tree vlan 2 port-priority 10  //将 vlan 2 的设为 10，vlan3-5 在该端口上是默认的 128
S1(config-if)#exit
S1(config)#interface f0/24      //进入 f0/24，Trunk2
S1(config-if)#spanning-tree vlan 3 port-priority 10
S1(config-if)#spanning-tree vlan 4 port-priority 10
S1(config-if)#spanning-tree vlan 5 port-priority 10  //同上，将 vlan3-5 的端口优先级设为 10
```

由于我们分别设置了不同 Trunk 上不同 VLAN 的优先级。而默认是 128，这样，STP 协议就可以根据这个优先级的大小来使 Trunk1 发送接收 vlan1-2 的数据；Trunk2 发接 vlan3-5 的数据，从而实现负载均衡。

### 2、使用 STP 路径值实现负载均衡

如图示：Trunk1 走 VLAN1-2 的数据，而 Trunk2 走 VLAN3-5 的数据。



其中 vtp、vlan、和 trunk 端口的配置都和上面一样，不再列出。各项都配置好后，在服务器模式的交换机 S1 上执行路径值的配置（路径值也叫端口开销，IEEE802.1d 规定默认值：10Gbps=2；1Gbps=4；100Mbps=19；10Mbps=100）

```
S1(config)#interface f0/23
S1(config-if)#spanning-tree vlan 3 cost 30
S1(config-if)#spanning-tree vlan 4 cost 30
S1(config-if)#spanning-tree vlan 5 cost 30  //分别设置 vlan 3-5 生成树路径值为 30
S1(config-if)#exit
S1(config)#interface f0/24
S1(config-if)#spanning-tree vlan 1 cost 30
S1(config-if)#spanning-tree vlan 2 cost 30
```

这样，通过将希望阻断的 VLAN 的生成树路径设大，stp 协议就会阻断该 VLAN 从该 Trunk 上通过。

## 五、EtherChannel

EtherChannel 有两个版本，Cisco 的称为端口聚合协议 PAgP(Port Aggregation Protocol)，IEEE 的 802.3ad 标准称为链路汇聚控制协议 LACP(Link Aggregation Control Protocol)，他们的配置有些不同。

为避免有冗余链路的网络环境里出现环路，我们采用 STP 技术，但 Spanning Tree 冗余连接的工作方式是：除了一条链路工作外，其余链路实际上是处于待机(Stand By)状态。那么能不能把这些冗余的链路利用起来以增加带宽呢？又如何让两条或多条链路同时工作呢？这就是在网络工程中常用的 EtherChannel 技术。Etherchannel 特性在交换机到交换机、交换机到路由器、交换机到服

务器之间提供冗余的、高速的连接方式，简单说就是将两个设备间多条 FE 或 GE 物理链路捆在一起组成一条设备间逻辑链路，从而达到增加带宽，提供冗余的目的。该技术容错能力好，实体线路中断可以在数秒内切换至别条线路使用。

以太 channel 在交换机间或者交换机和主机间提供最多 800Mbps(fast etherchannel)或者最多 8Gbps(Gigabit etherchannel)的全双工带宽。一个以太 channel 最多由八个配置正确的端口构成。所有在同一个以太 channel 中的接口必须具有相同的特性(如双工模式、速度、同为 FE 或 GE 端口、native VLAN、VLAN range、and trunking status and type.等)，并且都要同时配置成二层接口或者三层接口。

设定范例：



```
S1(config)#interface range f0/1 - 2
```

```
S1(config-if-range)#channel-group 1 mode passive
```

//表示将 Fa0/1,Fa0/2 设成同一个 group，使用 LACP 的被动模式！

```
S2(config)#interface range f0/1 - 2
```

```
S2(config-if-range)#channel-group 1 mode active
```

//将另一边的端口也以同样方式设定，但 mode 设成 active 即可，交换机会自己新增一个虚拟端口：Port-channel1(Po1)，它和实体接口一样使用，其成本为 12。

检查其状态，可用命令：

```
show etherchannel detail;
```

```
show etherchannel load-balance;
```

```
show etherchannel port
```

```
show etherchannel port-channel;
```

```
show etherchannel protocol;
```

```
show etherchannel summary
```

## 04-ACL 配置

### ACL 命令和过滤规则

访问控制列表(ACL)是应用在路由器接口上的指令列表。这些指令列表用来告诉路由器哪些数据包可以收、哪些数据包需要拒绝。至于数据包是被接收还是拒绝，可以由类似于源地址、目的地址、端口号等的特定指示条件来判断决定。

将数据包和访问列表进行比较时应遵循的重要规则：

1. 数据包到来，则按顺序比较访问列表的每一行。
2. 按顺序比较访问列表的各行，直到找到匹配的一行，一旦数据包和某行匹配，执行该行规则，不再进行后续比较。
3. 最后一行隐含“deny”的意义。如果数据包与访问列表中的所有行都不匹配，将被丢弃。
4. IP 访问控制列表会发送一个 ICMP 主机不可达的消息到数据包的发送者，然后丢弃数据包。
5. 如果某个列表挂接在实际接口上，删除列表后，默认的 deny any 规则会阻断那个接口的所有数据流量。

有两种类型的 ACL：标准的访问列表和扩展的访问列表

**标准访问控制列表**控制基于过滤源地址的信息流。编号范围 1-99，举例：

```
router(config)#access-list 10 permit 172.16.0.0 0.0.255.255
```

```
router(config)#access-list 20 deny 192.168.1.0 0.0.0.255
```

```
router(config)#access-list 20 permit any //注意这个编号 20 的 ACL 的顺序，如果调换顺序，就不起任何作用。
```

```
router(config)#access-list 30 deny host 192.168.1.1 //host 表精确匹配，默认掩码为 0.0.0.0，指定单个主机。
```

注意：在访问列表的最后默认定义了一条 deny any any 语句。



---

**扩展访问控制列表**比标准访问控制列表具有更多的匹配项，包括协议类型、源地址、目的地址、源端口、目的端口、IP 优先级等。编号范围是从 100 到 199，格式一般为：

access-list ACL 号 [permit|deny] [协议] [定义过滤源主机范围] [定义过滤源端口] [定义过滤目的主机访问] [定义过滤目的端口]

```
access-list 100 permit ip 172.16.0.0 0.0.255.255 192.168.1.0 0.0.0.255
```

access-list 101 deny tcp any host 192.168.1.1 eq www //这句命令是将所有主机访问 192.168.1.1 上网页服务（WWW）TCP 连接的数据包丢弃。

access-list 101 permit tcp any host 198.78.46.8 eq smtp //允许来自任何主机的 TCP 报文到达特定主机 198.78.46.8 的 smtp 服务端口(25)

注意：这里 eq 就是等于的意思。端口号的指定可以用几种不同的方法。可以用数字或一个可识别的助记符。可以使用 80 或 http 或 www 来指定 Web 的超文本传输协议。对于使用数字的端口号，还可以用 "<"、">"、"=" 以及不等于来进行设置。

**命名访问列表**只是创建标准或扩展访问列表的另一种方法而已。所谓命名是以列表名代替列表编号来定义 IP 访问控制列表的。举例如下：

```
Router(config)#ip access-list extended http-not
```

```
Router(config-ext-nacl)#deny tcp 172.16.10.0 0.0.0.255 host 172.16.1.2 eq 23
```

```
Router(config-ext-nacl)# permit ip any any
```

```
Router(config-ext-nacl)#exit
```

**入口访问列表和出口访问列表：**通过上面方法创建的访问列表，要应用到路由器的一个要进行过滤的接口上，并且指定将它应用到哪一个方向的流量上时，才真正的被激活而起作用。

访问列表的调用：在接口下使用：

```
router(config)#interface s0/1
```

```
router(config-if)# ip access-group 10 in
```

```
router(config-if)# ip access-group 20 out
```

ACL 可在 VTY 下调用，但只能用标准列表：

```
R1(config)#access-list 10 permit 192.168.1.1
```

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

```
R1(config-line)#access-class 10 in
```

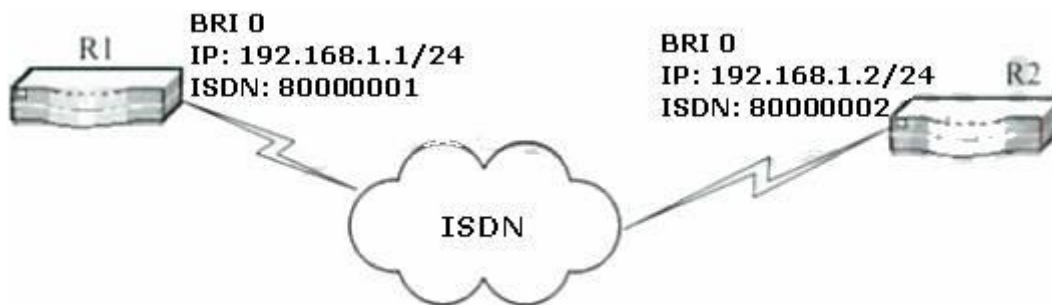
下面是通用的 ACL 配置规则：

1. 每个接口、每个协议或每个方向上只可以应用一个访问列表。（因为 ACL 末尾都隐含拒绝的语句，经过第一个 ACL 的过滤，不符合的包都被丢弃，也就不会留下任何包和第二个 ACL 比较）。
2. 除非在 ACL 末尾有 permit any 命令，否则所有和列表条件不符的包都将丢弃，所以每个 ACL 至少要有一个运行语句，以免其拒绝所有流量。
3. 要先创建 ACL，再将其应用到一个接口上，才会生效。
4. ACL 过滤通过路由器的流量，但不过滤该路由器产生的流量。
5. IP 标准访问列表尽可能的应用在靠近目的地的接口上，因为他是基于源地址过滤的，放在源端没有意义。
6. IP 扩展访问列表尽可能的应用在靠近源地址的接口上，因为它可以基于目的地址、协议等过滤，放在源端过滤，免得需要过滤的数据包还被路由到目的端才被过滤，以节省带宽。

## 05-广域网配置

### 一、ISDN 配置

ISDN(综合业务数字网)，提供两种类型的访问接口，即基本速率接口 BRI 和主要速率接口 PRI。ISDN BRI 提供 2B<sub>64</sub>+D<sub>16</sub> PRI 提供 30B+D(均为 64Kb/s)。下面通过一个实例来说明两台路由器通过 ISDN 线路连接时最基本的配置。



## R1 配置

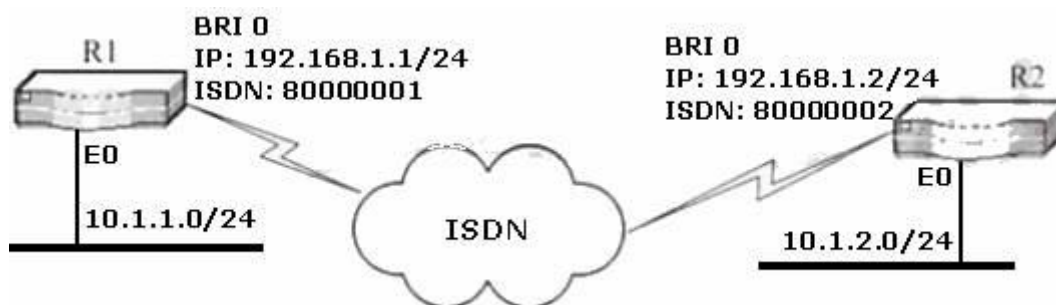
```
R1(config)#isdn switch-type basic-net3 //设置交换机类型为 basic-net3，这取决于所连接的 ISDN 类型，中国用的这个
R1(config)#interface bri 0 //进入 BRI 接口配置模式
R1(config-if)#ip address 192.168.1.1 255.255.255.0 //设置接口 IP 地址
R1(config-if)#encapsulation ppp //设置封装协议为 PPP
R1(config-if)#dialer string 80000002 //设置拨号串，为对方的 ISDN 号
R1(config-if)#dialer-group 1 //在该接口应用拨号列表 1 的设置
R1(config-if)#no shutdown //激活端口
R1(config-if)#exit
R1(config)#dialer-list 1 protocol ip permit //设置拨号列表 1，即当 IP 包需要在此拨号链路上传输时引起拨号
```

## R2 配置

```
R2(config)#isdn switch-type basic-net3
R2(config)#interface bri 0
R2(config-if)#ip address 192.168.1.2 255.255.255.0 //设置接口 IP 地址
R2(config-if)#encapsulation ppp
R2(config-if)#dialer string 80000001 //设置拨号串，为对方(R1)的 ISDN 号
R2(config-if)#dialer-group 1
R2(config-if)#no shutdown
R2(config-if)#exit
R2(config)#dialer-list 1 protocol ip permit //R2 和 R1 配置基本相同
```

## 二、PPP 配置

PPP(点对点协议)，DDR 是按需拨号路由，一般在实际应用中，ISDN、PPP、DDR 经常综合应用  
下面结合实例说明基本的配置命令：



## R1

```
R1(config)#username R2 password 0 cisco //定义用户名和口令，注意用户名设为对方，口令双方设置一致。
由于是在 ISDN 线路进行 PPP 封装，对 ISDN 和端口的配置都和上面 ISDN 的配置相同，下面只列出配置清单。
```

```
isdn switch-type basic-net3
Interface E0
```

---

```
ip address 10.1.1.1 255.25.255.0
```

```
Interface BRI0
```

```
ip address 192.168.1.1 255.255.255.0
```

```
encapsulation ppp
```

```
dialer idle-timeout 300
```

//设置拨号空闲时间，如果超时没有 IP 数据包从该接口发送，就断开连接。默认 120s

```
dialer map ip 192.168.1.2 name R2 broadcast 80000002 （注：设置的 IP、name、isdn 号都是对方的）
```

//设置拨号映射，当有列表定义的数据包传输时就使用这个定义的映射发起拨号连接并进行认证操作。

```
ppp multilink //启用 PPP 多链路的功能特性
```

```
dialer load-threshold 128
```

//设置启用多链路的条件，即：当实际负载占一个 B 信道带宽的  $(128/256)\% = 50\%$  时，就启用第二个 B 信道，设为 1 时为无条件启用 2 个 B 信道。

```
dialer-group 1
```

```
no cdp enable //通常在拨号链路上都禁用 CDP 发现协议
```

```
ppp authentication chap //设置 PPP 认证方式为 CHAP，或设为：pap（口令认证方式）。或设为：chap pap(混合模式)。
```

```
ip route 10.1.2.0 255.255.255.0 192.168.1.2
```

```
dialer-list 1 protocol ip permit
```

R2 配置基本和 R1 相同，只有下面几个地方需注意配置正确的参数：

```
username R1 password 0 cisco
```

```
ip address 10.1.2.1 255.25.255.0
```

```
ip address 192.168.1.2 255.255.255.0
```

```
dialer map ip 192.168.1.1 name R1 broadcast 80000001
```

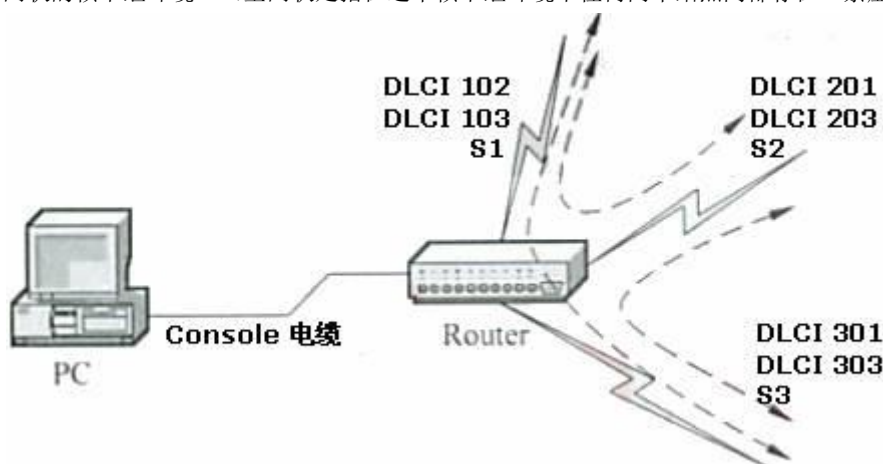
```
ip route 10.1.1.0 255.255.255.0 192.168.1.1
```

### 三、 帧中继配置

帧中继是 X.25 的简化版本。帧中继广域网设备分为 DTE 和 DCE，路由作为 DTE 设备。帧中继提供面向连接的数据链路层通信。通过帧中继虚电路为每个链路分配一个链路识别码（DLCI）。

#### 1、 配置帧中继交换机：

就是配置一个帧中继的环境，为下面的基本帧中继配置做准备，现以一个具有 3 个串行接口的路由器为例，通过下面配置来实现全网状的帧中继环境。（全网状是指在这个帧中继环境中任何两个结点间都存在一条虚电路）



每个接口上的 DLCI 都标在图上，虚线箭头表示两结点间的虚电路。下面是配置清单：

#### Interface serial1

```
no ip address
```

```
encapsulation frame-relay
```

```
clockrate 64000
frame-relay lmi-type cisco
frame-relay lmi-type dce
frame-relay route 102 interface serial2 201
frame-relay route 103 interface serial3 301
!
```

### Interface serial2

```
no ip address
encapsulation frame-relay
clockrate 64000
frame-relay lmi-type cisco
frame-relay lmi-type dce
frame-relay route 201 interface serial1 102
frame-relay route 203 interface serial3 303
!
```

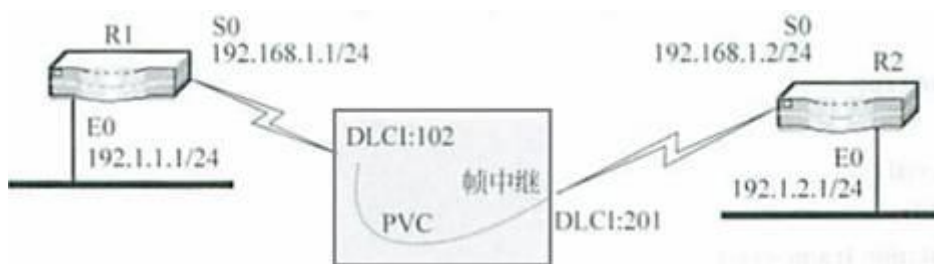
### Interface serial3

```
no ip address
encapsulation frame-relay
clockrate 64000
frame-relay lmi-type cisco
frame-relay lmi-type dce
frame-relay route 301 interface serial1 103
frame-relay route 303 interface serial2 203
```

配置完成，用 `show frame route` 显示完整路由信息

## 2、基本帧中继配置

上面的帧中继环境已经配置好，现在在该环境中接入两个路由器，以实现端到端的连通性。



### 一、配置基本的帧中继连接

这里忽略 E0 口的 IP 配置和 `no shutdown` 激活配置

#### 路由器 R1

```
R1(config)#interface s0
R1(config-if)#ip address 19.168.1.1 255.255.255.0
R1(config-if)#encap frame-relay //该接口使用帧中继封装
R1(config-if)#no frame-relay inverse-arp //关闭帧中继逆向 ARP
R1(config-if)#frame map ip 192.168.1.2 cisco
R1(config-if)#no shutdown
R1(config-if)#end
```

#### 路由器 R2

---

```
R2(config)#interface s0
R2(config-if)#ip address 19.168.1.2 255.255.255.0
R2(config-if)#encap frame-relay //该接口使用帧中继封装
R2(config-if)#no frame-relay inverse-arp //关闭帧中继逆向 ARP，防止多个 DLCI 之间的映射产生混乱。
R2(config-if)#frame map ip 192.168.1.1 cisco
R2(config-if)#no shutdown
R2(config-if)#end
```

二、配置静态路由并测试连通性。静态路由配置方法上面已经学习，不再详述。然后可以使用下列命令查看配置状态信息。

Show frame pvc; Show frame map; Show frame traffic; Show frame lmi

#### 四、 L2TP 配置与测试

第二层通道协议（Layer 2 Tunneling Protocol）是一广泛使用的隧道技术，应用 L2TP，方便远程用户随时通过 Internet 安全的接入公司局域网。L2TP 有两种报文：

控制报文：建立、维护和释放隧道。

数据报文：包装通过隧道传输的 PPP 帧。L2TP 配置实例：

Vpdn-group 1 //创建 vpdn 组 1，并记入 VPDN 组 1 配置模式。

Accept-dialin protocol l2tp virtual-template 1 terminate-from hostname as8010

//接受 L2TP 通道连接请求，并根据 Virtual-template 1 创建 Virtual-access 接口。

Local name keith //设置 Tunnel 本端名称为 Keith

Lcp renegotiation always //LCP 再次协商

No l2tp tunnel authentication //设置不验证通道对端。

## 06-NAT 配置

在配置网络地址转换之前，首先必须搞清楚内部接口和外部接口，以及在哪个外部接口上启用 NAT。通常情况下，连接到用户内部网络的接口是 NAT 内部接口，而连接到外部网络(如 Internet)的接口是 NAT 外部接口。

### 1). 静态地址转换的实现

假设内部局域网使用的 IP 段为 192.168.0.1~192.168.0.254，路由器局域网端口(即默认网关)的 IP 地址为 192.168.0.1，子网掩码为 255.255.255.0。网络分配的合法 IP 地址范围为 61.159.62.128~61.159.62.135，路由器在广域网中的 IP 地址为 61.159.62.129，子网掩码为 255.255.255.248。要求将内网 192.168.0.2 和~.3 分别转换为公网 IP 地址 61.159.62.130 和~131。

第一步，设置外部端口。

```
R(config)#interface serial 0
R(config-if)#ip address 61.159.62.129 255.255.255.248
R(config-if)#ip nat outside
```

第二步，设置内部端口。

```
R(config)#interface ethernet 0
R(config-if)#ip address 192.168.0.1 255.255.255.0
R(config-if)#ip nat inside
```

第三步，在内部本地与内部合法地址之间建立静态地址转换。

格式：ip nat inside source static 内部本地地址 内部全局地址

```
R(config)#ip nat inside source static 192.168.0.2 61.159.62.130
```

```
R(config)#ip nat inside source static 192.168.0.3 61.159.62.131
```

至此，静态地址转换配置完毕。

---

## 2). 动态地址转换的实现

假设内网使用 IP 段为 172.16.100.1~172.16.100.254,路由器局域网端口（默认网关）的 IP 为 172.16.100.1,子网掩码为 255.255.255.0。ISP 分配的合法 IP 地址范围为 61.159.62.128~61.159.62.191, 路由器在外网端的 IP 地址设为 61.159.62.129,子网掩码为 255.255.255.192。要求将内网 172.16.100.1~172.16.100.254 动态转换为外网 IP: 61.159.62.130~61.159.62.190。

第一步, 设置外部端口, 语法: ip nat outside

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

```
R(config-if)#ip address 61.159.62.129 255.255.255.248
```

```
R(config-if)#ip nat outside //将串行口 serial 0 设置为外网端口。注意, 可以定义多个外部端口。
```

第二步, 设置内部端口。语法: ip nat inside

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

```
R(config-if)#ip address 172.16.100.1 255.255.255.0
```

```
R(config-if)#ip nat inside //将 Ethernet 0 设置为内网端口。注意, 可以定义多个内部端口。
```

第三步, 定义内部全局合法 IP 地址池, 语法: ip nat pool 地址池名称 起始 IP 终止 IP 子网掩码

```
R(config)#ip nat pool net 61.159.62.130 61.159.62.190 netmask 255.255.255.192
```

需要注意的是, 即使掩码为 255.255.255.0, 也会由起始 IP 地址和终止 IP 地址对 IP 地址池进行限制。如果有多个合法 IP 地址范围, 可以分别添加。

第四步, 定义内部网络中允许访问 Internet 的访问列表, 语法: access-list 标号 permit 源地址 通配符 (标号 1~99)

```
R(config)#access-list 1 permit 172.16.100.0 0.0.0.255
```

//允许访问 Internet 的网段 172.16.100.0~172.16.100.255, 主机掩码为 0.0.0.255。需要注意的是, 在这里采用的是主机掩码, 而非子网掩码。子网掩码与主机掩码的关系为: 主机掩码+子网掩码=255.255.255.255。

另外, 如果想将多个 IP 地址段转换为合法 IP 地址, 可以添加多个访问列表。

第五步, 实现网络地址转换, 语法: ip nat inside source list 访问列表标号 pool 内部全局地址池名字

```
R(config)#ip nat inside source list 1 pool net
```

如果有多个内部访问列表, 可以一一添加, 以实现网络地址转换, 如

```
R(config)#ip nat insde source list 2 pool chinanet
```

如果有多个地址池, 也可以一一添加, 以增加合法地址池范围, 如

```
R(config)#ip nat insde source list 2 pool cernet
```

至此, 动态地址转换设置完毕。

## 3). 端口复用动态地址转换(PAT)

内部网络使用的 IP 地址段为 10.100.100.1~10.100.100.254,路由器局域网端口（默认网关）的 IP 地址为 10.100.100.1, 子网掩码为/24。网络分配的合法 IP 地址范围为 202.99.160.0~202.99.160.3,路由器在广域网端口的 IP 为 202.99.160.1,子网掩码为 255.255.255.252, 可用于转换的 IP 地址为 202.99.160.2。要求将内部网址 10.100.100.1~10.100.100.254 转换为合法 IP 地址 202.99.160.2。

第一步, 设置外部端口。

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

```
R(config-if)#ip address 202.99.160.1 255.255.255.252
```

```
R(config-if)#ip nat outside
```

第二步, 设置内部端口。

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

```
R(config-if)#ip address 10.100.100.1 255.255.255.0
```

```
R(config-if)#ip nat inside
```

第三步, 定义合法 IP 地址池。

```
R(config)#ip nat pool onlyone 202.99.160.2 202.99.160.2 netmask 255.255.255.252
```



// 指明地址缓冲池的名称为 **onlyone**,IP 地址范围为 202.99.160.2,子网掩码为 255.255.255.252。由于本例只有一个 IP 地址可用,所以起始 IP 与终止 IP 相同。如果有多个 IP 地址,则应当分别键入起止的 IP。

第四步,定义内部访问列表。

```
R(config)#access-list 1 permit 10.100.100.0 0.0.0.255
```

第五步,设置复用动态地址转换,语法: ip nat inside source list 访问列表号 pool 内部合法地址池名字 overload

```
R(config)#ip nat inside source list 1 pool onlyone overload //注意:overload 是复用动态地址转换的关键词
```

至此,端口复用动态地址转换完成。

## 1. NAT 验证与故障排除

Router#show ip nat translations 查看 nat 配置信息

Router#debug ip nat 输出 nat 运行的过程

Router#show ip nat statistics 显示 NAT 配置的汇总情况

## 07-路由器配置

### 一、静态路由的配置和验证 (III)

静态路由配置很简单:

静态路由器配置: ip route 172.16.1.0 255.255.255.0 192.168.2.4 150 permanent

目标网络      子网掩码      下一跳路由器入口地址,或(s0/0)本地路由器退出接口      管理距离AD,默认是1      强制保留路由项

静态缺省路由配置:

```
R(config)# ip route 0.0.0.0 0.0.0.0 192.168.2.1 (or s0/0)
```

```
R(config)# ip classless //在路由表查不到的网络地址都发往下一跳 192.168.2.1 或从本地路由器的 s0/0 出去
```

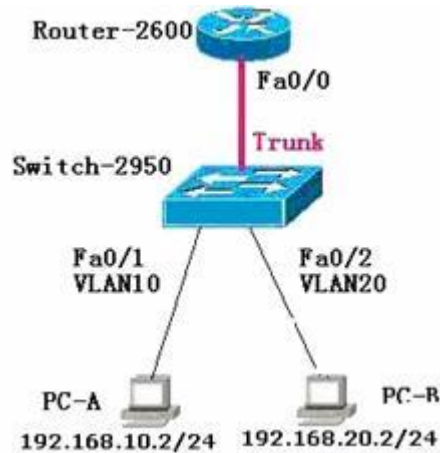
默认路由配置:

```
R(config)# ip default-network 192.168.2.0 //和上面缺省路由区别是,路由协议会传播这条路由信息
```

### 二、单臂路由器的配置 (III)

在学习单臂路由的配置之前,起码要知道 VLAN 是怎么配置的,这样学单臂路由就轻松多了。先来了解一下什么是单臂路由,为什么要用到单臂路由。VLAN (虚拟局域网) 技术是路由交换中非常基础的技术。在网络管理实践中,通过在交换机上划分适当数目的 vlan,不仅能有效隔离广播风暴,还能提高网络安全系数及网络带宽的利用效率。划分 vlan 之后, vlan 与 vlan 之间是不能通信的,只能通过路由或三层交换来实现。我们知道路由器实现路由功能通常是数据报从一个接口进来然后另一个接口出来,现在路由器与交换机之间通过一条主干现实通信或数据转发,也就是说路由器仅用一个接口实现数据的进与出,因为我们形象地称它为单臂路由。单臂路由是解决 vlan 间通信的一种廉价而实用的解决方案。

下面请看图，PC-A 和 PC-B 分别属于 vlan10 和 vlan20，Switch2950 是一个 cisco 的二层交换机，欲实现 vlan10 和 vlan20 的通信，我们要增加一个路由器来转发 vlan 之间的数据包，路由器与交换机之间使用单条链路相连（图中画红线），这条链路也叫主干，所有数据包的进出都要通过路由器 2600 的 f0/0 端口来现实数据转发。



接下来，结合以上网络拓扑探讨一下单臂路由的配置。图中路由器是 cisco 的 2600 系列，交换机采用 cisco 的 2950。

配置交换机

```
Switch#vlan database
Switch(vlan)#vlan 10 name caiwu
Switch(vlan)#vlan 20 name renshi
Switch(vlan)#exit
Switch#interface f0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access allowed vlan 10 //同样的方法，将 f0/2 接口分配给 VLAN 20
...
Switch#interface f0/12
Switch(config-if)#switchport mode trunk
```

配置路由器

```
Router(config)#interface f0/0
Router(config-if)#no shutdown
Router(config-if)#interface f0/0.1 //配置子接口，注意后面的.1，这是配置单臂路由的关键步骤。
Router(config-subif)#ip address 192.168.10.1 255.255.255.0
Router(config-subif)#encapsulation dot1q 10 //为该接口配置 802.1Q 协议，关键步骤
Router(config-subif)#exit
Router(config)#interface f0/0.2 //配置第二个子接口
Router(config-subif)#ip address 192.168.20.1 255.255.255.0
Router(config-subif)#encapsulation dot1q 10
Router(config-subif)#end
```

最后配置 PC-A 和 PC-B 的 IP 地址和子网掩码，PC-A 主机 ping PC-B 主机。实验结果能 Ping 通，配置成功。

### 三、RIP 协议的配置（III）

RIP 配置：

```
R(config)# router rip
R(config)# network 10.0.0.0 （一个有类网络号，要声明所有连接到该路由器接口上的有类网络）
```

RIPv2 配置：

```
R(config)# version 2
```

RIP 抑制（被动接口）：

R(config)# passive-interface s0/0(表示 s0/0 接口只接受 RIP 路由更新而不再发送更新)

#### 四、终端服务器的配置（III）

可以通过配置终端服务器（Terminal server）来实现用一台 PC 机同时访问多个网络设备（如路由器或交换机），减少配置管理的负担。终端服务器是路由器的一种功能，现在大多路由器都可以通过加装异步/同步网络模块用作终端服务器。



如上图：PC 机与终端服务器之间通过 Console 线缆直接相连；终端服务器通过 1 拖 8 与两台路由器相连，线号为 1 和 2。

终端服务器的配置清单如下：

```
hostname Term_Server
!
interface loopback0      //增加一个 loopback 接口，用于逆向 Telnet
ip address 10.1.1.1 255.255.255.255 //最节省地址空间的方法设置一个 ip 给环回接口。
!
ip host router1 2001 10.1.1.1 //配置主机表，使得 router1 与 2001，router2 与 2002 联系起来，方便访问其他路由器
ip host router2 2002 10.1.1.1 //我们访问它们时就不用 telnet 10.1.1.1 2001，而直接用主机名：router1
!
line 1 8
no exec //在 line1-8 上配置 no exec 禁止这 8 条异步线路上产生 EXEC 进程，而只允许从终端服务器到其他路由器的连接。
transport input all //指明在这 8 条线路的输入方向上允许所有的协议。
```

配置好后，首先登录到终端服务器：

```
Term_Server#
Term_Server#router1
Trying router1 (10.1.1.1,2001) ...Open
```

Router1>

这时可以用会话切换命令 Ctrl+Shift+6 然后按 X，切换回终端服务器：

```
Term_Server#
Term_Server#router2
Trying router1 (10.1.1.1,2002) ...Open
```

Router2>

< Ctrl+Shift+6, x>

```
Term_Server#show sessions
```

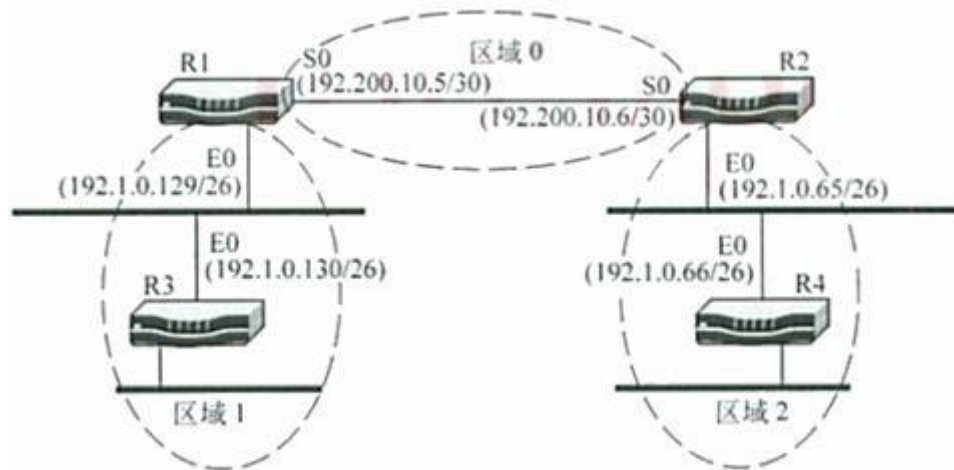
Conn	Host	Address	Byte	Idle	Conn Name
1	router1	10.1.1.1	0	0	router1
2	router2	10.1.1.1	0	0	router2

```
Term_Server#disconnect 2 //断开会话 2
```

```
Term_Server#show line 1 //查看线路 1 状态
```

```
Term_Server#clear line 2 //清除线路 2
```

#### 六、单区域/多区域 OSPF 协议的配置（III）



路由器 R1

```
Interface e0
 ip address 192.1.0.129 255.255.255.192
Interface s0
 ip address 192.200.10.5 255.255.255.252
router ospf 100
 network 192.200.10.1 0.0.0.3 area 0
 network 192.1.0.128 0.0.0.63 area 1
```

路由器 R2

```
Interface e0
 ip address 192.1.0.65 255.255.255.192
Interface s0
 ip address 192.200.10.6 255.255.255.252
router ospf 200
 network 192.200.10.4 0.0.0.3 area 0
 network 192.1.0.64 0.0.0.63 area 2
```

路由器 R3

```
Interface e0
 ip address 192.1.0.130 255.255.255.192
router ospf 300
 network 192.1.0.128 0.0.0.63 area 1
```

路由器 R4

```
Interface e0
 ip address 192.1.0.66 255.255.255.192
router ospf 400
 network 192.1.0.64 0.0.0.63 area 2
```

注：实际配置通配符掩码时，记住其值就等于：块大小-1；/28 块大小是 16，通配符就用 15

## 七、不等量负载均衡（III）

<variance 命令意义>

## 08-ARP 欺骗与木马攻击

### ● ARP 欺骗 (III)

#### 【一、ARP 概念】

ARP (Address Resolution Protocol) 地址解析协议，是一种将 IP 地址转化成物理地址的协议。具体说来就是将网络层 (OSI 的第三层) 地址解析为数据链路层 (OSI 的第二层) 的物理地址 (注: 并不一定指 MAC 地址)。

ip地址	mac地址
192.168.1.1	00-aa-00-62-c6-09
192.168.1.2	00-aa-00-62-c5-03
192.168.1.3	00-aa-01-75-c3-06
.....	.....

ARP 分为请求与响应数据包，请求包是广播，应答包是单播。ARP 协议在设计之初就没有验证机制。ARP 协议并不只在发送了 ARP 请求才接收 ARP 应答。当计算机接收到 ARP 应答数据包，不论其之前是否发过请求包，它都对本地的 ARP 缓存进行更新，将应答中的 IP 和 MAC 地址存储在 ARP 缓存中。ARP 缓存表采用老化的机制，在一段时间里表中的某一行没有使用，就会被删除，这样可以大大减少 ARP 缓存表的长度，加快查询的速度。

输入 “arp -a” 就可以查看 arp 缓存表的内容：

用 “arp -d” 可以删除 arp 缓存表里的所有内容。

用 “arp -s” 可以手动在 arp 表中制定 ip 地址与 mac 地址的对应关系。

#### 【二、ARP 攻击类型】

常见的 ARP 攻击为两种类型：ARP 扫描和 ARP 欺骗：

##### 1、ARP 扫描 (ARP 请求风暴)

通讯模式 (可能)：请求-> 请求-> 请求-> 请求-> 请求-> 应答 -> 请求-> 请求 -> 请求...

描述：正常情况下，主机广播一个 ARP 请求包，就会收到一个 ARP 响应包，当网络中出现 ARP 请求包远远多于 ARP 响应包，就有可能存在 ARP 扫描攻击。大量 ARP 请求广播包会占用网络带宽资源；ARP 扫描一般为 ARP 攻击的前奏。

出现原因 (可能)：\*病毒程序，侦听程序，扫描程序。\*如果网络分析软件部署正确，可能是我们只镜像了交换机上的部分端口，所以大量 ARP 请求是来自与非镜像口连接的其它主机发出的。\*如果部署不正确，这些 ARP 请求广播包是来自和交换机相连的其它主机。

#### 【三、ARP 欺骗】

ARP 协议没有验证机制，它并不只在发送了 ARP 请求才接收 ARP 应答。当计算机接收到 ARP 应答数据包的时，就会对本地的 ARP 缓存进行更新。所以在网络中，如果有人发送一个自己伪造的 ARP 应答，网络可能就会出现问题。这可能就是协议设计者当初没考虑到的！举例说明 arp 欺骗原理：

假设一个网络环境中，网内有三台主机，分别为主机 A、B、C。主机详细信息如下描述：

A 的地址为：IP: 192.168.10.1 MAC: AA-AA-AA-AA-AA-AA

B 的地址为：IP: 192.168.10.2 MAC: BB-BB-BB-BB-BB-BB

C 的地址为：IP: 192.168.10.3 MAC: CC-CC-CC-CC-CC-CC

正常情况下 A 和 C 之间进行通讯，但是此时 B 向 A 发送一个自己伪造的 ARP 应答，而这个应答包中的源 IP 地址是 192.168.10.3 (C 的 IP 地址)，MAC 地址是 BB-BB-BB-BB-BB-BB，这里被 B 伪造了，当 A 据此伪造应答包更新了 ARP 表后，那它发往 C 的 IP 包会在链路层被发往 B 而不是 C。这时对于 A 来说，B 就伪装成 C 了。同理，B 同样向 C 发送一个 ARP 应答，应答包中源 IP 地址：192.168.10.1 (A 的 IP 地址)，MAC 地址是 BB-BB-BB-BB-BB-BB，当 C 据此伪造应答包更新了 ARP 表后，那它发往 A 的 IP 包会被发往 B 而不是 A。这时 B 就伪装成 A 了。这样主机 A 和 C 都被主机 B 欺骗，A 和 C 之间通讯的数据都经过了 B。主机 B 完全可以知道他们之间说的什么：)。这就是典型的 ARP 欺骗过程。注意：一般情况下，ARP 欺骗的某一方应该是网关。

ARP 欺骗存在两种情况：

---

一种是欺骗主机作为“中间人”，被欺骗主机的数据都经过它中转一次，这样欺骗主机可以窃取数据，这就是上面所讲的典型的 ARP 欺骗；

另一种让被欺骗主机直接断网，这类情况就是在 ARP 欺骗过程中，欺骗者只欺骗了其中一方，如 B 欺骗了 A，但是同时 B 没有对 C 进行欺骗，这样 A 实质上是在和 B 通讯，所以 A 就不能和 C 通讯了，也有可能就是欺骗者伪造一个并不存在地址进行欺骗。

#### 【四、常用的防护方法】

1、静态绑定：就是做 IP 和 MAC 静态绑定，在网内把主机和网关都做 IP 和 MAC 绑定。

用命令“arp -s IP MAC 地址”可以实现。例如：“arp -s 192.168.10.1 AA-AA-AA-AA-AA-AA”。

2、使用 ARP 防护软件，主要是欣向 ARP 工具，Antiarp 等。

3、具有 ARP 防护功能的路由器，其实它的原理就是定期的发送自己正确的 ARP 信息。但是路由器的这种功能对于真正意义上的攻击，是不能解决的。ARP 的最常见的特征就是掉线，一般情况下不需要处理一定时间内可以回复正常上网，因为 ARP 欺骗是有老化时间的，过了老化时间就会自动的回复正常。现在大多数路由器都会在很短时间内不停广播自己的正确 ARP 信息，使受骗的主机回复正常，但这总归不是解决问题的根本，它会影响到我们的网络速度。关键是分析并找出产生 ARP 病毒的源头主机，进行彻底的处理。

## 09-网络管理命令

### ● 网络管理命令（ping 、ipconfig 、winipcfg 、netstat 、arp、tracert 和 nslookup）(II)

#### Ping

-t 一直 Ping 指定的计算机,直到从键盘按下 Control-C 中断。

-n 发送 count 指定的 ECHO 数据包数。可以自己定义发送的个数。

#### Ipconfig

当使用 IPConfig 时不带任何参数选项，那么它为每个已经配置了的接口显示 IP 地址、子网掩码和缺省网关值。（个人多次遇到过 ipconfig 命令可以修复 wifi 网络连接受限的状况，与仅仅显示配置的功能矛盾，很奇怪，期待高人解释。以后大家在密码正确而 wifi 连接受限时可以尝试此操作。）

ipconfig /all: IPConfig 能为 DNS 和 WINS 服务器显示它已配置且所要使用的附加信息，并显示本地网卡的物理地址 MAC。

ipconfig /release 和 ipconfig /renew: 只能在向 DHCP 服务器租用 IP 的计算机上起作用。release 释放 IP，renew 重获 IP。

Ipconfig /displaydns: 显示本地 DNS 内容；

#### Winipcfg

Winipcfg 程序采用 Windows 窗口的形式来显示 IP 协议的具体配置信息，类似于 ipconfig

#### Netstat

是一个监控 TCP/IP 网络的工具，它可以显示路由表、实际的网络连接以及每一个网络接口设备的状态信息.Netstat 用于显示与 IP、TCP、UDP 和 ICMP 协议相关的统计数据，一般用于检验本机各端口的网络连接情况。经常使用 netstat -na

-a 显示所有连接和监听端口。

-b 显示包含于创建每个连接或监听端口的可执行组件。

-e 显示以太网统计信息。此选项可以与 -s 选项组合使用。

-n 以数字形式显示地址和端口号。

-p proto 显示 proto 指定的协议连接；proto 可以是下列协议之一：TCP、UDP、TCPv6 或 UDPv6。

-r 显示路由表。

-s 显示按协议统计信息。本选项能够按照各个协议分别显示其统计数据。

#### Arp

-a 显示当前 ARP 项

-d 删除由 inet\_addr 指定的项。

-s 在 ARP 缓存中添加项 inet\_addr ether\_addr



---

## Tracert 跟踪路由

-d 指定不将 IP 地址解析到主机名称。

-h maximum\_hops 指定跃点数以跟踪到称为 target\_name 的主机的路由。

-j host-list 指定 Tracert 实用程序数据包所采用路径中的路由器接口列表。

-w timeout 等待 timeout 为每次回复所指定的毫秒数。

## Nslookup 用于测试或解决 DNS 服务器问题

举例：>nslookup www.airluxgroup.com

Server: server

Address: 192.168.1.2

Non-authoritative answer:

Name: www.airluxgroup.com

Address: 202.96.137.11

以上结果显示，正在工作的 DNS 服务器的主机名为 server，它的 IP 地址是 192.168.1.2，而域名 www.airluxgroup.com 所对应的 IP 地址为 202.96.137.11，这说明该 DNS 服务器已经能顺利实现正向解析了，它的反向解析是否正常呢？也就是说，能否把 IP 地址反向解析为域名吗？

C:\>的后面键入 Nslookup 202.96.137.11，得到结果如和上面的一样，说明 DNS 服务器 server 的反向解析功能也正常。

# 10-Email 配置

## 一、实训目的

- (1) 学会使用 Windows 2000 建立 SMTP 服务器。
- (2) 学会创建虚拟 SMTP 站点和 SMTP 作用域。
- (3) 学会对 SMTP 服务器进行安全性控制。
- (4) Linux 下的 Sendmail 的简单配置理解。

## 二、实训理论基础

**1. 电子邮件地址的格式：**电邮地址格式为：账户名@邮件服务器域名，如 uesrname@domain.com。一般情况下，每个电子邮件用户都会在邮件服务器有一个不同的目录，用于存放用户的电子邮件。

### 2. 电子邮件系统组成部分

- (1) 用户代理。用户代理能够通过一个很友好的接口（如窗口界面）来发送和接收邮件。
- (2) 邮件服务器。主要是接收和发送邮件，同时还向发信人报告邮件传送的情况，如成功交付、被拒绝、丢失。

### 3. SMTP

**SMTP：**简单邮件传输协议（Simple Mail Transfer Protocol），是定义邮件传输的协议，它是基于 TCP 服务的应用层协议，SMTP 是一组规则，用于从源地址到目的地址传送电子邮件。每一个想接收电子邮件的主机都安装了 SMTP 服务器。当主机由用户接收了电子邮件并想传送到另一台服务器，则它联络 SMTP 服务器，SMTP 服务器会做出反应，显示确认、错误消息或特定的请求信息。**SMTP 使用客户/服务器方式。**用户代理把所有要发送的邮件发送给发送人所在的 SMTP 服务器（即邮件服务器），SMTP 服务器监听 25 端口，将接收到的邮件暂时放入邮件缓存队列，并与收信人所在的 SMTP 服务器进行联络、连接，最后收件人所在的 SMTP 服务器把收到的邮件放入收件人的邮箱中，完成信件传递过程。

### 4. POP3

**POP3：**邮局通信协议 3（Post Office Protocol 3），主要用于收信件。POP3 使用客户/服务器方式，收信人利用用户代理，将自己的邮件从邮件服务器的用户邮箱内取回。此用户代理即为负责获取邮件的 POP3 客户。而在邮件服务器上监听 TCP 端口 110，负责读取并发送邮件的就是 POP3 服务进程（即 POP3 服务器）。获取信件时，POP3 服务器需要用户输入合法的账户信息。

### 5. 公用电子邮件服务

对于公用电子邮件服务提供商（比如 163.com）通常使用的工作方式。它们通常同时使用两个邮件服务器地址，如 163.com，它的 POP3 服务器为 pop.163.com，SMTP 服务器为 smtp.163.com。用户通过客户端邮件软件（需要指定 SMTP 及 POP3 服务器地址）连接到 163 免费邮件服务器之后，先要通过账号身份验证，然后进行邮件收发。这里需要两个过程：SMTP 和 POP3。用户使用 SMTP（TCP 端口 25）发送邮件，而 POP3（TCP 端口 110）检索用户的新邮件并将新邮件发送到用户本地。因此，完善的邮件服务需要 SMTP 和 POP3 的共同作用。

## 6. IIS 5.0 提供的 SMTP 服务

所熟悉的电子邮件服务除了依赖于 SMTP 协议之外，还需要 POP 协议的支持，而 POP 协议是 IIS 所不能支持的，所以使用 IIS 5.0 的 SMTP 服务器并不能实现完整的邮件服务。笼统地说，SMTP 负责邮件的传递，例如从客户机到邮件服务器以及服务器之间的传递工作。而 POP 协议能够让客户检索到由 SMTP 发送来的邮件，并将新的邮件下载到用户本地。IIS 5.0 的 SMTP 服务执行两个任务：向其它 SMTP 服务器转发邮件或者把本地域为目的地的消息存放在一个单独的目录中（默认为 Drop）。SMTP 通过文件夹方式实现邮件的传送，一封邮件在存送的各个不同过程（状态）中被 SMTP 放入不同文件夹中。例如，用户只需将待发送的邮件投入发送文件夹就可以由 IIS 实现自动发送，而用户收到的新邮件也是被 IIS 投放到收件文件夹中。由 IIS 自动生成的默认 SMTP 站点具有如右图所示的默认文件夹，它们位于 inetpub\mailroot 文件夹中，主要的功能文件夹有：

Pickup：拾取待发送邮件，用户将待发邮件投入此文件夹。

Queue：因网络繁忙、目标服务器无响应等不能一次发送成功的邮件暂存在此等待继续发送。

Drop：接收所有传入邮件。

Badmail：存放不能投递且不能返回发送者的邮件（称为死信）。

Route、SortTemp、MailBox：IIS 使用这些目录对发往其它服务器的邮件进行排序和重组，从而使投递过程有序、快捷。注意：上述文件夹中，除了 Badmail 和 Drop 可以移到其它分区外，其余文件夹必须保存在 NTFS 分区。

## 三、实训 Window IIS SMTP 配置管理

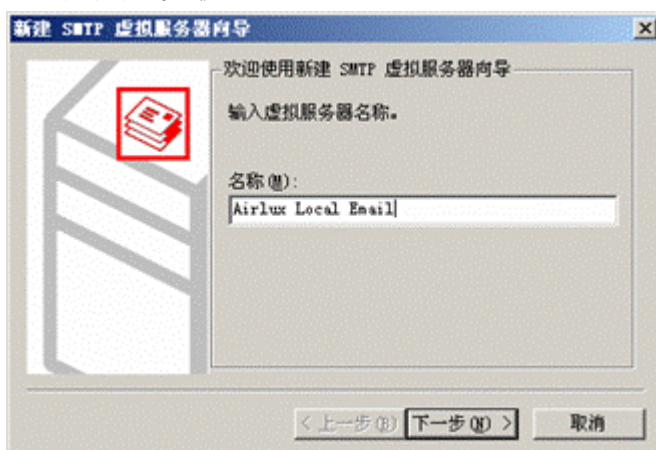
### 1. 创建 SMTP 虚拟服务器

SMTP 服务也有虚拟服务器的概念，通过创建 SMTP 虚拟服务器可以在同一台计算机上实现多个 SMTP 邮件服务器。当然，不同虚拟服务器的默认目录（邮件文件夹）是不同的。在网络中惟一区分 SMTP 服务器的标识有 IP 地址和 TCP 端口号，SMTP 服务的默认 TCP 端口号为 25。如果在 IIS 安装过程中已选择 SMTP 服务，安装完成之后，系统自动生成一个默认 SMTP 站点，它与默认 Web 服务器和默认 FTP 服务器共享系统默认的 IP 地址。

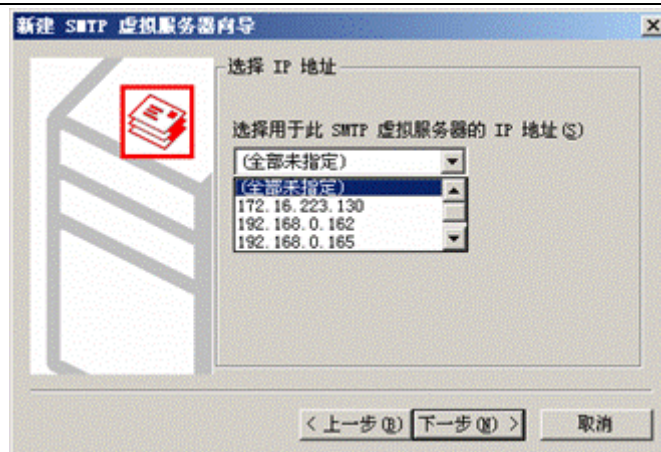
下面将新建一个 SMTP 虚拟服务器站点。

（1）在 IIS 管理控制树中右击计算机结点，在弹出菜单中指向“新建”，单击“SMTP 虚拟服务器”。

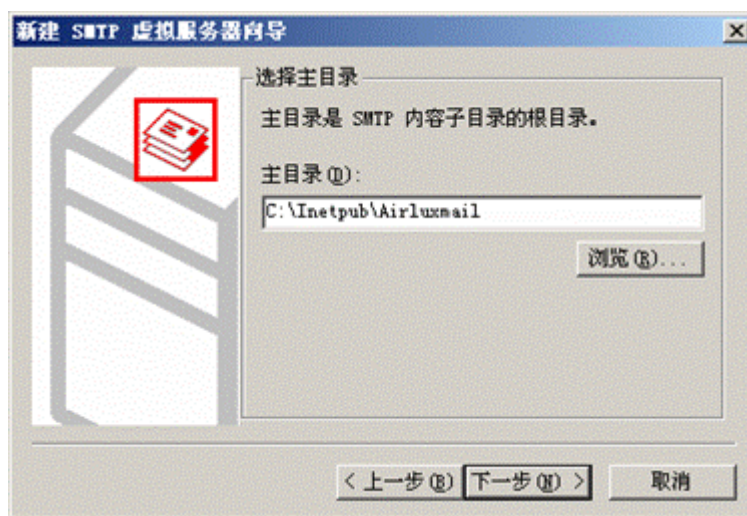
（2）在“SMTP 虚拟服务器创建向导”对话框中指定新站点的标识，在“SMTP 虚拟服务器描述”中的名称实际用于在 IIS 内部区分站点，注意这一名称并非 SMTP 服务器的域名。单击“下一步”按钮。



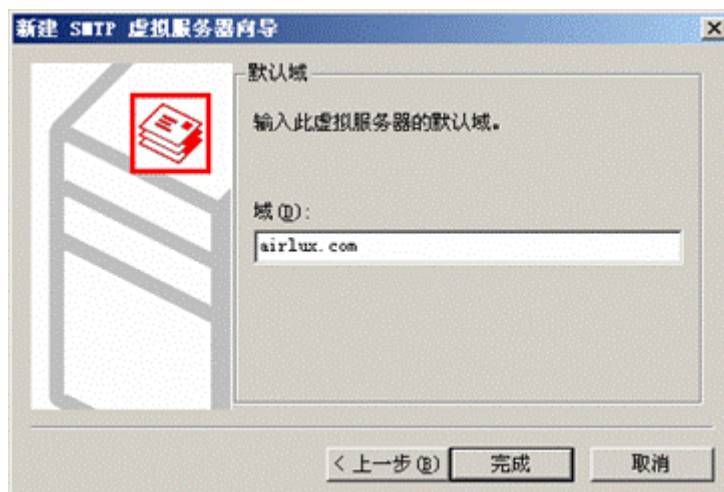
（3）在“选择 IP 地址”对话框中指定 SMTP 虚拟服务器使用的 IP。单击“下一步”。



(4) 在“选择主目录”对话框中，单击“浏览”指定服务器主目录，前面提到的 SMTP 服务文件夹都包含在服务器主目录下。注意，SMTP 主目录必须位于 NTFS 分区中，否则向导将不予认可，但 Badmail 和 Drop 文件夹可以稍后移动到其它分区中。单击“下一步”。



(5) 在“选择默认域”对话框中，指定当前 SMTP 虚拟服务器的默认域名称，默认域将继承全部的站点属性，一个 SMTP 虚拟服务器只能有一个默认域，该域将不能被删除，除非预先将默认域职责转移到其它域中。IIS 将使用这里指定的域名命名默认域。



(6) 单击“完成”按钮结束 SMTP 虚拟服务器创建工作。

## 2. 创建 SMTP 作用域

其实，经过上面的配置，SMTP 虚拟服务器里已经有一个 airlux.com 的作用域了，它是本地默认域，用于标记一般性消息。本地默认域只有一个，并且不能删除。再创建 SMTP 作用域的方法如下：



(1) 在 IIS 管理控制树中右击虚拟 SMTP 服务器结点下一级的“域”子结点，在弹出菜单中指向“新建”，单击“域”。



(2) 制定新建域类型，别名域是本地默认域的一个副本域，它使用与本地默认域相同的系统文件夹（Drop 等目录）；远程域是为了方便对远程站点进行管理而设置的域。单击“下一步”继续。



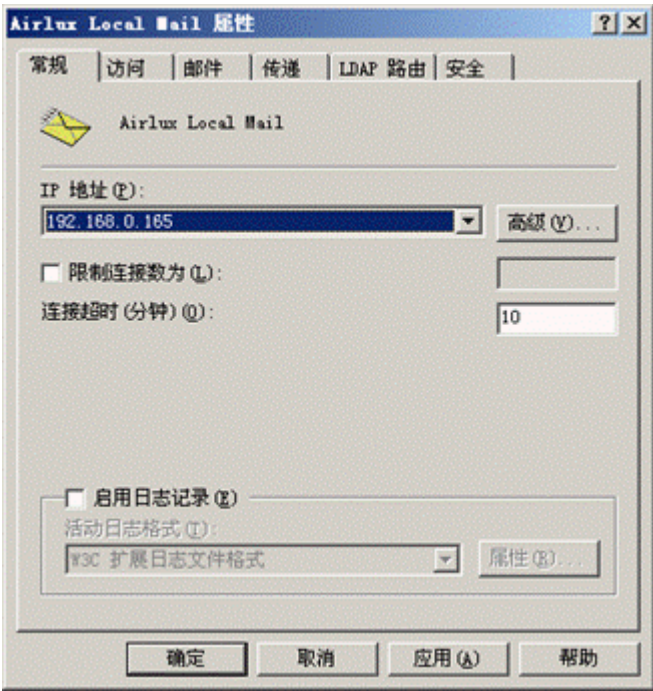
(3) 如图 4.30 所示，在“选择域名”对话框的“名称”栏中，指定用户由此域接收/发送的电子邮件所使用的域名，即用户电子邮件地址中“@”字符后面的域名部分。单击“完成”结束配置。此时，新建的域出现在 SMTP 虚拟服务器地域列表中。



### 3. SMTP 虚拟服务器常规属性

SMTP 虚拟服务器属性表单提供了对 SMTP 虚拟服务器属性进行详细配置的操作界面。在属性的“常规”选项卡中，可以对一般性的虚拟服务器参数进行指定，其执行步骤如下：

(1) IIS 管理控制树中右击 SMTP 虚拟服务器 Airlux Local Mail 结点，在弹出菜单中单击“属性”，在属性表单“常规”选项卡的“名称”栏中，可以修改在创建站点时指定的虚拟服务器标识，默认 SMTP 虚拟服务器是在 IIS 安装时创建的。



(3) 在“IP 地址”下拉列表中，指定此站点使用的 IP 地址（默认以 25 为端口号）。一般，为一个虚拟服务器指定一个 IP 地址完全能够满足用户要求，但是有时也需要复杂的 IP 地址及 TCP 端口号设置，例如同时使用多个 IP 地址作为虚拟服务器的可用地址，或者指定同一（乃至多个）地址的多个 TCP 端口号，这时，就要配置高级 IP 地址属性。单击“高级”按钮，打开高级地址设置对话框。



(4) 在“高级”对话框中的“地址”列表中，默认列出了单一 IP 地址和默认的端口号 25。单击“添加”，指定额外可用的 IP 地址和 TCP 端口号。同一虚拟服务器的标识在数量上是不受限制的。但是，如果这里指定的非默认（25）端口号，客户端邮件程序则需要额外配置。完成后单击“确定”按钮。

(5) 在“常规”选项卡中选择“启用日志记录”复选框，指定活动日志格式，并单击“属性”设定日志的记录时间及方式等参数。

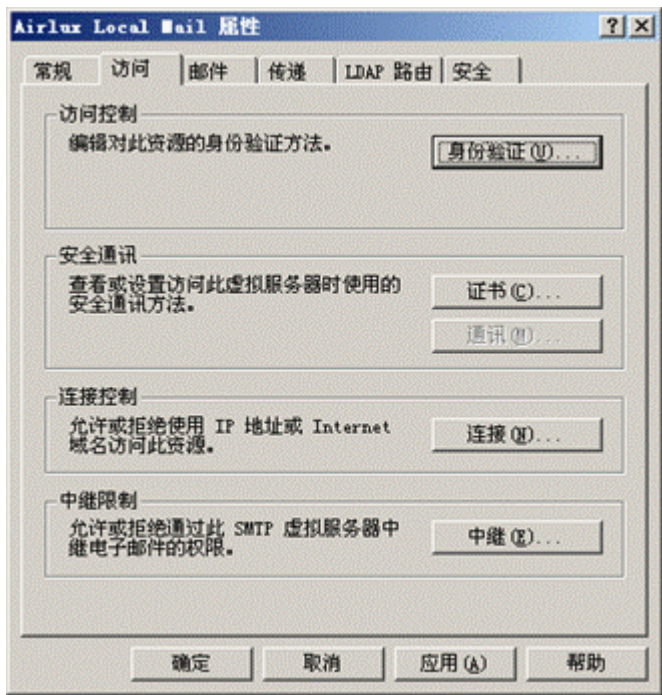
4. SMTP 服务器安全性设置

邮件安全的重要性对于企业而言并不亚于 Web 站点的安全性，某些带有商业秘密的邮件甚至关系到企业的生死存亡，所以，在此单独用一节的篇幅讨论 SMTP 服务安全性的问题。

SMTP 服务器的安全性保护方式是多种多样的，从邮件发送角度区分，有传入安全和传出安全限制；从客户角度区分，有身份验证和 IP 地址、域名限制；从主机角度区分，有中继限制和操作员账号限制；从数据加密角度区分，有账号加密和 SSL 加密以及 TLS 加密。以下将具体分析这些安全要素的配置。

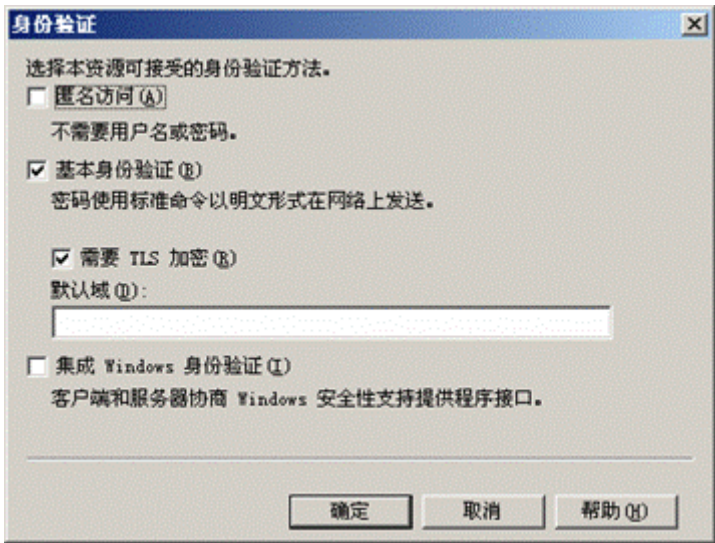


SMTP 服务器的安全性设置主要是在 SMTP 虚拟服务器属性表单中的“访问”选项卡中设置的。此外“安全”选项卡中可以指定站点操作员账号；“传递”选项卡中可以对出站安全进行详细设定。



5. 用户身份验证

在 SMTP 虚拟服务器属性表单中选择“访问”选项卡，单击“身份验证”按钮，打开如图 4.31 所示的“身份验证”对话框。可供 SMTP 服务使用的用户身份验证方法主要有三种：匿名访问、基本身份验证和 Windows 安全程序包。

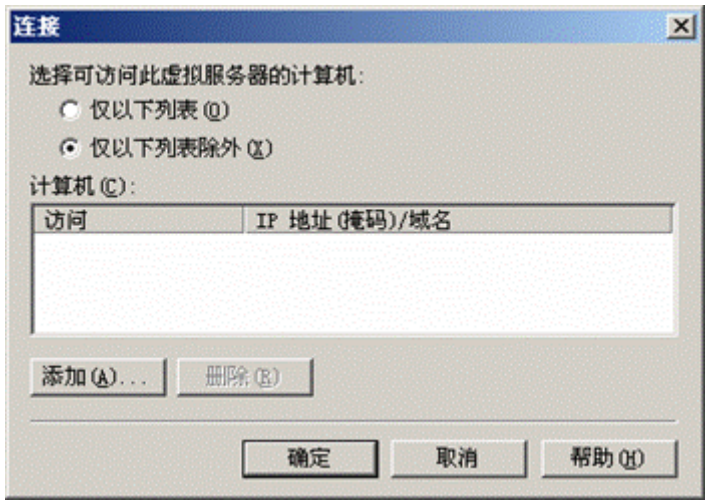


- 1、匿名访问：选择该选项允许对所有的客户机对此资源下的全部内容进行访问，且不需要提供用户名和密码。
- 2、基本身份验证：这是一种标准的身份验证方式，要求用户必须是 Windows 2000 的合法域用户。且使用该方法时，用户账号和密码使用明文（不加密）在网络中传送，并不安全，容易被黑客截获并破解。建议与 SSL（安全套接字）或 TLS（传输层安全）共同使用之。一旦选用基本身份验证，用户不得不承受安全上的风险，IIS 也会提出安全性警告。使用基本身份验证时，需要为用户指定一个用于验证其账号身份的域，在“默认域”栏中指定。此外，需要邮件进行 TLS 加密时，选择相应复选框。
- 3、Windows 身份验证：该选项采用 Windows 提供的安全程序包机制进行安全性用户账号验证，这里的用户密码是经过加密的。这种安全机制是单一、公共的，得到 Windows 系统的统一支持。

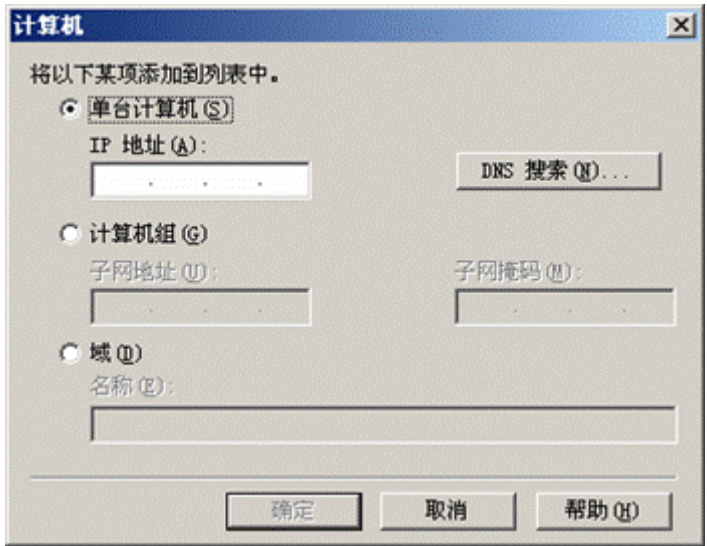
6. IP 地址和域名访问限制



IP 地址和域名访问限制是针对邮件客户机的地址来源进行访问控制的方式。在 SMTP 虚拟服务器属性表单中选择“访问”选项卡，单击“连接”按钮，打开“连接限制”对话框。

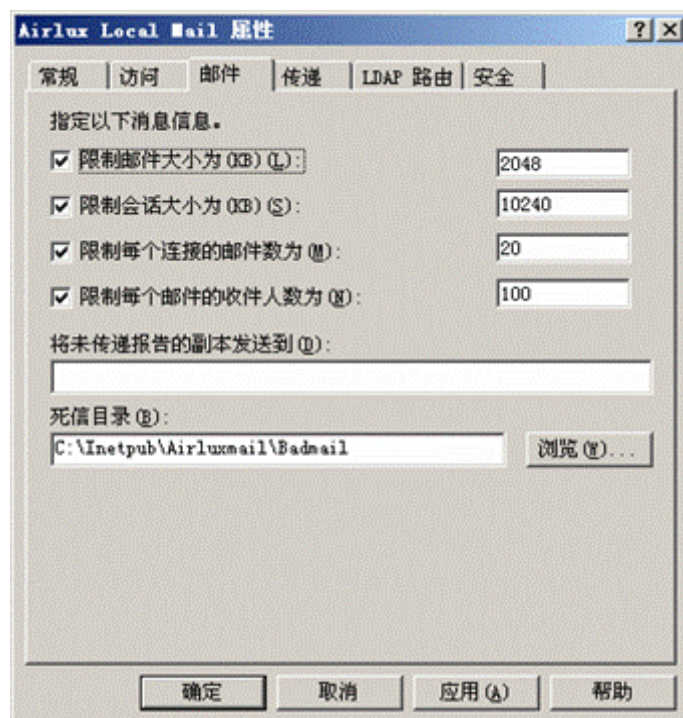


可以采用两种方式之一进行客户机限制：“仅以下列表”和“仅以下列表除外”，前者规定了所有许可客户机（除此之外均被拒绝），后者则规定了所有拒绝访问客户机（除此之外均被许可）。注意对比 WWW 或 FTP 属性表单中的 IP 地址限制方式。有三种地址指定方式可供选择：单台计算机、计算机组、域。

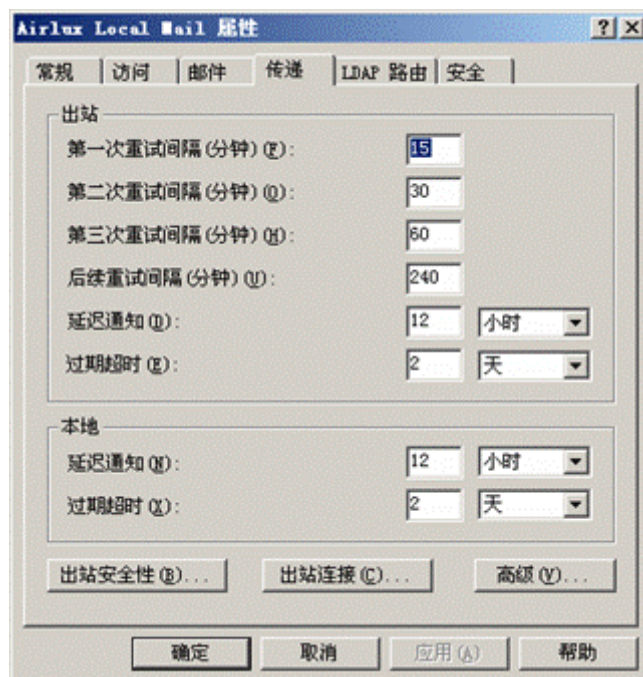


7、设置邮件选项：

其中现在会话大小就是每一次传送邮件的大小。



## 8、设置传递参数



## 四、实训 Linux Sendmail 配置管理

### 三、实训 Window IIS SMTP 配置管理

sendmail 的功能强大，配置起来也十分繁琐。本例仅介绍如何架设小型的局域网电子邮箱。

#### 1、设置 sendmail 作为守卫进程 (daemon) 启动

可以使用以下命令来确认 sendmail 是否已经启动：ps -A|grep sendmail

如果启动了 sendmail，那么这个命令将显示出它的相关信息。如果没有启动，则需要要在 **/etc/rc.d/rc.net** 文件中加上如几行代码：

```
if [-f /usr/lib/sendmail ];
```

---

```
then (cd /usr/spool/mqueue;rm -f if *)  
/usr/lib/sendmail-bd-qlh;echo -n `sendmail`>/dev/console  
fi
```

## 2、设置 sendmail.cf

sendmail.cf 是 sendmail 的配置文件。在安装 Linux 系统之后，它将自动生成一个适合本系统使用的 sendmail.cf 文件，位于 /etc/sendmail.cf

## 3、开启 SMTP、POP 端口

在默认情况下，SMTP 端口是打开的，而 POP 端口是关闭的，我们必须将它打开：

- 1) 用 root (超级权限) 登录到服务器上；
- 2) 编辑文件 **/etc/inetd.conf**；
- 3) 找到描述 POP 端口的语句：**#pop3 stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.pop3d**
- 4) 将 POP3 的描述语句那一行的注释号“#”去掉。
- 5) 如果没有找到，就手工补上这一行；（另外，我们还可以验证一下是否存在 SMTP 端口的描述语句）；
- 6) 存盘后退出。
- 7) 运行 **inetd** 命令，使得设置生效。
- 8) 用以下命令验证，看 POP3 端口是否打开（生效）：**netstat -a |grep pop**

如果打开的话，可以看到以下信息：**tcp 0 0 \*: pop3 \*: \* LISTEN**

## 4、为新用户开 E-Mail 帐号

在 Linux 中开设 E-Mail 帐号十分简单，只要在 Linux 系统中新增一个用户即可。该用户帐号和密码就是 E-Mail 的帐号和密码。如：为新用户 **guest** 开一个 E-Mail 帐号，用以下命令即可：

**adduser guest passwd guest**

这样，该新用户的 E-Mail 地址就是：**guest@fddu2000.com**；密码为：**guest**

## 5、为 E-Mail 帐号设置别名

如果某个用户想使用多个 E-Mail 地址，可通过设置别名的方法来实现。比：用户“杜方冬”想同时拥有 E-Mail 地址：**dfd@fddu2000.com ;fddu@fddu2000.com ;fddu2000@fddu2000.com** 就可通过以下步骤来实现这样的别名设置：

- 1) 以 root 登录服务器；
- 2) 新增一个账号 **dfd**；
- 3) 编辑文件 **/etc/aliases**，加上两行：  
**fddu:dfd**  
**fddu2000:dfd**
- 4) 存盘退出；
- 5) 执行命令：**newaliases**

这样，用户杜方冬就拥有三个邮件地址，杜方冬只需使用一个 E-Mail 帐号：**dfd@fddu2000.com** 就可以接收所有寄给以上三个 E-Mail 邮件地址的电子邮件。

## 6、邮箱空间的限定

如果你想控制用户邮箱空间的大小，可以对它进行限定。实现方法是利用磁盘限额功能来实现的。电子邮件的暂存空间是在 **/var/spool/mail** 目录下，只要通过磁盘限额设定每一个用户在这个目录下能使用的最大空间就可以了。

# 11-DNS 配置

DNS 服务器用于 TCP/IP 网络（如一般的局域网或互联网等）中，担任“DNS 名称 $\longleftrightarrow$ IP 地址”翻译机的角色,通过用户友好的名称代替难记的 IP 地址以定位计算机和服务。DNS 名称是由主机名称与域名称组成，以“www.nh.edu.sh.cn”为例：

🌐www：就是 WEB 站点所在计算机的主机名称   🌐nh.edu.sh.cn：就是 www 这台计算机所在的域名。

正向解析（域名 $\rightarrow$ IP 地址）；反向解析（IP 地址 $\rightarrow$ 域名）

## DNS 的结构

如果将整个因特网的“DNS 名称 $\longleftrightarrow$ IP 地址”翻译工作，都交由一台 DNS 服务器来做，不但效率差，也提高了风险，因此实际上 DNS 是采用层叠式的分布式数据库的结构。DNS 查询过程：

假如小明要在学校查询 [www.nh.edu.cn](http://www.nh.edu.cn)。学校默认 DNS 服务器>>RootDNS>>CN>>EDU>>NH。因此，只要你需要用到如“www.eicnh.sh.cn”之类域名的地方，你都得首先确保已为此名字在 DNS 服务器中作好了相应的和 IP 地址的映像工作。

下面通过实例说明 DNS 服务器的设置方法：

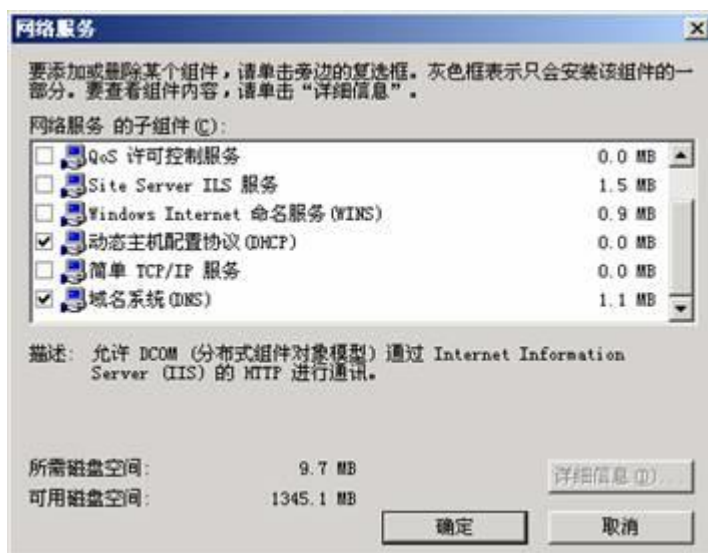
问题 1、我在学校内部网上建立了一个 WEB 服务器，但是它只能用 IP 地址或计算机名进行访问，如何使它同时可用如 <http://www.webadmin.com> 形式的域名进行访问呢？”

问题 2：不管是在局域网还是互联网上，计算机在网络上通讯时本来只能识别如“202.109.122.105”之类的数字地址，那么为什么当我们打开浏览器，在地址栏中输入如“www.eicnh.sh.cn”的域名后，就能看到我们所需要的页面呢？

上面的两个问题，都只是一个 IP 地址和域名相互“翻译”的过程。前者得建立一个指向相应 IP 地址的域名映像记录；对于后者，此记录已经建立并且在生效了。而这种“翻译”记录的建立，则需要用到同一种被称之为“DNS 服务器”的计算机。本文将以 Windows Server 自带的 DNS 服务为例，一步步教会你如何在局域网中完成这个“翻译系统”的组建工作。另外，再用 Bind 来简单介绍 Linux 下的 DNS 配置。

## 1、添加 DNS 服务

默认的，当你安装好 Win2000 之后，DNS 服务并没有被添加进去。请打开“控制面板 $\rightarrow$ 添加/删除程序 $\rightarrow$ 添加/删除 Windows 组件” $\rightarrow$ “网络服务”，选中其下的“DNS 服务器”一项，“确定”即可（图 1）。



## 2、DNS 服务器的配置和管理

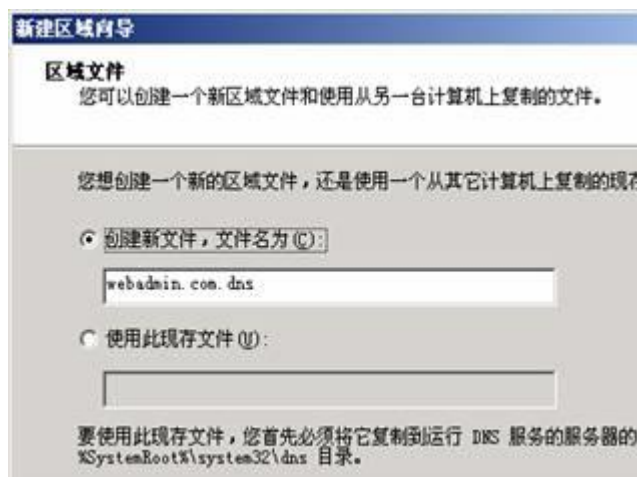
实例 1. 假设本机拥有一个“192.168.0.1”的 IP 地址，如何实现它与“my.webadmin.com”、“www.webadmin.com”和“ftp.webadmin.com”三个域名的对应。

实例 2. 假设本机还拥有如“192.168.0.2”和“192.168.0.3”的 IP 地址，也想要让它们分别和“www.webadmin.com”及“edu.webadmin.com”两个域名对应起来。（这叫 DNS 的循环复用，可以用多台 web 服务器起到客户访问负载均衡的作用）

### 2.1 实例 1 的实现



1. 通过选“开始→程序→管理工具→DNS”来打开 DNS 控制台管理器（以下简称“DNS 管理器”）。
2. 建立“webadmin.com”区域
  - （1）在 DNS 管理器中，在“SERVER”（本服务器名）上单击右键，选“新建区域”以进入新建区域向导。
  - （2）在向导中，应选“标准主要区域”；然后选“正向搜索区域”；各步选择之后单击“下一步”继续。
  - （3）随后在“区域名->名称”后的文字框中输入“webadmin.com”；下一步，用默认，创建新文件



- （4）下面都用默认项完成区域的建立。在 SERVER→正向搜索区域”里可看到“webadmin.com”区域。
3. 新建主机记录(A)

右击“webadmin.com”区域，选“新建主机(A)”，在对话框的“名称”输入主机名“www”，“IP 地址”处输入 IP 地址“192.168.0.1”，再单击“添加主机”按钮，即成功地创建了主机地址记录“www.webadmin.com”。



4. 新建别名(CNAME)

右击“webadmin.com”区域，选“新建别名”，在其后的对话框中的“别名”处输入“ftp”，“目标主机的完全合格的名称”中输入“www.webadmin.com”（或用“浏览”逐步选择），最后“确定”即可为“www.webadmin.com”建立一个名为“ftp.webadmin.com”的别名记录（图 4）。



5. 再用和上步类似的方法来为“www.webadmin.com”建立一个名为“my.webadmin.com”的别名记录。

6. 剩下的工作就是检验工作的成效了！

“ping www.webadmin.com”的格式去一一测试，如果所建立的域名“www.webadmin.com”、“ftp.webadmin.com”和“my.webadmin.com”均能显示出连接的四行如“Reply from 192.168.0.1: bytes=32 time<10ms TTL=128”的响应，则恭喜你成功了！

也可以用 nslookup 来验证。

**注意：**一定要正确设置 IP 地址和 DNS 服务器的地址。否则看不到上述信息。

### 3.2.2 实例 2 的实现

1. 先把“webadmin.com 前文所述方法建立好之后，再在其下建立“www”的“主机”，将其 IP 地址对应到“192.168.0.2 即可。

2. 把“webadmin.com，再在其下建立“edu 主机”，将其 IP 地址对应到“192.168.0.2 即可。

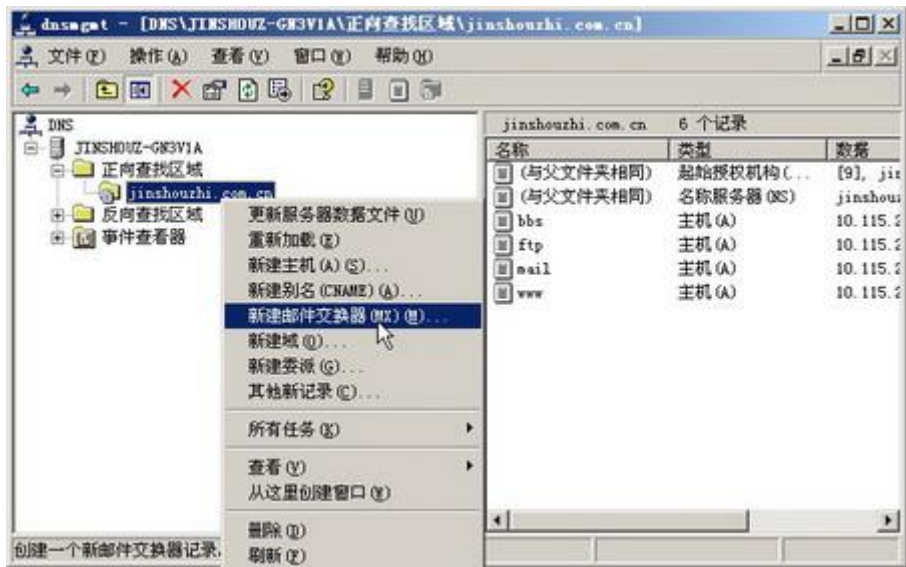
#### DNS 服务器配置教程 添加 MX 记录

MX (Mail Exchanger, 邮件交换) 记录用以向用户指明可以为该域接收邮件的服务器。那么为什么要添加 MX 记录呢？首先用户来举一个例子。如用户准备发邮件给 chhuian@jinshouzhi.com.cn，这个邮件地址只能表明收邮件人在 jinshouzhi.com.cn 域上拥有一个账户。可是仅仅知道这些并不够，因为电子邮件程序并不知道该域的邮件服务器地址，因此不能将这封邮件发送到目的地。而 MX 记录就是专门为电子邮件程序指路的，在 DNS 服务器中添加 MX 记录后电子邮件程序就能知道邮件服务器的具体位置（即 IP 地址）了。在主 DNS 服务器中添加 MX 记录的操作步骤如下所述：

Step1 在 DNS 控制台窗口中首先添加一个主机名为 mail 的主机记录，并将域名 mail.jinshouzhi.com.cn 映射到提供邮件服务的计算机 IP 地址上。

Step2 在“正向查找区域”目录中右键单击准备添加 MX 邮件交换记录的域名，选择【新建邮件交换器 (MX)】命令，如图所示：





Step3 打开“新建资源记录”对话框，在【邮件服务器的完全合格的域名（FQDN）】编辑框中输入事先添加的邮件服务器的主机域名（如 mail.jinchouzhi.com.cn）。或单击【浏览】按钮，在打开的“浏览”对话框中找到并选择作为邮件服务器的主机名称（如 mail），如图所示：



Step4 返回“新建资源记录”对话框，当该区域中有多个 MX 记录（即有多个邮件服务器）时，则需要在【邮件服务器优先级】编辑框中输入数值来确定其优先级。通过设置优先级数字来指明首选服务器，数字越小表示优先级越高。最后单击【确定】按钮使设置生效，如图所示。



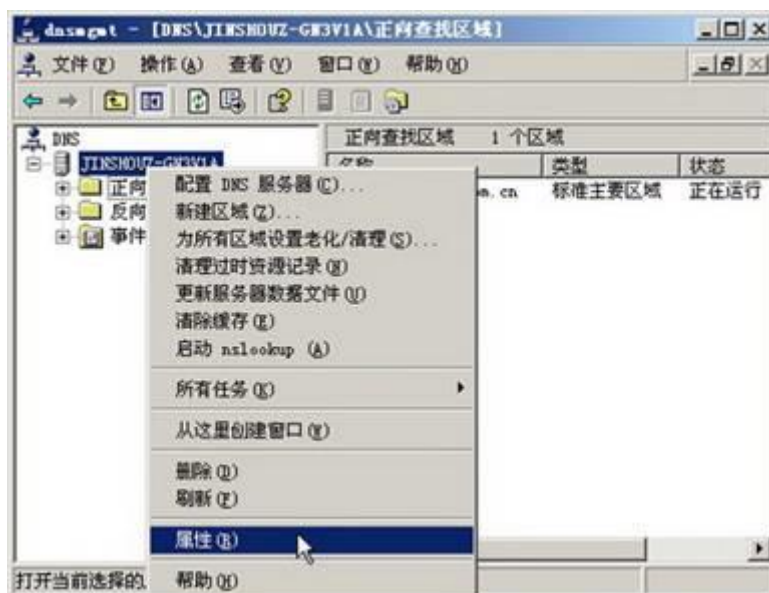
提示：一般情况下【主机或子域】编辑框中应该保持为空，这样才能得到诸如 user@jinchouzhi.com.cn 之类的信箱地址。如果在【主机或子域】编辑框中输入内容（如 mail），则信箱名将会成为 user@mail.jinchouzhi.com.cn。

Step5 重复上述步骤可以添加多个 MX 记录，并且需要在【邮件服务器优先级】编辑框中分别设置其优先级。

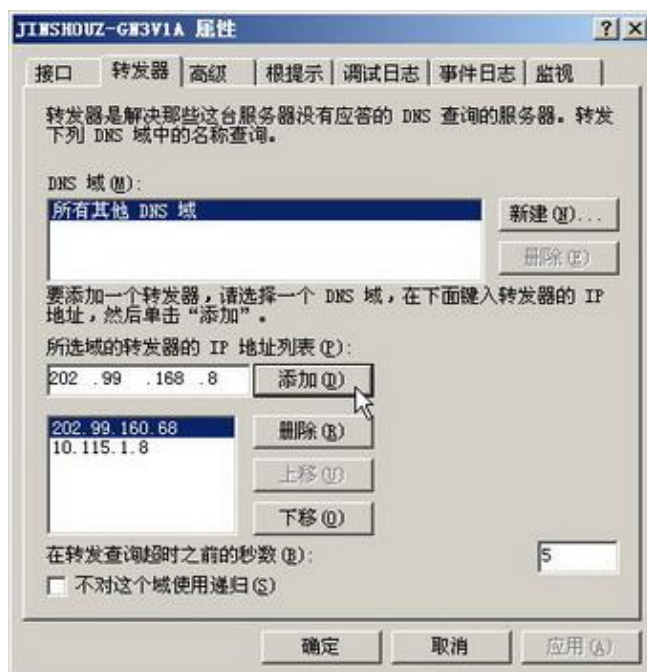
### DNS 服务器配置教程（5）设置 DNS 转发器

尽管在 DNS 安装配置的过程中已经设置了 DNS 转发器，但有时还需要添加多个 DNS 转发器或调整 DNS 转发器的顺序。设置 DNS 转发器的操作步骤如下所述：

Step1 打开 DNS 控制台窗口，在左窗格中右键单击准备设置 DNS 转发器的 DNS 服务器名称，选择【属性】命令，如图所示。

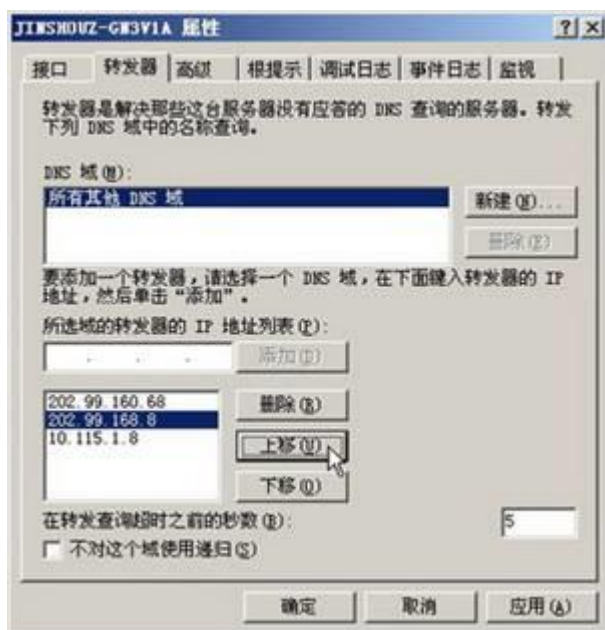


Step2 打开服务器属性对话框，并切换到【转发器】选项卡。在【所选域的转发器的 IP 地址列表】编辑框中输入 ISP 提供的 DNS 服务器的 IP 地址，并单击【添加】按钮，如图所示。



提示：重复操作可以添加多个 DNS 服务器的 IP 地址。需要注意的是，除了可以添加本地 ISP 提供的 DNS 服务器 IP 地址外，还可以添加其他地区 ISP 的 DNS 服务器 IP 地址。

Step3 用户还可以调整 IP 地址列表的顺序。在转发器的 IP 地址列表中选中准备调整顺序的 IP 地址，单击【上移】或【下移】按钮即可进行相关操作。一般情况下应将响应速度较快的 DNS 服务器 IP 地址调整至顶端。单击【确定】按钮使设置生效，如图所示。



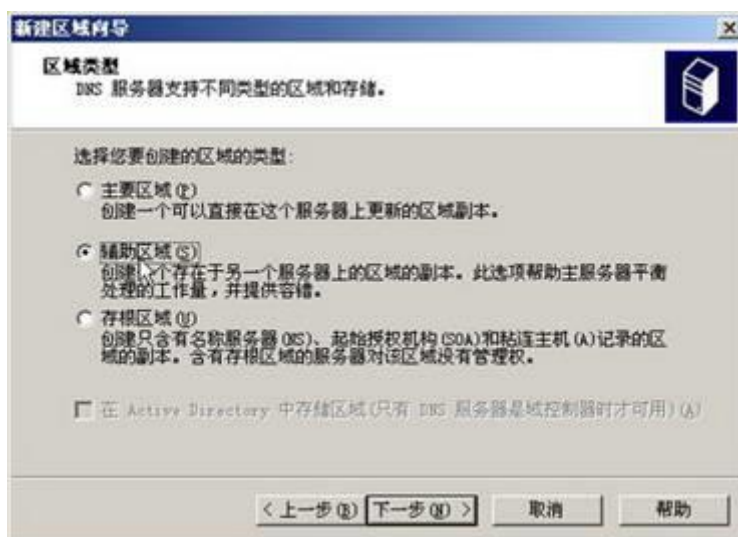
## DNS 服务器配置教程（6）创建辅助区域

为了防止 DNS 服务器由于各种软硬件故障导致停止 DNS 服务，建议在同一个网络中部属两台或两台以上的 DNS 服务器。其中一台作为主 DNS 服务器，其他的作为辅助 DNS 服务器。当主 DNS 服务器正常运行时，辅助 DNS 服务器只起备份作用。当主 DNS 服务器发生故障后，辅助 DNS 服务器立即启动承担 DNS 解析服务。另外，辅助 DNS 服务器会自动从主 DNS 服务器中获取相应的数据，因此无需在辅助 DNS 服务器中添加各个主机记录。创建辅助区域的步骤如下所述：

Step1 在另一台运行 Windows Server 2003 (SP1) 或 Windows 2000 Server 的服务器中安装 DNS 服务器组件，然后打开 dnsmgmt 窗口。在左窗格中展开 DNS 服务器目录，然后右键单击“正向查找区域”目录，选择【新建区域 (Z)】命令，如图所示。



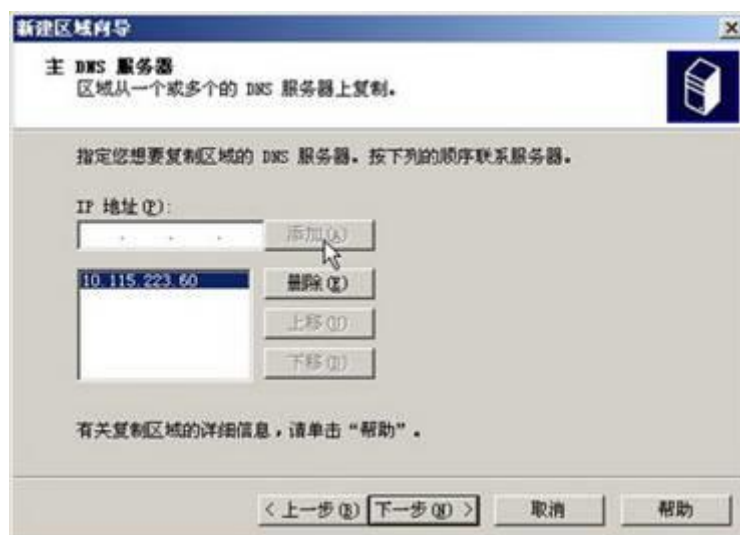
Step2 打开“新建区域向导”，在欢迎对话框中单击【下一步】按钮。在打开的“区域类型”对话框中选中【辅助区域】单选按钮，并单击【下一步】按钮，如图所示。



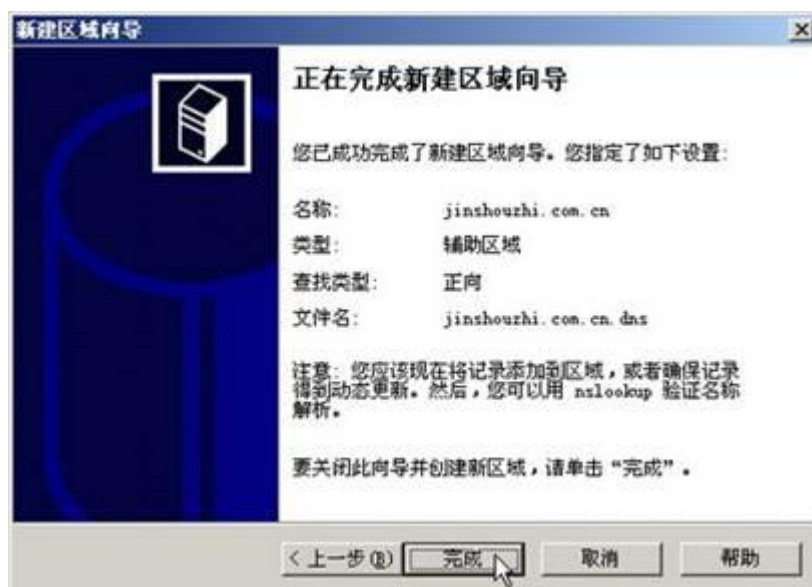
Step3 在打开的“区域名称”对话框中需要输入区域名称，注意这里输入的区域名称必须和主要区域的名称完全相同。用户在【区域名称】编辑框中输入 jinshouzhi.com.cn，并单击【下一步】按钮，如图所示。



Step4 打开“主 DNS 服务器”对话框。在【IP 地址】编辑框中输入主 DNS 服务器的 IP 地址，以便从主 DNS 服务器中复制数据。完成输入后依次单击【添加】→【下一步】按钮，如图所示。



Step5 最后打开“正在完成新建区域向导”对话框，列出已经设置的内容。确认无误单击【完成】按钮完成辅助 DNS 区域的创建过程，该辅助 DNS 服务器会每隔 15 分钟自动和主 DNS 服务器进行数据同步操作，如图：





---

## LINUX 下 DNS 服务器管理与设置

DNS 服务的安装:

[root@localhost /]# rpm -q bind ---查看服务器是否安装了 DNS 服务。

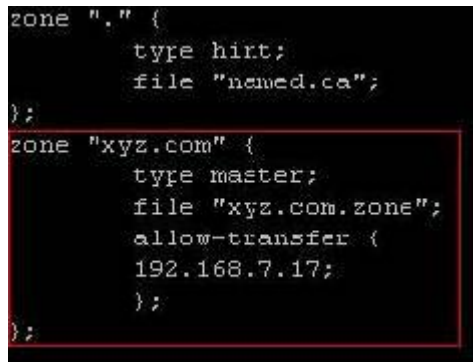
如果没有安装则将 Red Hat Enterprise Linux as4 的第四张安装盘中的 bind-9.2.4.i386.rpm 和 bind-chroot-9.2.4.i386.rpm 文件用 FTP 传送到服务器中, 然后使用下面的命令安装 DNS。

[root@localhost /]# rm -ivh /home/test/bind-9.2.4.i386.rpm

[root@localhost /]# rm -ivh /home/test/bind-chroot-9.2.4.i386.rpm

配置主域名服务器

用 vi 命令编辑 **/etc/named.conf** 文件, 在其内添加以下内容。如图:



```
zone "." {
    type hint;
    file "named.ca";
};
zone "xyz.com" {
    type master;
    file "xyz.com.zone";
    allow-transfer {
        192.168.7.17;
    };
};
```

保存并退出, 再在命令行下 @localhost /]# vi /var/named/chroot/var/named/xyz.com.zone 建 xyz.com.zone 区域文件, 在其内容中写入

\$ttl 38400

xyz.com. IN SOA dns .xyz.com. admin.xyz.com. (

2005090508

10800

3600

604800

38400 )

xyz.com. IN NS dns.xyz.com.

dns.xyz.com. IN A 192.168.16.177 (A 表示新建的主机记录)

www.xyz.com. IN A 192.168.16.9

mail.xyz.com. IN A 192.168.16.178

bbs.xyz.com. IN CNAME [www.xyz.com](http://www.xyz.com). (CNAME 表示建立别名)

xyz.com. IN MX 10 mail.xyz.com (MX 表示建立邮件交互记录域名)

保存并退出。

[root@localhost /]# /etc/init.d/named start 启动服务即可。

DNS 客户端测试

只需在“本地连接”里面的 TCP/IP 里输入一下测试的 DNS 地址就可以了。可以用 ping 或 nslookup 命令来检测 DNS 是否安装成功。

### ➤ 提示

DNS 查询通常使用的是 UDP 和 TCP 端口 53, 所以在防火墙打开的情况下, 需开启这些端口。

DNS 客户端在指定 DNS 的时候, 测试的域名有时不对, 这与 DNS 缓存有关。

## 3.3 常见问题解答

Q: 我听说有个 hosts 文件, 它是用来做什么的? 和 DNS 服务器有什么关系? 我可以用 hosts 文件来代替 DNS 服务器吗?



---

**A:** 主机文件 `hosts` 在各种版本的 Windows 中都可以找到（通过查找的方式寻找），它里面可以手动输入域名和 IP 地址的映像表；不过这里的“域名”实际上被当作“主机名”来看了。在实际效果中，`hosts` 文件可以被看成是只能在本机使用的 DNS 服务器。如果你的域名解析工作只需要满足本机使用，则可以只用 `hosts` 文件；如果你还想要其它计算机使用你的 DNS 服务，则不行。

**Q:** 在 Win2000 中安装 DNS 组件之前是否需要先升级到域控制器？

**A:** 不需要。普通的独立服务器也一样可以安装和使用 DNS 服务。

**Q:** 我想建立几个域名，分别让它们只可以在 HTTP 浏览、FTP 登录和 E-mail 收发等方面用上，那么我在 DNS 服务器中应该如何操作？

**A:** 请一定理解这一点：DNS 服务器只提供域名和 IP 地址的映像工作，而那个域名究竟用来做什么，并不是由 DNS 服务器控制，而是由其对应的 IP 地址所绑定的相关服务器（HTTP、FTP 或 E-mail 等）来决定的。

举个例子。让我们假设你已安装好一个 E-mail 服务器，其绑定的 IP 地址为“192.168.0.3”，如果你在 DNS 服务器中，将域名“`www.webadmin.com`”和“`ftp.webadmin.com`”及“`mail.webadmin.com`”都指向此 IP 地址，则你用 IP 地址或此三个域名中的任何一个都可以访问这个 E-mail 服务器；如果你只让域名“`mail.webadmin.com`”绑定此 IP 地址，就只有 IP 地址和此域名才可以访问它。

**Q:** 我想建立包括的域名如同“`public.school.sh.cn`”和“`pub2.school.sh.cn`”之类的 DNS 记录，对于这么长的域名，在具体操作时应该怎样做呢？

**A:** 你可以将“`school.sh.cn`”部分当成一个“区域”来建立，然后在其下新建“名称”分别为“`public`”和“`pub2`”的主机记录即可。

**Q:** 我需要建立大量如“`webadmin.com`”、“`adminweb.com`”形式的以“.com”结尾的 DNS 记录，如何进行合理的安排呢？

**A:** 对于这种情况，你可以将“`com`”看成一个单独的“区域”建立好；再在此区域下面将“`webadmin`”等看成是一个一个的“域”添加进去就行了。

**Q:** 上文讲的都是建立在局域网中的 DNS 记录，如果我想在自己的计算机上建立互联网上域名的 DNS 映射记录，那这和局域网中的操作有什么不同吗？

**A:** 当然有所不同！不过这个不同不是在建立 DNS 记录的具体操作上，而在于建立此 DNS 记录之前的一些必要条件。

1. 你的计算机所绑定的 IP 地址必须是互联网上“合法”的那种。如果你只是在局域网中，IP 地址不会有冲突的问题，因此你可以随意选择使用，而在互联网中，由于 IP 地址资源有限，都有做相应的控制和分配。一般在当地电信部门可以申请到这种“合法”的 IP 地址。这是自己解析互联网上域名的前提。

2. 你的域名已进行了合法申请，并且指向了你的 IP 地址。和 IP 地址一样的道理，互联网上的域名也是不能被你任意使用的，你需要先向相关的网络管理中心（比如 InterNIC、CNNIC 或其代理商）申请成功此域名，并设置成功方可。好了，如果你已满足了以上所必需的那两个条件，则在你自己的计算机上建立 DNS 映射的方法就和前文没什么不同了。

**Q:** 我所在的学校有一条专线直接连到信息中心机房，并且有一个固定的、在互联网上“合法”的 IP 地址；最近我们在“中国教科研网”申请了一个国际域名，由我们自己的计算机解析。请问，怎样才能够利用现有的这些资源，让互联网上的其它任何用户均可以浏览到放在我们自己的计算机上的主页呢？

**A:** 你需要在自己的计算机上建立一个 Web 服务器。一般来说，使用 Win2000 自带的 IIS（互联网信息服务）就可以很轻松地实现这种功能了。IIS 中除了包括 Web 服务之外，还提供 FTP 和 SMTP 服务，不再需要你购买第三方软件，对于小型的 Web 站点，有它就可以满足普通需求了。

## 12-Apache 主要配置参数

Apache 服务器的设置文件位于 `/usr/local/apache/conf/` 目录下，传统上使用三个配置文件 `httpd.conf`、`access.conf` 和 `srm.conf`，来配置 Apache 服务器的行为。`httpd.conf` 提供了最基本的服务器配置，是对守护程序 `httpd` 如何运行的技术描述；事实上当前版本的 Apache 将所有配置参数均放在了一个配置文件 `httpd.conf` 中。以下使用缺省提供的 `httpd.conf` 为例，解释 Apache 服务器的各

---

个设置选项。然而不必因为它提供设置的参数太多而烦恼，基本上这些参数都很明确，也可以不加改动运行 Apache 服务器。但如果需要调整 Apache 服务器的性能，以及增加对某种特性的支持，就需要了解这些设置参数的含义。httpd.conf 中首先定义了一些 httpd 守护进程运行时需要参数，来决定其运行方式和运行环境。

### **ServerType standalone**

ServerType 定义服务器的启动方式，缺省值为独立方式 standalone，httpd 服务器将由其本身启动，并驻留在主机中监视连接请求。在 Linux 下将在启动文件 /etc/rc.d/rc.local/init.d/apache 中自动启动 Web 服务器，这种方式是推荐设置。启动 Apache 服务器的另一种方式是 inet 方式，使用超级服务器 inetd 监视连接请求并启动服务器。当需要使用 inetd 启动方式时，便需要更改为这个设置，并屏蔽/etc/rc.d/rc.local/init.d/apache 文件，以及更改/etc/inetd.conf 并重启 inetd，那么 Apache 就能从 inetd 中启动了。两种方式的区别是独立方式是由服务器自身管理自己的启动进程，这样在启动时能立即启动服务器的多个副本，每个副本都驻留在内存中，一有连接请求不需要生成子进程就可以立即进行处理，对于客户浏览器的请求反应更快，性能较高。而 inetd 方式要由 inetd 发现有连接请求后才去启动 http 服务器，由于 inetd 要监听太多的端口，因此反应较慢、效率较低，但节约了没有连接请求时 Web 服务器占用的资源。因此 inetd 方式只用于偶尔被访问并且不要求访问速度的服务器上。

### **ServerRoot "/usr/local"**

ServerRoot 用于指定守护进程 httpd 的运行目录，httpd 在启动之后将自动将进程的当前目录改变为这个目录，因此如果设置文件中指定的文件或目录是相对路径，那么真实路径就位于这个 ServerRoot 定义的路径之下。由于 httpd 会经常进行并发的文件操作，就需要使用加锁的方式来保证文件操作不冲突，由于 NFS 文件系统在文件加锁方面能力有限，因此这个目录应该是本地磁盘文件系统，而不应该使用 NFS 文件系统。

### **Alias /icons/ "/www/icons/"**

#### **Options Indexes MultiViews**

#### **AllowOverride None**

#### **Order allow,deny**

#### **Allow from all**

Alias 参数用于将 URL 与服务器文件系统中的真实位置进行直接映像，一般的文档将在 DocumentRoot 中进行查询，然而使用 Alias 定义的路径将直接映射到相应目录下，而不再到 DocumentRoot 下面进行查询。因此 Alias 可以用来映射一些公用文件的路径，例如保存了各种常用图标的路径。这样使得除了使用符号连接之外，文档根目录（DocumentRoot）外的目录也可以通过使用了 Alias 映射，提供给浏览器访问。定义好映射的路径之后，应该需要使用 Directory 语句设置访问限制。

### **ScriptAlias /cgi-bin/ "/www/cgi-bin/"**

#### **AllowOverride None**

#### **Options None**

#### **Order allow,deny**

#### **Allow from all**

ScriptAlias 也是用于 URL 路径的映射，但与 Alias 的不同在于，ScriptAlias 是用于映像 CGI 程序的路径，这个路径下的文件都被定义为 CGI 程序，通过执行它们来获得结果，而非由服务器直接返回其内容。缺省情况下 CGI 程序使用 cgi-bin 目录作为虚拟路径。

### **虚拟主机**

**#NameVirtualHost 12.34.56.78:80**

**#NameVirtualHost 12.34.56.78**

**# ServerAdmin webmaster@host.some\_domain.com**

**# DocumentRoot /www/docs/host.some\_domain.com**

**# ServerName host.some\_domain.com**

**# ErrorLog logs/host.some\_domain.com-error\_log**

**# CustomLog logs/host.some\_domain.com-access\_log common**

缺省设置文件中的这些内容是用于设置命名基础的虚拟主机服务器时使用。其中 NameVirtualHost 来指定虚拟主机使用的 IP 地址，这个 IP 地址将对应多个 DNS 名字，如果 Apache 使用了 Listen 参数控制了多个端口，那么就可以在这里加上端口号以进一步进行区分对不同端口的不同连接请求。虚拟主机是在一台 Web 服务器上，可以为多个单独域名提供 Web 服务，并且每个域名都完全独立，

---

包括具有完全独立的文档目录结构及设置，这样域名之间完全独立，不但使用每个域名访问到的内容完全独立，且使用另一个域名无法访问其它域名提供的网页内容。虚拟主机的概念对于 ISP 来讲非常有用，因为虽然一个组织可以将自己的网页挂在具备其它域名的服务器上的下级网址上，但使用独立的域名和根网址更为正式，易为众人接受。ISP 没有必要为一个机构提供一个单独的服务器，完全可以使用虚拟主机能力，使服务器为多个域名提供 Web 服务，而且不同的服务互不干扰，对外就表现为多个不同的服务器。

有两种设定虚拟主机的方式，一种是基于 HTTP 1.0 标准，需要一个具备多 IP 地址的服务器，再配置 DNS 服务器，给每个 IP 地址以不同的域名，最后才能配置 Apache 的配置文件，使服务器对不同域名返回不同的 Web 文档。由于这需要使用额外的 IP 地址，对每个要提供服务的域名都要使用单独的 IP 地址，因此这种方式实现起来问题较多。可以在一个网络界面上绑定多个 IP 地址，Linux 下需要使用 ifconfig 的 alias 参数来进行这个配置，但此时会影响网络性能。

HTTP 1.1 协议中规定了对浏览器和服务器通信时，服务器能够跟踪浏览器请求的是哪个主机名字。可以利用这个新特性更轻松的设定虚拟主机。这种方式不需要额外的 IP 地址，但需要新版本的浏览器支持。这种方式已经成为建立虚拟主机的标准方式。要建立非 IP 基础的虚拟主机，多个域名是不可少的配置，因为每个域名就对应一个要服务的虚拟主机。因此需要更改 DNS 服务器的配置，为服务器增加多个 CNAME 选项，如：

**linux IN A 192.168.1.64**

**vhost1 IN CNAME linux**

**vhost2 IN CNAME linux**

基本的设置选项都是为了 linux 主机设定的，如果要为 vhost1 和 vhost2 设定虚拟主机，就要使用 VirtualHost 语句定义不同的选项，在语句中可以使用配置文件前面中的大部分选项，而可以重新定义几乎所有的针对服务器的设置。

**NameVirtualHost 192.168.1.64**

**DocumentRoot /www/data**

**ServerName linux.example.org.cn**

**DocumentRoot /vhost1**

**ServerName vhost1.example.org.cn**

**DocumentRoot /vhost2**

**ServerName vhost2.example.org.cn**

这里需要注意的是，VirtualHost 的参数地址一定要和 NameVirtualHost 定义的地址相一致，必须保证所有的值严格一致，Apache 服务器才承认这些定义是为这个 IP 地址定义的虚拟主机。此外，定义过 NameVirtualHost 之后，那么对这个 IP 地址的访问都被区分不同的虚拟主机进行处理，而对其它 IP 地址的访问，例如 127.0.0.1，才应用前面定义的缺省选项。

**Timeout 300**

Timeout 定义客户程序和服务器连接的超时间隔，超过这个时间间隔（秒）后服务器将断开与客户机的连接。

**KeepAlive On**

在 HTTP 1.0 中，一次连接只能作传输一次 HTTP 请求，而 KeepAlive 参数用于支持 HTTP 1.1 版本的一次连接、多次传输功能，这样就可以在一次连接中传递多个 HTTP 请求。虽然只有较新的浏览器才支持这个功能，但还是打开使用这个选项。

**MaxKeepAliveRequests 100**

MaxKeepAliveRequests 为一次连接可以进行的 HTTP 请求的最大请求次数。将其值设为 0 将支持在一次连接内进行无限次的传输请求。事实上没有客户程序在一次连接中请求太多的页面，通常达不到这个上限就完成连接了。

**KeepAliveTimeout 15**

KeepAliveTimeout 测试一次连接中的多次请求传输之间的时间，如果服务器已经完成了一次请求，但一直没有接收到客户程序的下一次请求，在间隔超过了这个参数设置的值之后，服务器就断开连接。

**MinSpareServers 5 MaxSpareServers 10**

在使用子进程处理 HTTP 请求的 Web 服务器上，由于要首先生成子进程才能处理客户的请求，因此反应时间就有一点延迟。但是，Apache 服务器使用了一个特殊技术来摆脱这个问题，这就是预先生成多个空余的子进程驻留在系统中，一旦有请求出现，就立即使用这些空余的子进程进行处理，这样就不存在生成子进程造成的延迟了。在运行中随着客户请求的增多，启动的子进程会随之增多，但这些服务器副本在处理完一次 HTTP 请求之后并不立即退出，而是停留在计算机中等待下次请求。但是空余的子进程副本不能光增加不减少，太多的空余子进程没有处理任务，也占用服务器的处理能力，因此也要限制空余副本的数量，使其保持一个合适的数量，使得既能

---

及时回应客户请求，又能减少不必要的进程数量。因此就可以使用参数 **MinSpareServers** 来设置最少的空余子进程数量， 以及使用参数 **MaxSpareServers** 来限制最多的空闲子进程数量， 多余的服务器进程副本就会退出。

## **StartServers 5**

**StartServers** 参数就是用来设置 **httpd** 启动时启动的子进程副本数量，这个参数与上面定义的 **MinSpareServers** 和 **MaxSpareServers** 参数相关，都是用于启动空闲子进程以提高服务器的反应速度的。这个参数应该设置为前两个值之间的一个数值。

## **MaxClients 150**

在另一方面，服务器的能力毕竟是有限的，不可能同时处理无限多的连接请求，因此参数 **Maxclients** 就用于规定服务器支持的最多并发访问的客户数。这个参数限制了 **MinSpareServers** 和 **MaxSpareServers** 的设置，它们不应该大于这个参数的设置。

## **MaxRequestsPerChild 30**

使用子进程的方式提供服务的 **Web** 服务，常用的方式是一个子进程为一次连接服务，这样造成的问题就是每次连接都需要生成、退出子进程的系统操作，使得这些额外的处理过程占据了计算机的大量处理能力。因此最好的方式是一个子进程可以为多次连接请求服务，这样就不需要这些生成、退出进程的系统消耗，**Apache** 就采用了这样的方式，一次连接结束后，子进程并不退出，而是停留在系统中等待下一次服务请求，这样就极大的提高了性能。 但由于在处理过程中子进程要不断的申请和释放内存，次数多了就会造成一些内存垃圾，就会影响系统的稳定性，并且影响系统资源的有效利用。因此在一个副本处理过一定次数的请求之后，就可以让这个子进程副本退出，再从原始的 **httpd** 进程中重新复制一个干净的副本，这样就能提高系统的稳定性。这样，每个子进程处理服务请求次数由 **MaxRequestPerChild** 定义。 缺省的设置值为 30，设置为 0 支持每个副本进行无限次的服务处理。

## **Port 80**

**Port** 定义了 **Standalone** 模式下 **httpd** 守护进程使用的端口，标准端口是 80。这个选项只对于以独立方式启动的服务器才有效，对于以 **inetd** 方式启动的服务器则在 **inetd.conf** 中定义使用哪个端口。在 **Unix** 下使用 80 端口需要 **root** 权限，一些管理员为了安全的原因，认为 **httpd** 服务器不可能没有安全漏洞，因而更愿意使用普通用户的权限来启动服务器，这样就不能使用 80 端口及其它小于 1024 的端口，而必须使用大于 1024 的端口来启动 **httpd**，一般情况下 8000 或 8080 也是常用的端口。

## **User nobody Group nogroup**

**User** 和 **Group** 配置是 **Apache** 的安全保证，**Apache** 在打开端口之后，就将其本身设置为这两个选项设置的用户和组权限进行运行，这样就降低了服务器的危险性。这个选项也只用于 **Standalone** 模式，**inetd** 模式在 **inetd.conf** 中指定运行 **Apache** 的用户。一般情况下要为 **Web** 服务设定一个特定的用户和组，同时在这里更改用户和组设置。

## **ServerAdmin you@your.address**

配置文件中应该改变的也许只有 **ServerAdmin**， 这一项用于配置 **WWW** 服务器的管理员的 **email** 地址，这将在 **HTTP** 服务出现错误的条件下返回给浏览器，以便让 **Web** 使用者和管理员联系，报告错误。

## **#ServerName new.host.name**

缺省情况下，并不需要指定这个 **ServerName** 参数，服务器将自动通过名字解析过程来获得自己的名字，但如果服务器的名字解析有问题（通常为反向解析不正确），或者没有正式的 **DNS** 名字，也可以在这里指定 **IP** 地址。当 **ServerName** 设置不正确的时候，服务器不能正常启动。

## **DocumentRoot "/www/"**

它定义这个服务器对外发布的超文本文档存放的路径，客户程序请求的 **URL** 就被映射为这个目录下的网页文件。这个目录下的子目录，以及使用符号连接指出的文件和目录都能被浏览器访问，只是要在 **URL** 上使用同样的相对目录名。 注意，符号连接虽然逻辑上位于根文档目录之下，但实际上可以位于计算机上的任意目录中，因此可以使客户程序能访问那些根文档目录之外的目录，这在增加了灵活性的同时但减少了安全性。**Apache** 在目录的访问控制中提供了 **FollowSymLinks** 选项来打开或关闭支持符号连接的特性。

## **Options Indexes FollowSymLinks**

### **AllowOverride None**

### **Order allow,deny**

### **Allow from all**

这里定义的是系统对外发布文档的目录的访问设置，设置不同的 **AllowOverride** 选项，以定义配置文件中的目录设置和用户目录下的安全控制文件的关系，而 **Options** 选项用于定义该目录的特性。 配置文件和每个目录下的访问控制文件都可以设置访问限制，设置文

---

件是由管理员设置的，而每个目录下的访问控制文件是由目录的属主设置的，因此管理员可以规定目录的属主是否能覆盖系统在设置文件中的设置，这就需要使用 `AllowOverride` 参数进行设置。

### **UserDir public\_html**

当在一台 Linux 上运行 Apache 服务器时，这台计算机上的所有用户都可以有自己的网页路径，像 `http://example.org.cn/~user`，使用波浪符号加上用户名就可以映射到用户自己的网页目录上。映射目录为用户个人主目录下的一个子目录，其名字就用 `UserDir` 这个参数进行定义，缺省为 `public_html`。如果不想为正式的用户提供网页服务，使用 `DISABLED` 作 `UserDir` 的参数即可。