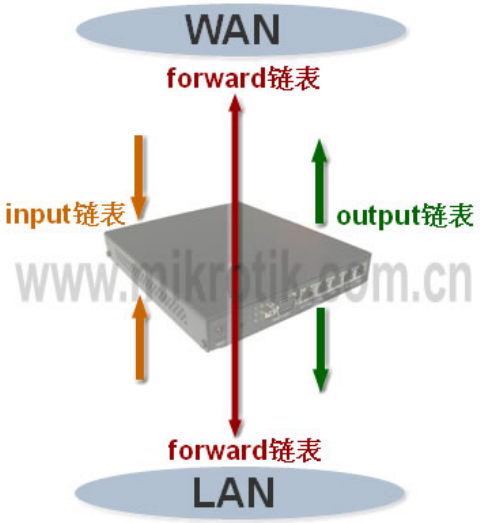
MikroTik RouterOS 应用事例讲解

防火墙规则

下面是三条预先设置好了的 chains，他们是不被能删除的：

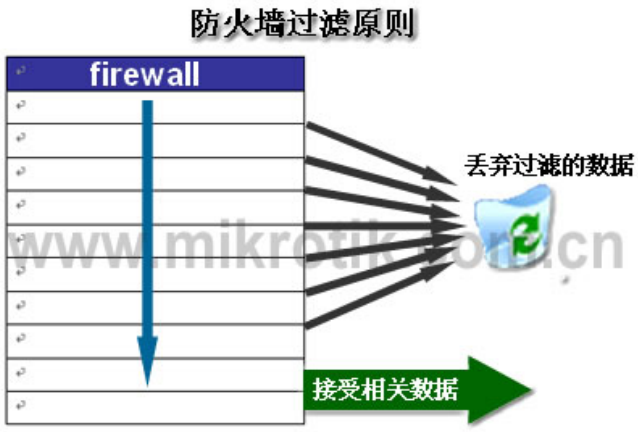
- **input** – 用于处理进入路由器的数据包，即数据包目标 IP 地址是到达路由器一个接口的 IP 地址，经过路由器的数据包不会在 input-chains 处理。
- **forward** – 用于处理通过路由器的数据包
- **output** – 用于处理源于路由器并从其中一个接口出去的数据包。

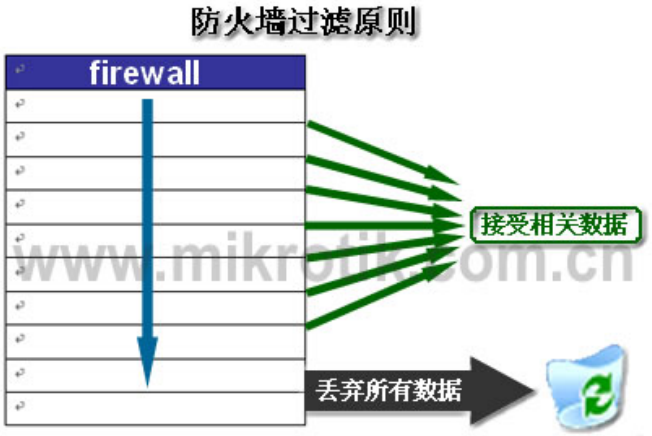
他们具体的区别如下：



当处理一个 chain（数据链），策略是从 chain 列表的顶部从上而下执行的。如果一个数据包满足策略的条件，这时会执行该操作。

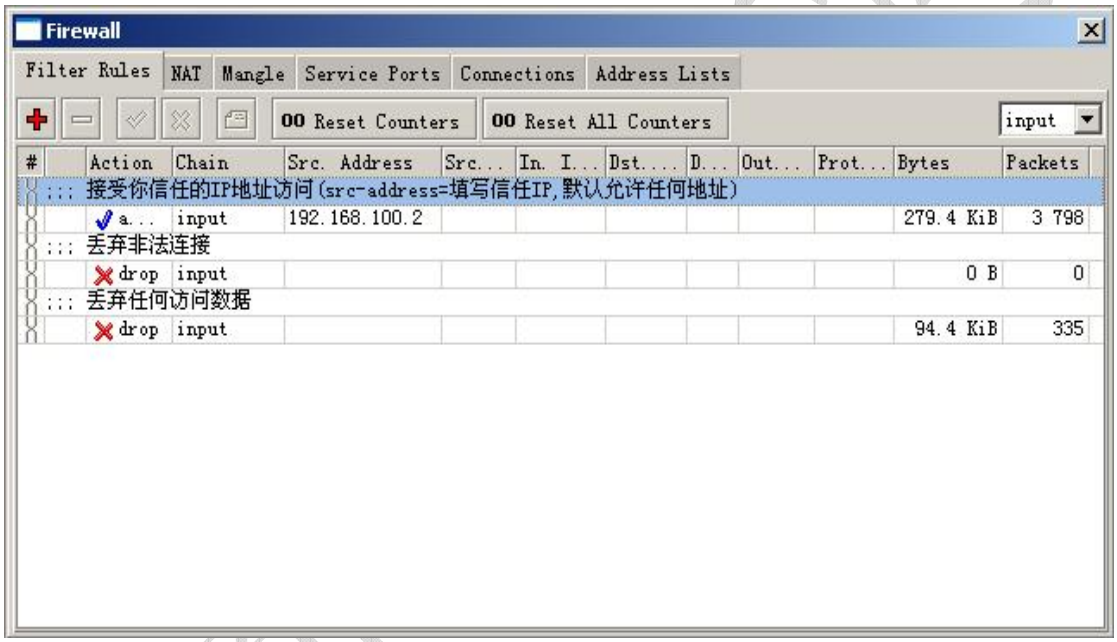
我们来看看防火墙过滤原则：





现在我来看事例中的防火墙规则：

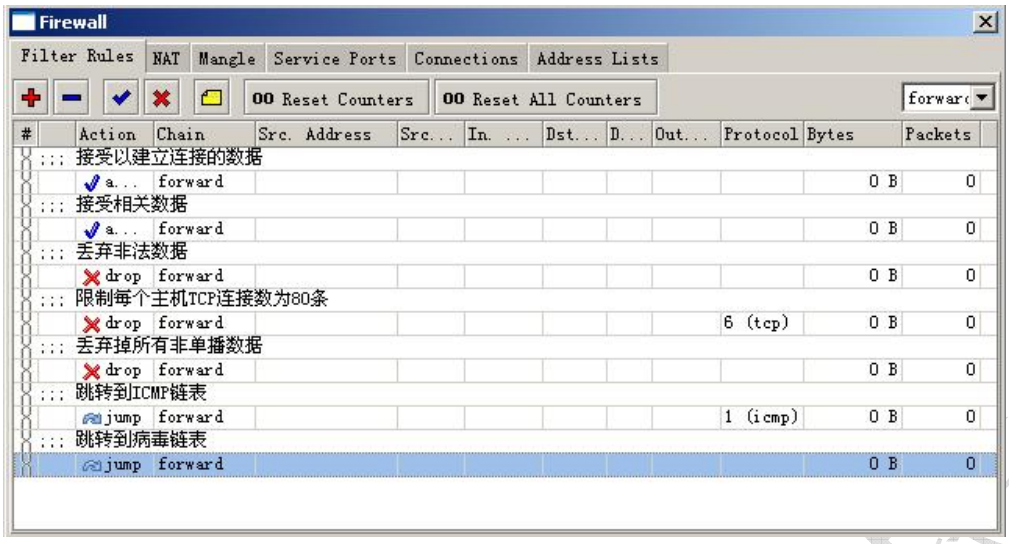
我先从 input 链表开始，这里是对所有访问路由的数据进行过滤和处理：



从 input 链表的第一条开始执行，这里一共有三条规则：

```
0   ;;;  接受你信任的 IP 地址访问(src-address=填写信任 IP,默认允许任何地址)
     chain=input src-address=192.168.100.2 action=accept
1   ;;;  丢弃非法连接
     chain=input connection-state=invalid action=drop
2   ;;;  丢弃任何访问数据
     chain=input action=drop
```

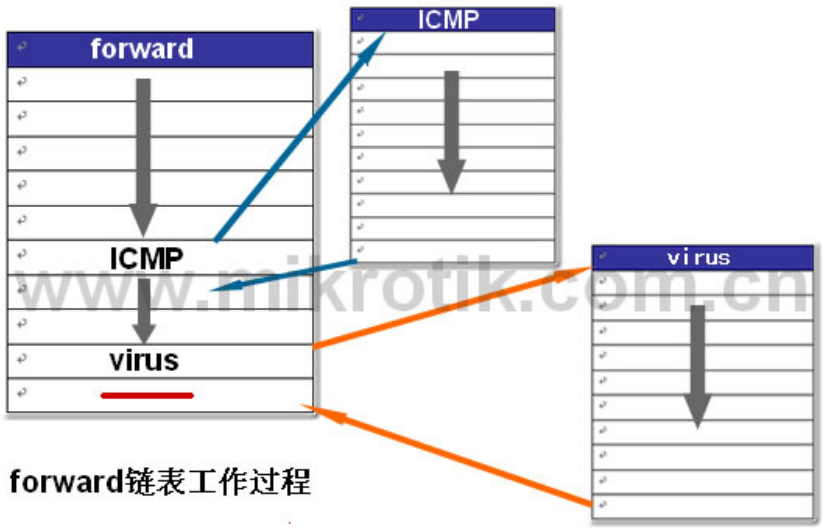
下面是 forward 链表



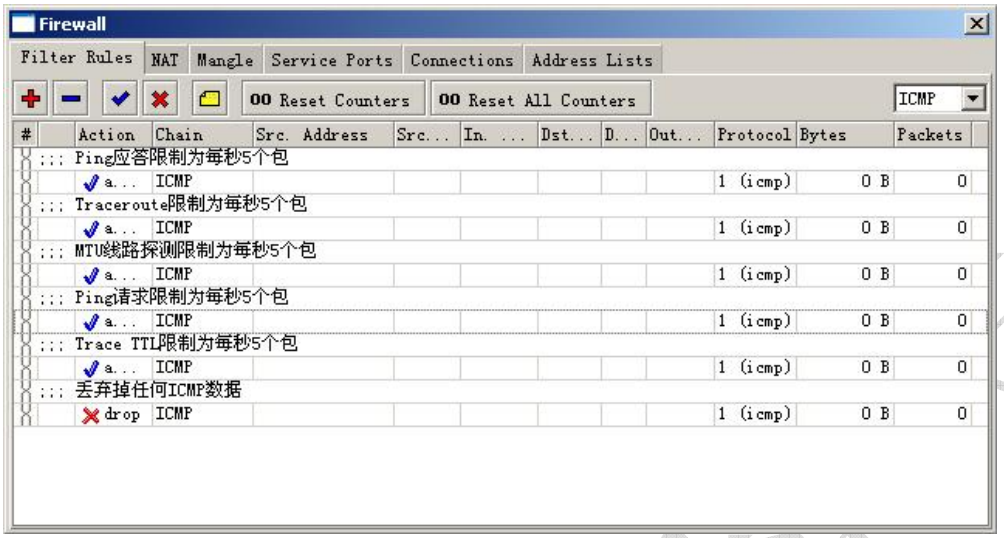
forward 链表，一共有 7 条规则，包括两个跳转到自定义链表 ICMP 和 virus 链表：

```
0  ;;;  接受已建立连接的数据
    chain=forward connection-state=established action=accept
1  ;;;  接受相关数据
    chain=forward connection-state=related action=accept
2  ;;;  丢弃非法数据包
    chain=forward connection-state=invalid action=drop
3   ;;;  限制每个主机 TCP 连接数为 80 条
    chain=forward protocol=tcp connection-limit=80,32 action=drop
4   ;;;  丢弃掉所有非单播数据
    chain=forward src-address-type=!unicast action=drop
5   ;;;  跳转到 ICMP 链表
    chain=forward protocol=icmp action=jump jump-target=ICMP
6   ;;;  跳转到病毒链表
    chain=forward action=jump jump-target=virus
```

forward 工作过程如下：



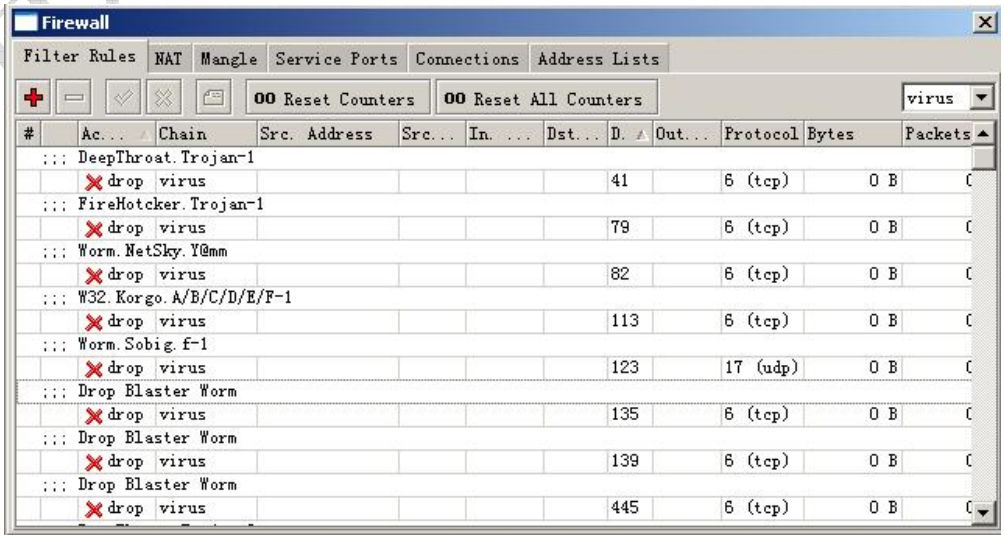
在自定义链表 ICMP 中，是定义所有 ICMP（Internet 控制报文协议），ICMP 经常被认为是 IP 层的一个组成部分。它传递差错报文以及其他需要注意的信息。ICMP 报文通常被 IP 层或更高层协议（TCP 或 UDP）使用。例如：ping、traceroute、trace TTL 等。我们通过 ICMP 链表来过滤所有的 ICMP 协议：



ICMP 链表操作过程：

```
0   ;;;  Ping 应答限制为每秒 5 个包
    chain=ICMP protocol=icmp icmp-options=0:0-255 limit=5,5 action=accept
1   ;;;  Traceroute 限制为每秒 5 个包
    chain=ICMP protocol=icmp icmp-options=3:3 limit=5,5 action=accept
2   ;;;  MTU 线路探测限制为每秒 5 个包
    chain=ICMP protocol=icmp icmp-options=3:4 limit=5,5 action=accept
3   ;;;  Ping 请求限制为每秒 5 个包
    chain=ICMP protocol=icmp icmp-options=8:0-255 limit=5,5 action=accept
4   ;;; Trace TTL 限制为每秒 5 个包
    chain=ICMP protocol=icmp icmp-options=11:0-255 limit=5,5 action=accept
5   ;;;  丢弃掉任何 ICMP 数据
    chain=ICMP protocol=icmp action=drop
```

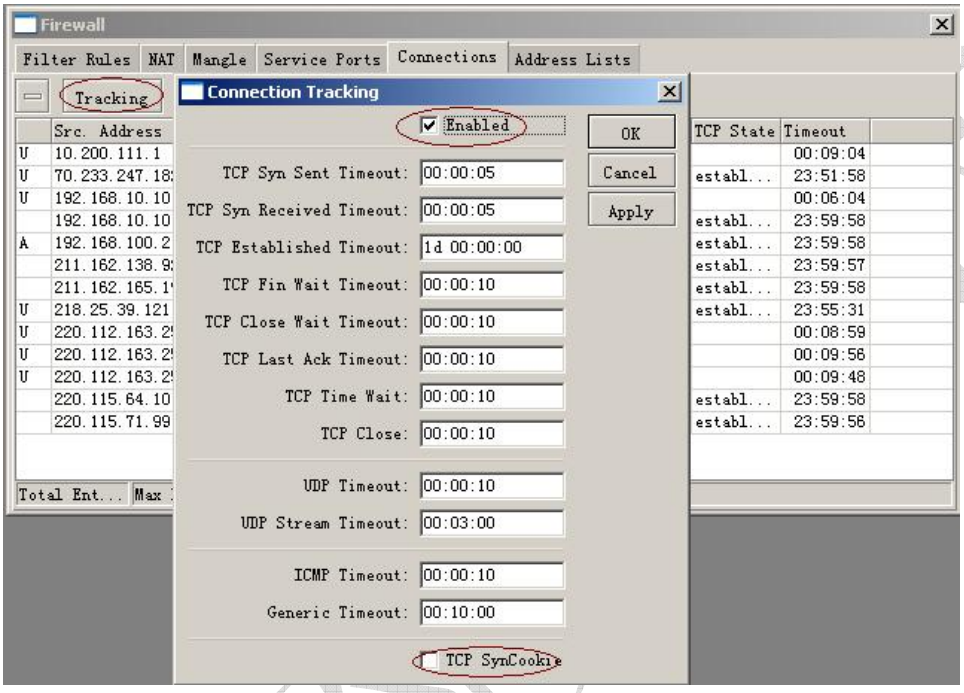
在 virus 链表中过滤常见的病毒，我可以根据需要在该链表中添加新的病毒对他们做过滤：



Tracking 设置

这里我们可以设置是否启用 tracking 连接跟踪，以及 TCP、UDP 和 ICMP 等协议的 timeout 时间，和 TCP-syncookie 设置，RouterOS 在 2.9.16 中增加了 TCP-syncookie 参数。

在使用 NAT 时需要启用 Tracking 连接跟踪，如果你的 RouterOS 没有使用 NAT（如在使用 bridge 模式下），可以选择关闭 tracking，降低系统资源。



SYN Cookie 原理

SYN Flood 是一种非常危险而常见的 DoS 攻击方式。到目前为止，能够有效防范 SYN Flood 攻击的手段并不多，而 SYN Cookie 就是其中最著名的一种

SYN Cookie 是对 TCP 服务器端的三次握手协议作一些修改，专门用来防范 SYN Flood 攻击的一种手段。它的原理是，在 TCP 服务器收到 TCP SYN 包并返回 TCP SYN+ACK 包时，不分配一个专门的数据区，而是根据这个 SYN 包计算出一个 cookie 值。在收到 TCP ACK 包时，TCP 服务器在根据那个 cookie 值检查这个 TCP ACK 包的合法性。如果合法，再分配专门的数据区进行处理未来的 TCP 连接。

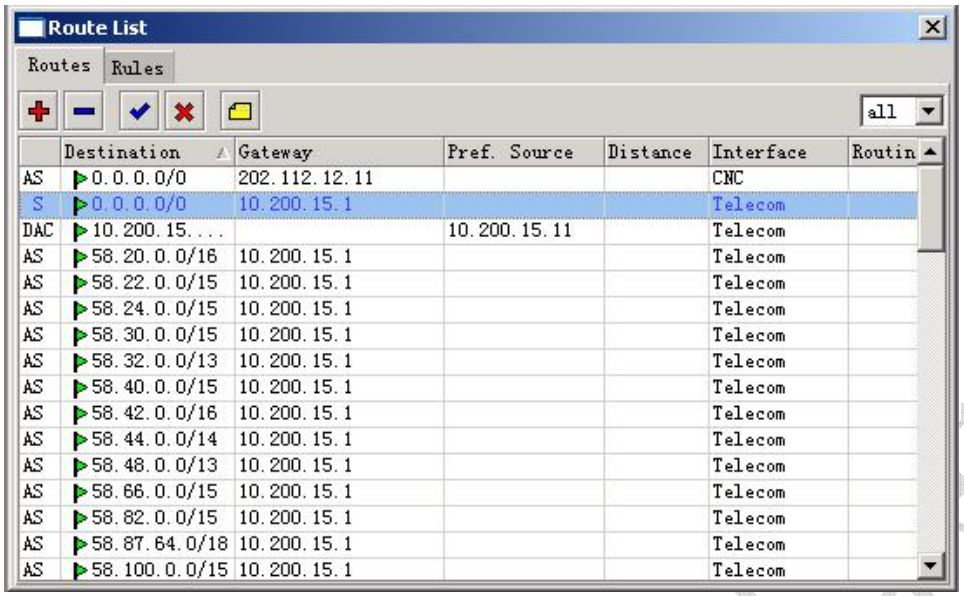
事例操作

如何实现其中一条线路掉线后，自动切换到另一条线路

RouterOS 2.9 中路由规则增加的两点功能：

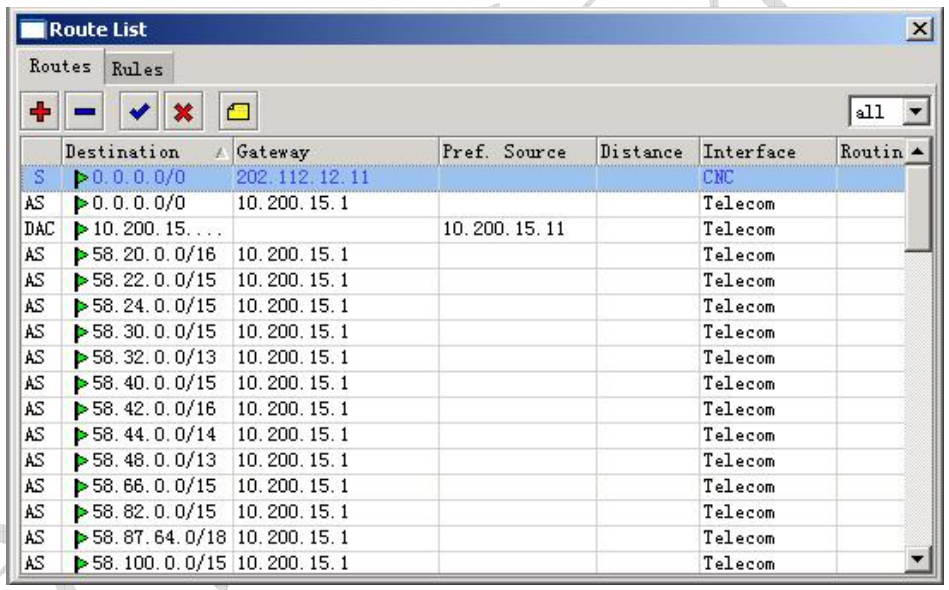
1、在 RouterOS 2.9 路由规则中增加了 check-gateway 的功能，能检测到网关的线路状态，如果网关无法探测到，便认为网关无法连接，会自动禁止访问网关的数据通过，check-gateway 功能的探测时间为 10s 一个周期。

2、在 RouterOS 2.9 中具备了对缺省网关的判断，在 RouterOS 2.9 的任何一个路由表中只能存在一个缺省网关，即到任何目标地址为 0.0.0.0/0，没有做路由标记（routing-mark）的规则，如果存在另一个缺省网关则认为是错误，路由将不予以执行。如下图：



	Destination	Gateway	Pref. Source	Distance	Interface	Routin
AS	0.0.0.0/0	202.112.12.11			CNC	
S	0.0.0.0/0	10.200.15.1			Telecom	
DAC	10.200.15....		10.200.15.11		Telecom	
AS	58.20.0.0/16	10.200.15.1			Telecom	
AS	58.22.0.0/15	10.200.15.1			Telecom	
AS	58.24.0.0/15	10.200.15.1			Telecom	
AS	58.30.0.0/15	10.200.15.1			Telecom	
AS	58.32.0.0/13	10.200.15.1			Telecom	
AS	58.40.0.0/15	10.200.15.1			Telecom	
AS	58.42.0.0/16	10.200.15.1			Telecom	
AS	58.44.0.0/14	10.200.15.1			Telecom	
AS	58.48.0.0/13	10.200.15.1			Telecom	
AS	58.66.0.0/15	10.200.15.1			Telecom	
AS	58.82.0.0/15	10.200.15.1			Telecom	
AS	58.87.64.0/18	10.200.15.1			Telecom	
AS	58.100.0.0/15	10.200.15.1			Telecom	

从上图我们可以看到，所有访问电信的 IP 段从 10.200.15.1 出去，其他的数据走网通的缺省网关出去，在我们可以这些网关的前缀都为“AS”，即确定的静态路由，而在第二排可以看到蓝色一行，他也是一个缺省网关，但因为一个路由表中只能存在一个缺省网关，所有前缀为“S”即静态但不确定的网关，被认为位非法的。如果当 202.112.12.12.11 网关断线，则 10.200.15.1 会自动启用，变为缺省路由，实现现在的切换，如下：

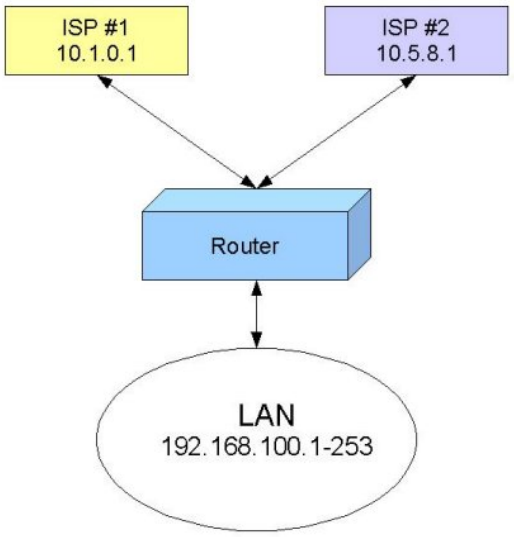


	Destination	Gateway	Pref. Source	Distance	Interface	Routin
S	0.0.0.0/0	202.112.12.11			CNC	
AS	0.0.0.0/0	10.200.15.1			Telecom	
DAC	10.200.15....		10.200.15.11		Telecom	
AS	58.20.0.0/16	10.200.15.1			Telecom	
AS	58.22.0.0/15	10.200.15.1			Telecom	
AS	58.24.0.0/15	10.200.15.1			Telecom	
AS	58.30.0.0/15	10.200.15.1			Telecom	
AS	58.32.0.0/13	10.200.15.1			Telecom	
AS	58.40.0.0/15	10.200.15.1			Telecom	
AS	58.42.0.0/16	10.200.15.1			Telecom	
AS	58.44.0.0/14	10.200.15.1			Telecom	
AS	58.48.0.0/13	10.200.15.1			Telecom	
AS	58.66.0.0/15	10.200.15.1			Telecom	
AS	58.82.0.0/15	10.200.15.1			Telecom	
AS	58.87.64.0/18	10.200.15.1			Telecom	
AS	58.100.0.0/15	10.200.15.1			Telecom	

当 202.112.12.11 断线后，check-gateway 在 10s 一个周期后探测到，并将 10.200.15.11 设置为缺省路由，如果 202.112.12.11 正常后，系统也将会将 202.112.12.11 设置为缺省路由，因为他是先于 10.200.15.1 添加入路由表中。

双线应用案例

这是一个典型的通过一个路由器并使用两条 ISP 线路接入的环境（比如都是两条电线的 ADSL 或者 LAN 接入）：



当然，你可以选择负载均衡！这里有多种方法可以选择，只是根据你的环境，选择最适合你解决方案。

基于用户端 IP 地址的策略路由

如果你有很多的主机地址，你可以通过 IP 地址将他们分组。这时，指定源 IP 地址，发送的传输通过 ISP1 或者 ISP2 的网关出去。 让我们假设终端电脑的网络地址段为 192.168.100.0/24，IP 分配如下：

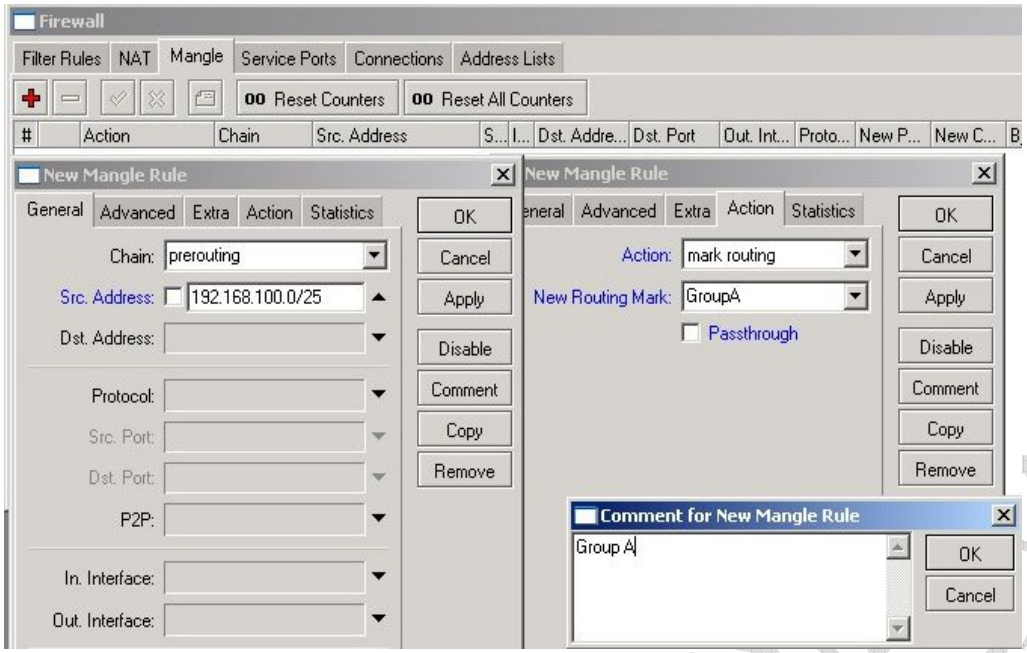
- 192.168.100.1-127 分配到 A 组
- 192.168.100.128-253 分配到 B 组
- 192.168.100.254 路由器本地 IP 地址（即内网的网关）

现在，我们通过子网划分的方式，将终端电脑进行分组：

- A 组为 192.168.100.0/25，地址范围：192.168.100.0-127
- B 组为 192.168.100.128/25，地址范围：192.168.100.128-255

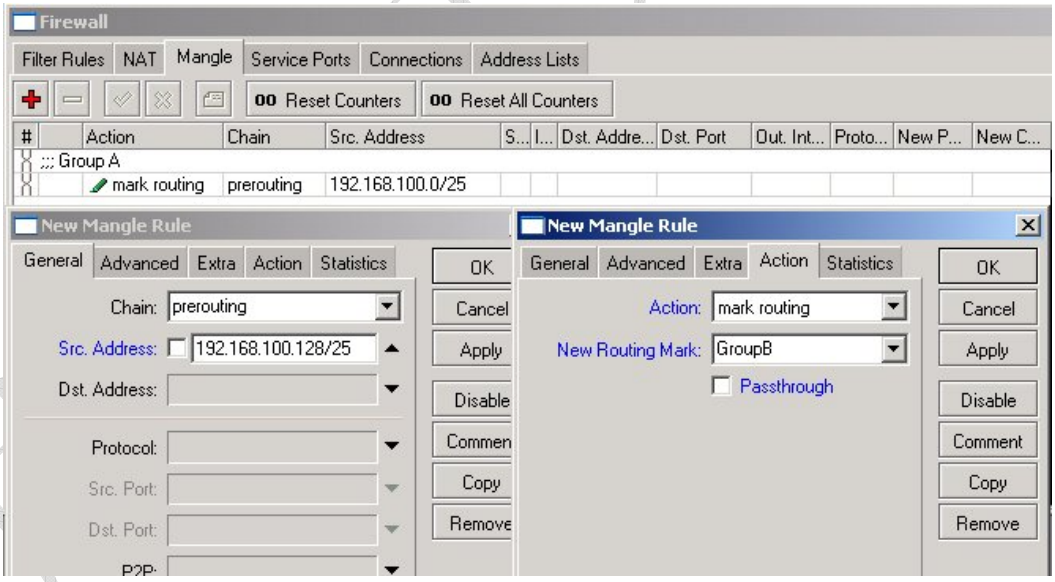
如果你不能理解，请你查阅 TCP/IP 的相关教材或通过网上查找相关的子网划分资料！我们需要添加两个 ip firewall mangle 的规则，标记来至 A 组和 B 组终端电脑的数据包。

定义 A 组：
链表为 chain=prerouting，源地址：src-address=192.168.100.0/25
操作为 Action=mark routing 并定义新的路由标记 GroupA.



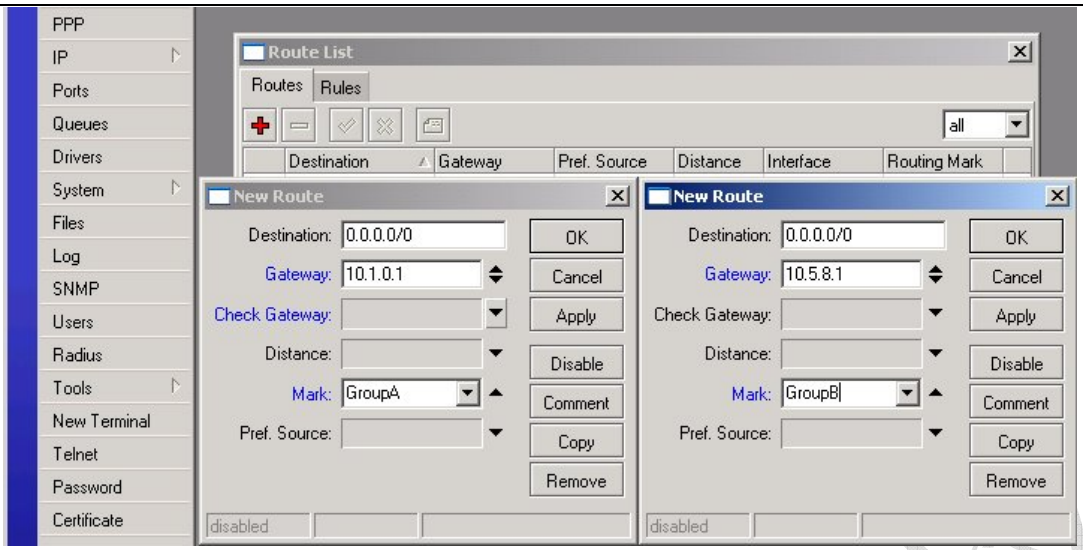
最好做一个注释，以便以后便于你自己或者别人查看和处理。

定义 B 组：
链表为 chain=prerouting，源地址：src-address=192.168.100.128/25
操作为 Action=mark routing 并定义新的路由标记 GroupB



所有来至终端电脑的 IP 传输都通过路由标记为 GroupA 或者 GroupB。这样我们可以标记到路由表中（routing table）。

下面，我们需要定义两个默认路给相应的路由标记和网关：



到这里，如果你没有对路由器做 NAT 的伪装，请在**/ip firewall nat** 里添加 src- Address=**192.168.100.0/24** action=**masquerade,** 在终端电脑上测试一下跟踪路由是否正确定义两个分组的默认路由：

A 组测试如下情况：

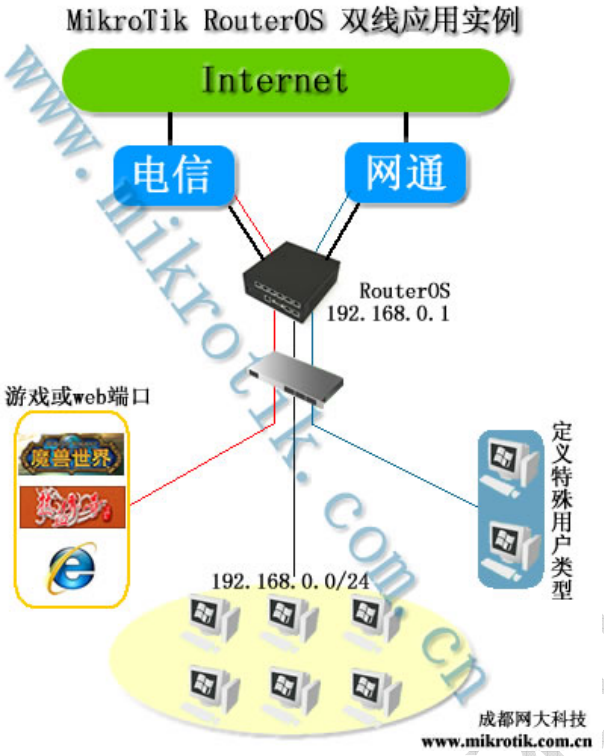
```
C:\>tracert -d 8.8.8.8
Tracing route to 8.8.8.8 over a maximum of 30 hops

  1    2 ms    2 ms    2 ms  192.168.100.254
  2   10 ms    4 ms    3 ms  10.1.0.1
  ...
```

B 组测试如下情况：

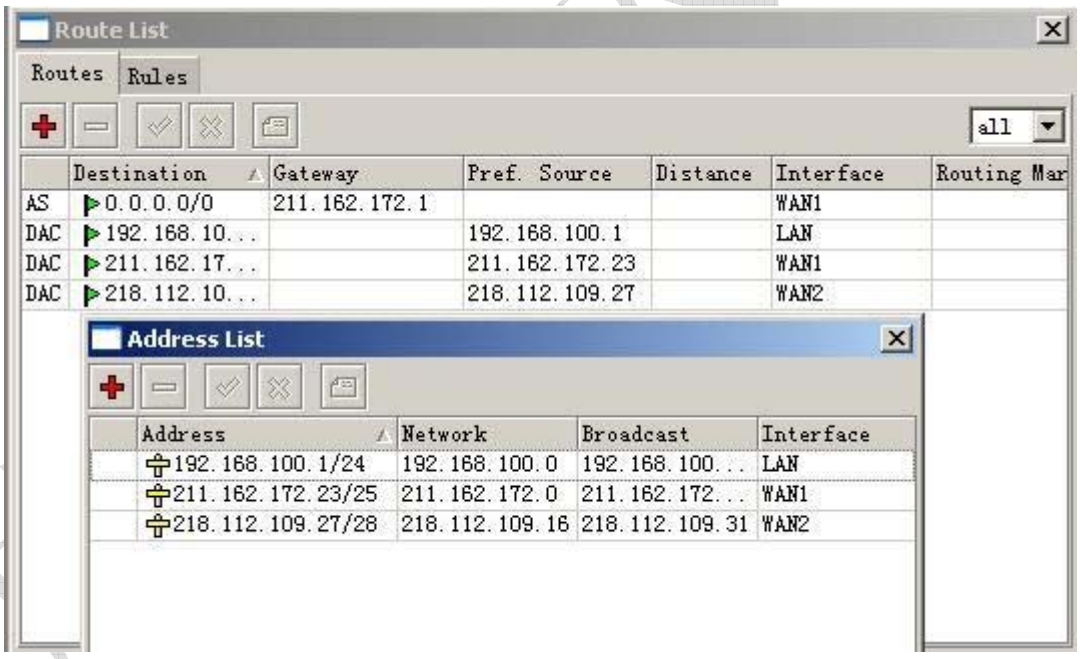
```
C:\>tracert -d 8.8.8.8
Tracing route to 8.8.8.8 over a maximum of 30 hops

  1    2 ms    2 ms    2 ms  192.168.100.254
  2   10 ms    4 ms    3 ms  10.5.8.1
  ...
```

如何做端口的策略路由：

MikroTik RouterOS 可以支持多种策略路由，如我们常见的源地址、目标地址，同样支持端口的策略路由，多种规则可以根据用户情况配合使用，如下图：

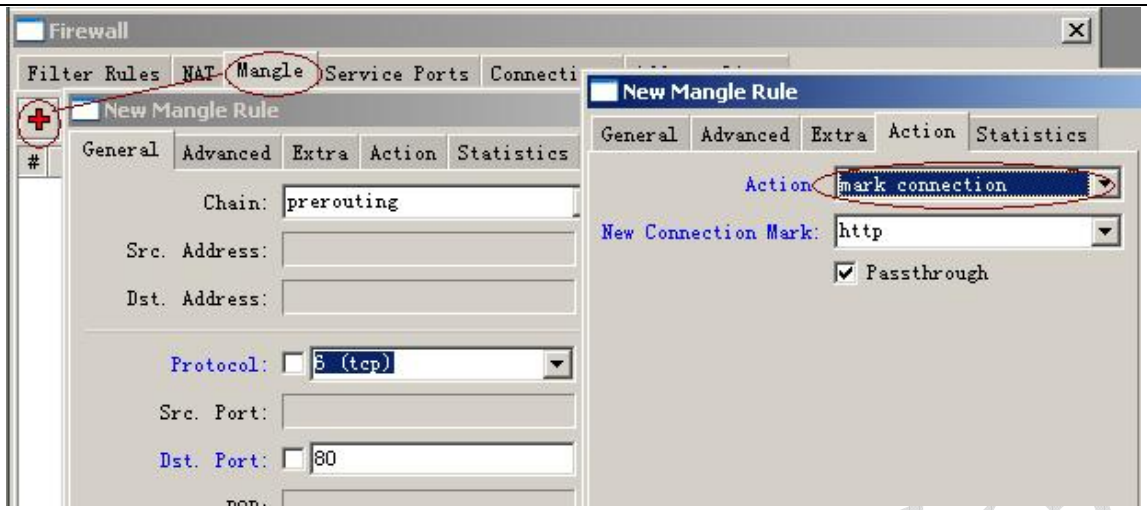


现在我们通过下面的图解一步步实现端口的策略路由：

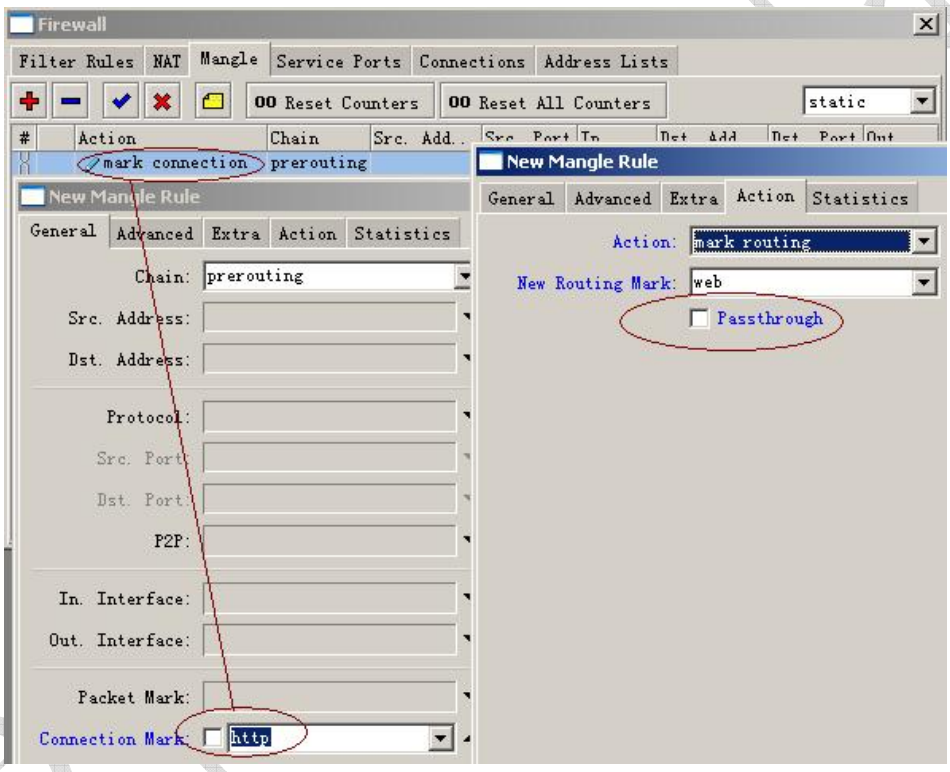


我们有两个 ISP 接入的线路，一个是 WAN1： 211.162.172.23，一个是 WAN2： 218.112.109.27（地址为假设），我们让默认的数据通过 WAN1，让访问网页的数据通过 WAN2。

现在我们定义访问网页的端口，访问网页的端口是 TCP 80 端口，我们进入/ip firewall mangle 中做数据标记

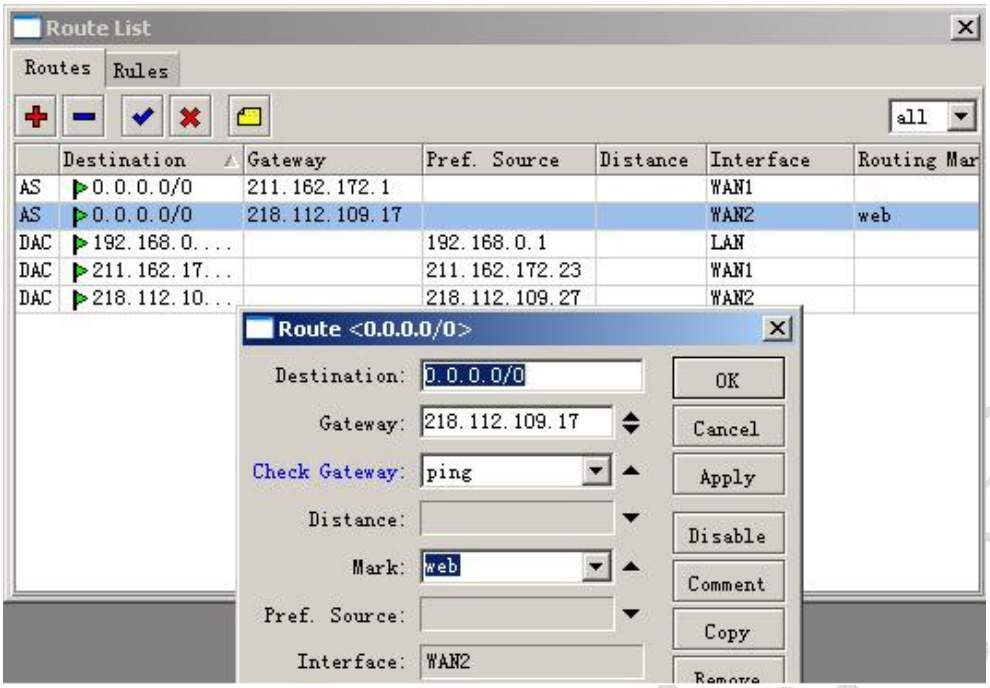


首先我们标记 80 端口的连接，标记名为“http”然后我们从这些连接中提取我们想要的数据：

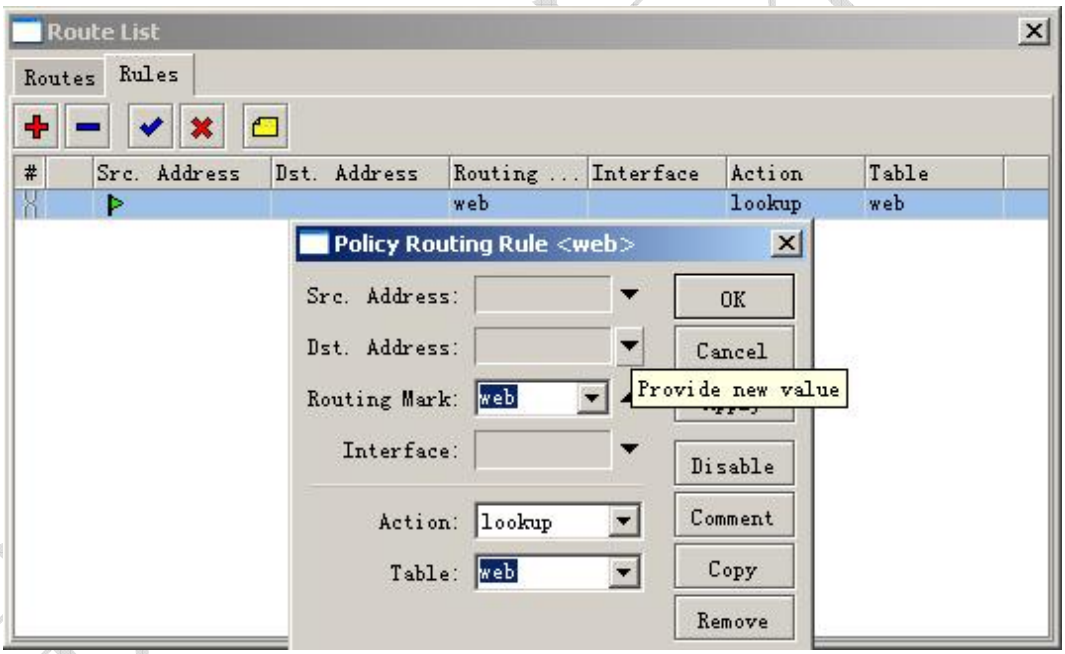


之后我们从标记中提取路由标记，命名为“web”，因为我们在前面的连接标记中做过了 passthrough 的设置，在这里就不用在重复设置。

然后我们进入/ip route，配置路由我们让标记好的 80 端口路由去 WAN2 的线路：



在这里，我们也可以通过/ip route rule 来定义端口的规则：

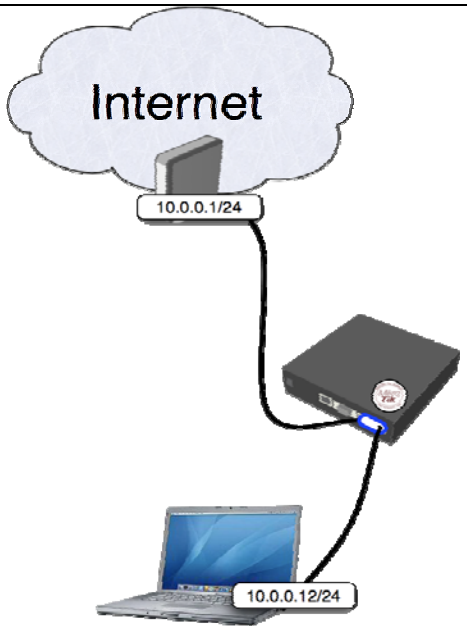


让定义的 web 标记在一次回到 web 路由表中去查找网关。

透明传输整形器（Transparent Traffic Shaper）

这个事例将介绍如何配置一个透明传输整形器。透明整形器是建立在一个桥上，能区分和优先考虑什么样的传输通过。

现在考虑下面的网络拓扑：



在这里配置一组队列限制，一个客户端的总的通过量和三个子队列（HTTP、P2P 和其他的传输数据），HTTP 传输将优先在其他传输之上。

快速配置

配置代码（可以复制到 MikroTik RouterOS 执行）：

```
/ interface bridge
add name="bridge1"
/ interface bridge port
add interface=ether2 bridge=bridge1
add interface=ether3 bridge=bridge1

/ ip firewall mangle
add chain=prerouting protocol=tcp dst-port=80 action=mark-connection \
    new-connection-mark=http_conn passthrough=yes
add chain=prerouting connection-mark=http_conn action=mark-packet \
    new-packet-mark=http passthrough=no
add chain=prerouting p2p=all-p2p action=mark-connection \
    new-connection-mark=p2p_conn passthrough=yes
add chain=prerouting connection-mark=p2p_conn action=mark-packet \
    new-packet-mark=p2p passthrough=no
add chain=prerouting action=mark-connection new-connection-mark=other_conn \
    passthrough=yes
add chain=prerouting connection-mark=other_conn action=mark-packet \
    new-packet-mark=other passthrough=no

/ queue simple
add name="main" target-addresses=10.0.0.12/32 max-limit=256000/512000
add name="http" parent=main packet-marks=http max-limit=240000/500000
add name="p2p" parent=main packet-marks=p2p max-limit=64000/64000
add name="other" parent=main packet-marks=other max-limit=128000/128000
```

分析

下面将解释每段代码的具体实现：

Bridge

```
/ interface bridge
add name="bridge1"
/ interface bridge port
add interface=ether2 bridge=bridge1
add interface=ether3 bridge=bridge1
```

建立一个新的 bridge 接口，并分配 2 个以太网卡给他：这样可以在两个网络间实现透明桥的功能

Mangle

```
/ ip firewall mangle
add chain=prerouting protocol=tcp dst-port=80 action=mark-connection \
   new-connection-mark=http_conn passthrough=yes
add chain=prerouting connection-mark=http_conn action=mark-packet \
   new-packet-mark=http passthrough=no
```

所有符合 TCP 端口 80 及 HTTP 协议传输的数据，将标记为数据包标记为 http ，注意：第一条规则设置为 passthrough=yes，第二条为 passthrough=no.

```
/ ip firewall mangle
add chain=prerouting p2p=all-p2p action=mark-connection \
   new-connection-mark=p2p_conn passthrough=yes
add chain=prerouting connection-mark=p2p_conn action=mark-packet \
   new-packet-mark=p2p passthrough=no
add chain=prerouting action=mark-connection new-connection-mark=other_conn \
   passthrough=yes
add chain=prerouting connection-mark=other_conn action=mark-packet \
   new-packet-mark=other passthrough=no
```

同上面所述，P2P 传输被标记为数据包标记 p2p 并将剩下的传输标记为 other.

Queues

```
/ queue simple
add name="main" target-addresses=10.0.0.12/32 max-limit=256000/512000
```

创建一个队列，限制所有聪客户端来的流量传输为(指定客户端的 target-address)256k/512k.

```
/ queue simple
```

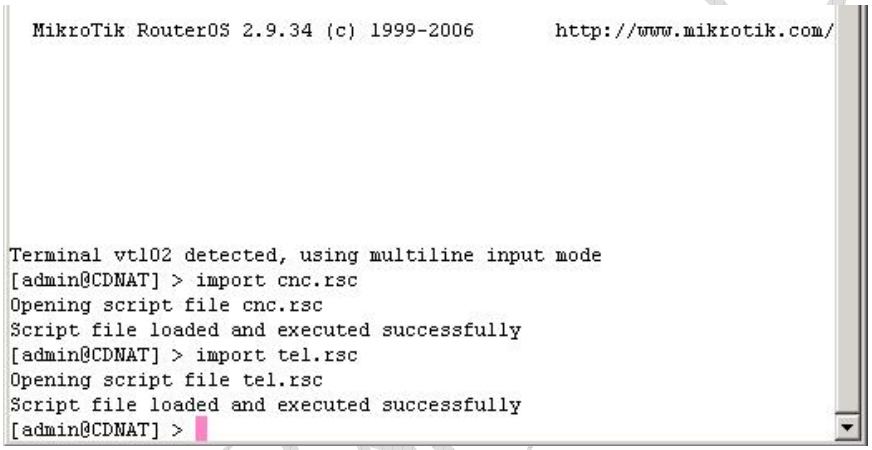
```
add name="http" parent=main packet-marks=http max-limit=240000/500000
add name="p2p" parent=main packet-marks=p2p max-limit=64000/64000
add name="other" parent=main packet-marks=other max-limit=128000/128000
```

所有子队列排列入 main 父系，因此所有的带宽流量不会超过指定的 main 队列注意:http 队列优先级高于其他队列,级 HTTP 流量将优先考虑。

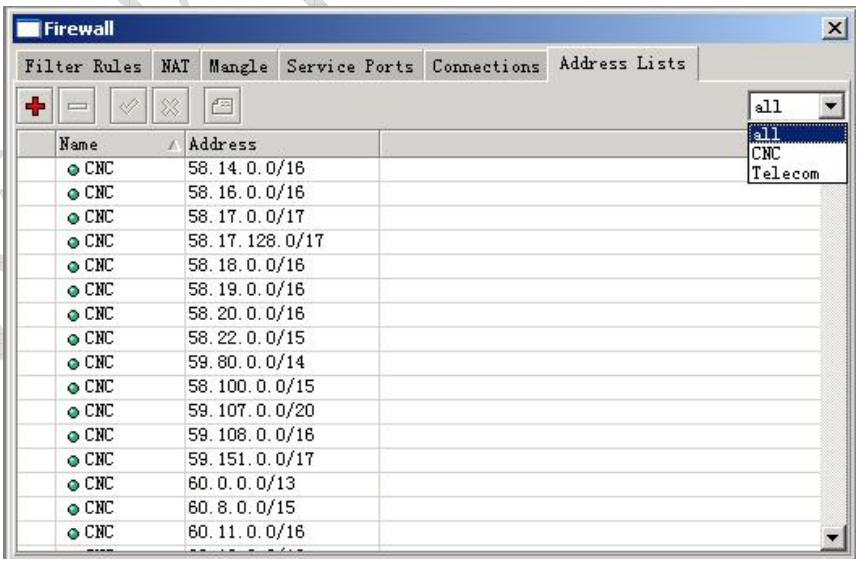
如果配置到电信网通的流量控制

对于电信和网通的IP地址段是已知，那么我们可以通过通过地址标记来实现对这些地址的流量控制，首先我们将电信和网通的地址段导入RouterOS的address-list中（可以在www.mikrotik.com.cn下载到）

通过 import 命令，导入地址列表：

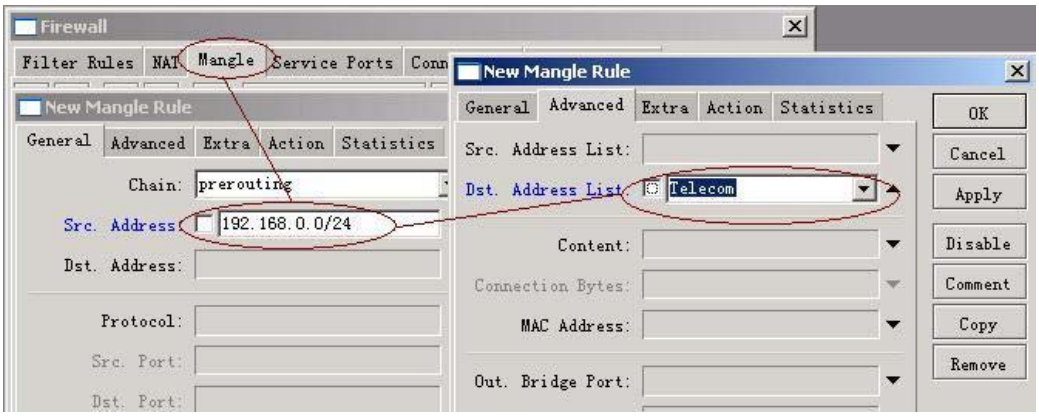


导入后我们可以在/ip firewall address-list 中找到：

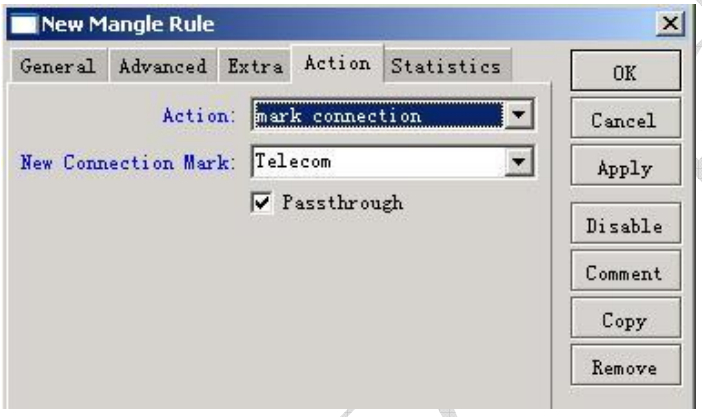


配置数据标记 mangle

进入/ip firewall mangle 设置，这里我们定义访问电信的流量控制，我们的内网地址段为 192.168.0.0/24，所有这里我们配置源地址 src-address=192.168.0.0/24。在 mangle 中先标记连接，然后在从连接中提取数据包：



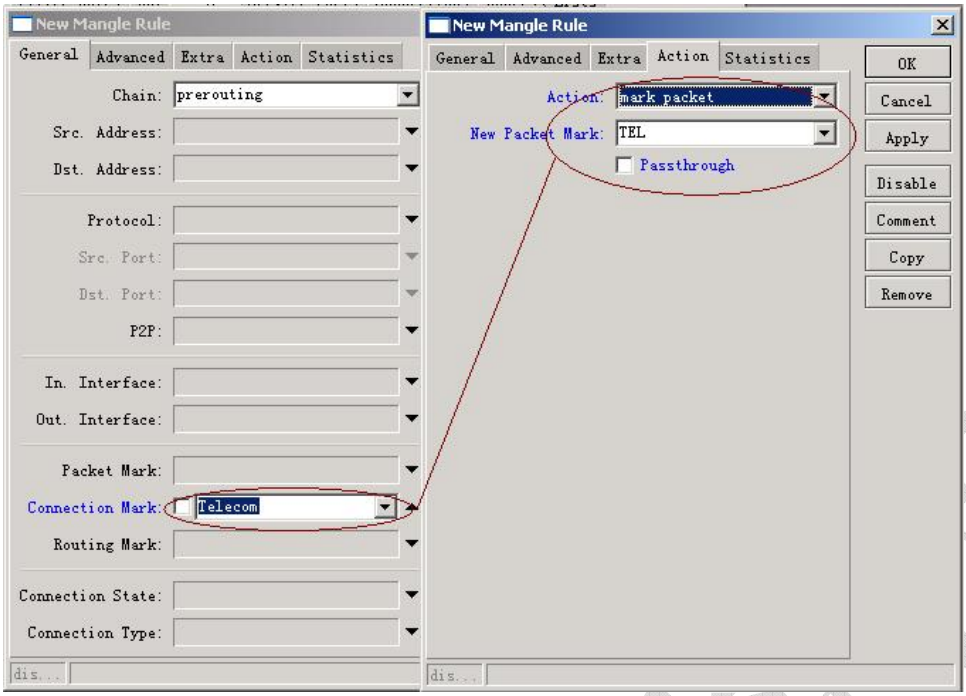
定义标记类型：



源代码：

```
/ ip firewall mangle
add chain=prerouting src-address=192.168.0.0/24 dst-address-list=Telecom
action=mark-connection  new-connection-mark=Telecom passthrough=yes comment=""
disabled=no
```

现在从标记的连接 Telecom 中提取数据包：

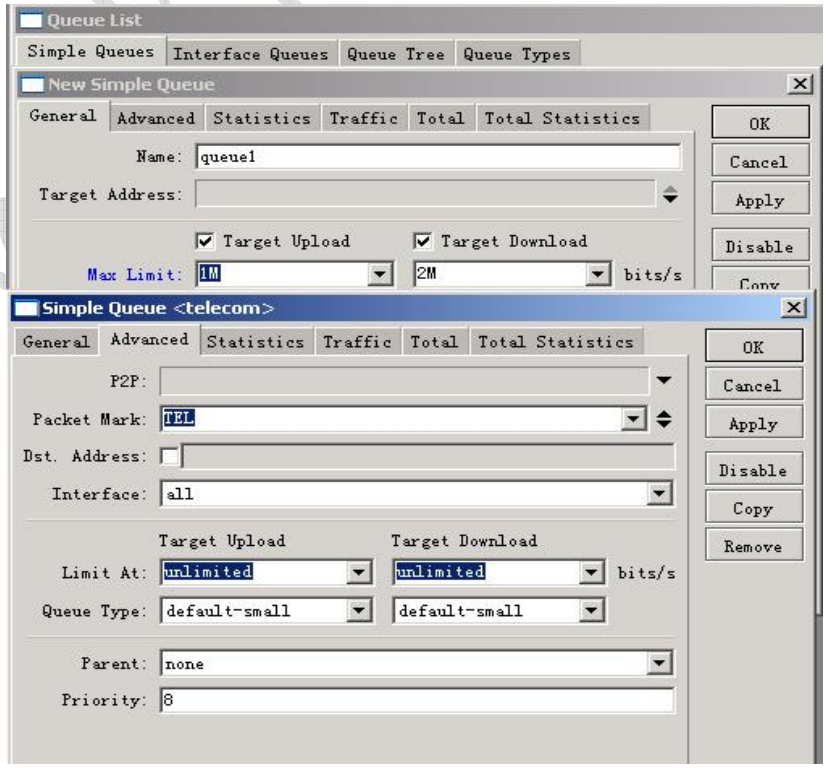


源代码：

```
/ ip firewall mangle
add chain=prerouting connection-mark=Telecom action=mark-packet    new-packet-mark=TEL
passthrough=no comment="" disabled=no
```

配置 simple queue

现在我们进入/queue simple 对列中配置流控规则，在这里我们把到电信的带宽控制在 1M 上行和 2M 下行



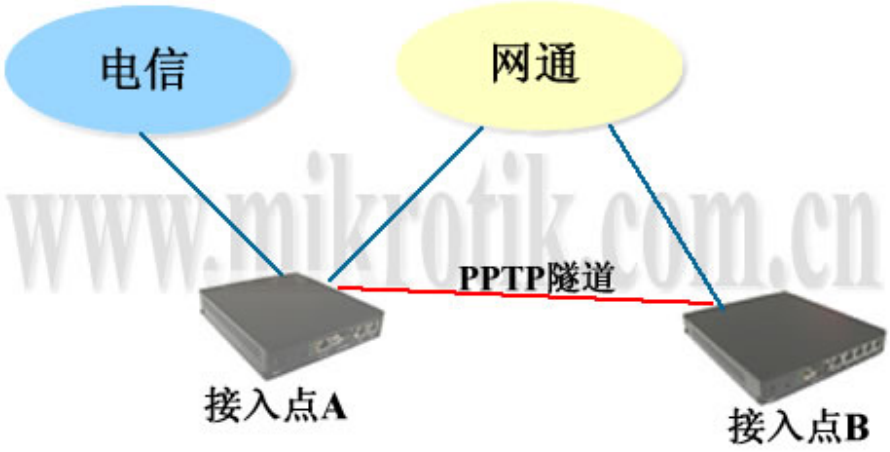
源代码：

```
/ queue simple
add name="telecom" dst-address=0.0.0.0/0 interface=all parent=none packet-marks=TEL
direction=both priority=8 queue=default-small/default-small limit-at=0/0
max-limit=1000000/2000000 total-queue=default-small disabled=no
```

这样对电信的带宽控制便完成，控制网通带宽同样的

PPTP 借线操作

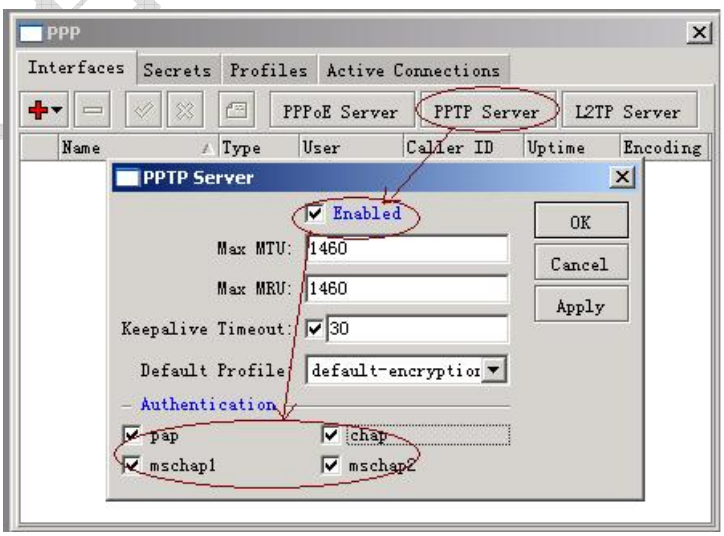
假设一个接入点 A 有电信和网通两条线路，并做了以网通为主，电信为静态路由策略设置。而另一个接入点 B 接入了网通的线路，并且想通过 PPTP 隧道的方式借用接入点 A 的电信线路，现在看下面的图例



根据上面的案例，接入点 A 和 B 他们都是共同使用了网通的线路，这里网通两个点之间的延迟小于 10ms，网络延迟小才能保证足够的网速给 B 做电信的访问。首先建立从接入点 B 到 A 的 PPTP 隧道，我们在接入点 A 设置 PPTP 服务器，在接入点 B 设置客户端。这里接入点 A 的网通 IP 地址为 202.112.12.10，B 网通地址为 202.112.12.12。

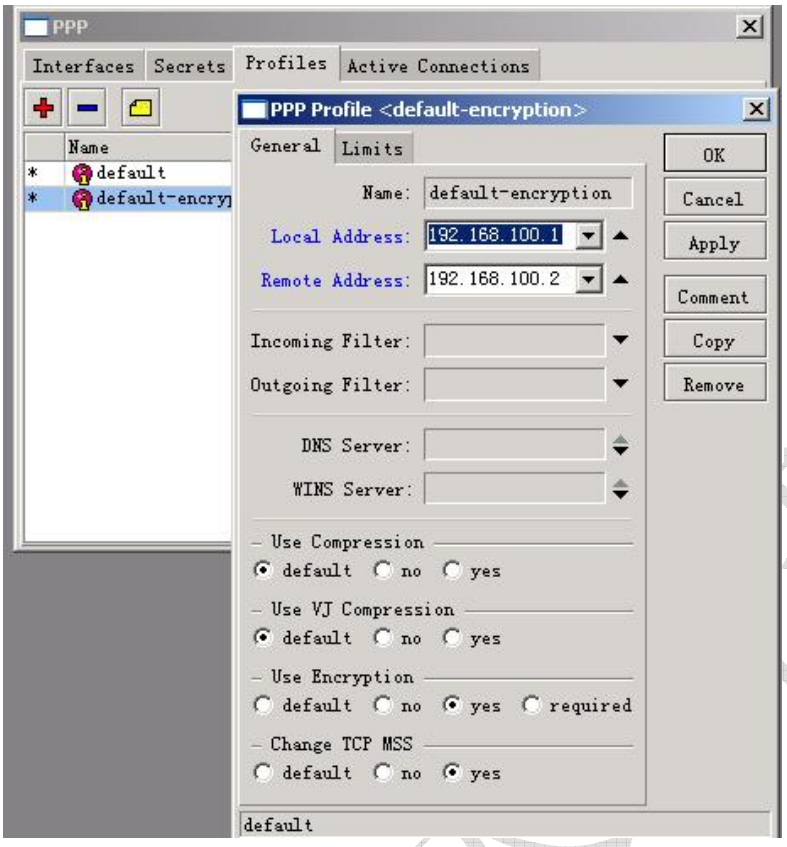
配置 PPPTP-Server

在接入点 A 启用 PPTP-Server，并设置密码传输的加密类型：



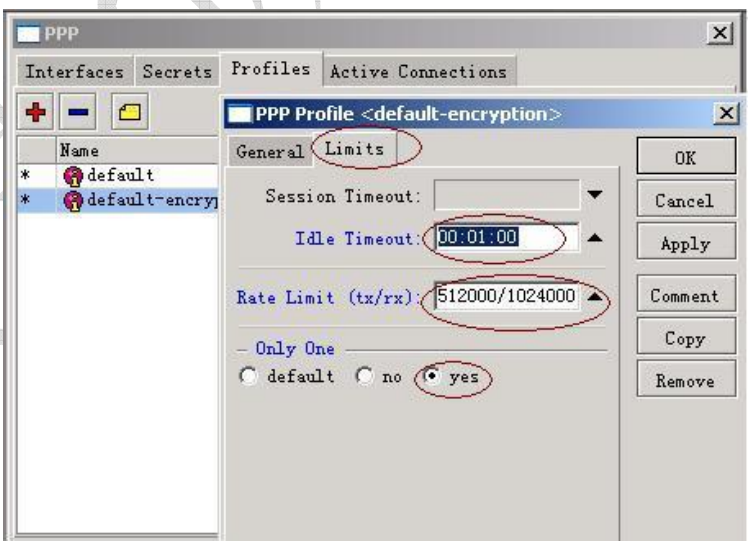
在这里 Default-Profile 我们采用 default-encryption，同样你也可以在 PPTP-Server 的 profiles 中创建自己的规则。Keepalive-Timeout 是 PPTP-Server 主动使用 ICMP 协议探测客户端是否在线，如果客户端使用了防火墙或禁止 ICMP 探测，那无法探测到客户端，Server 就会主动断开该客户端的连接，这个设置需要用户自己根据网络情况判断。

设置 Profile 定义客户和主机的访问地址：



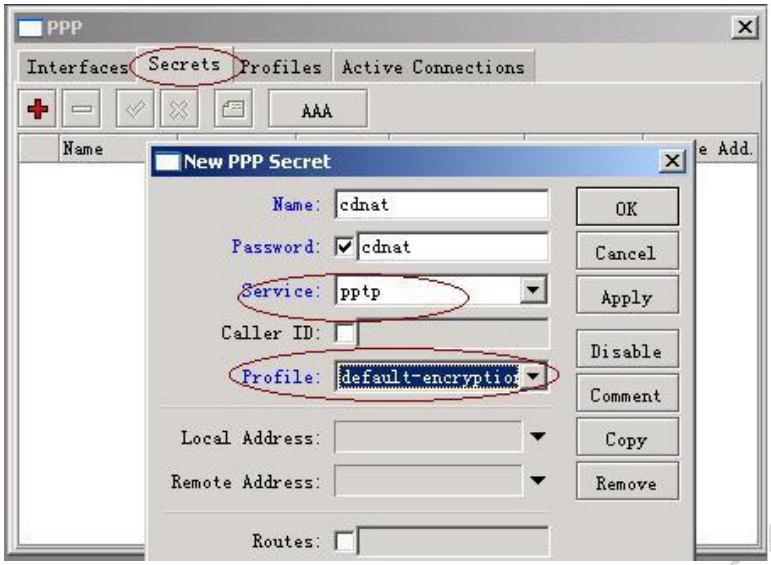
在这里我们给 PPTP-Server 分配的 IP 地址为 192.168.100.1(local-address) ，给客户端分配的地址为 192.168.100.2(remote-address) 。分配 IP 地址也可以通过账号设置 Secrets 进行，在这里我们只有一个客户端所有可以直接通过 profile 中的规则设置，如果有多个客户端也可以通过/ip pool 中的地址池做 DHCP 的分配。

配置 limit 参数：



在 limit 参数中，我们可以看到 idle-timeout，这个是客户端在没有流量超过 1 分钟后，就断开客户端。Rate-limit 是对该类用户的流量控制这里设置的上行为 512K，下行 1M 的带宽。最后是 only-one 该账户是否为唯一，这里设置为 yes。

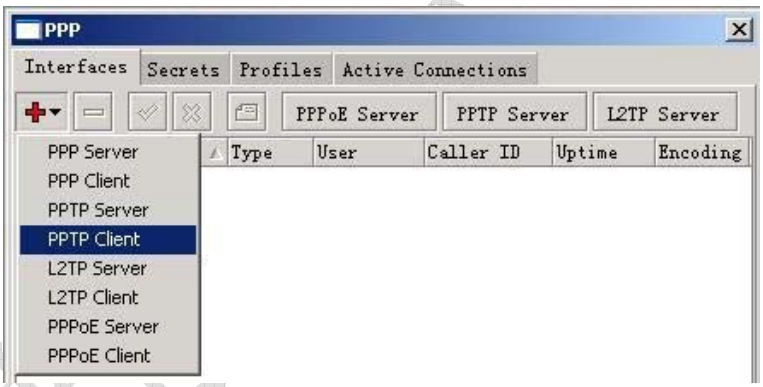
设置客户端的账号密码：



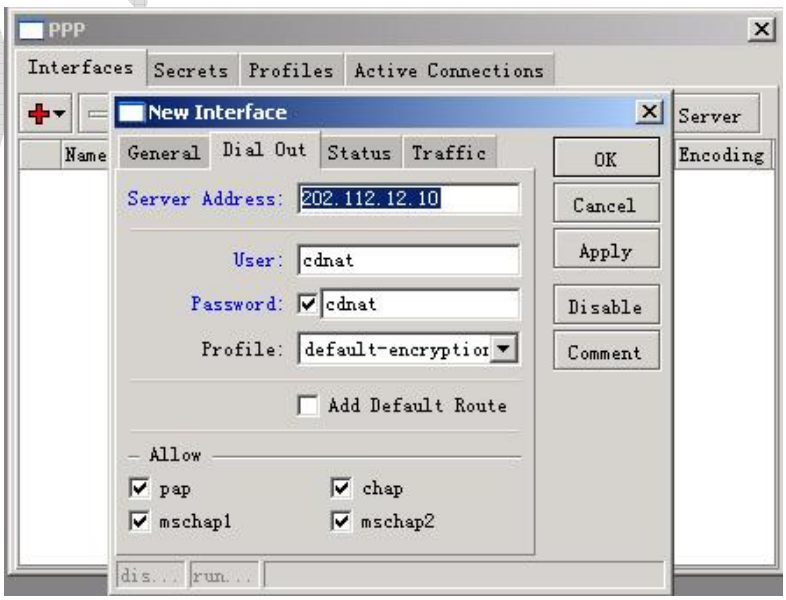
进入 secret 设置账号和密码以及相关信息，设置好 name 和 password 后，选择 service 服务类型为 pptp，profile 规则为 default-encryption。这样 PPTP-Server 就已经设置完成。

配置 PPTP-Client

完成 PPTP 服务设置后，现在开始设置接入点 B 的 PPTP-Client，进入 PPP 选项添加 PPTP-Client：



进入 dial-out 设置 PPTP 拨号信息，在 server-address 的地址为 202.112.12.10 级接入点 A 的网通地址：

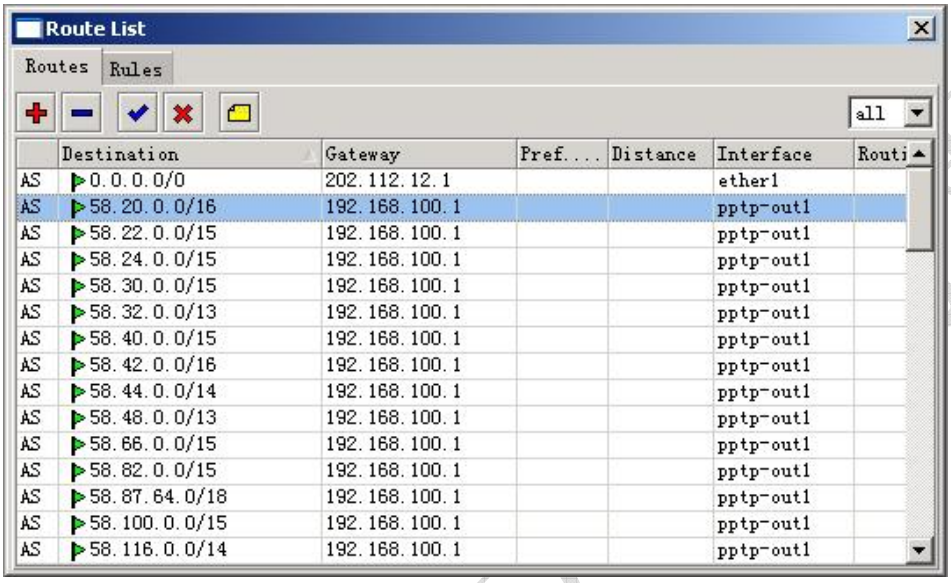


设置账号和密码分别为 cdnat，设置完成后，便可以与接入点 A 的 PPTP-Server 连接。

路由配置

在这里接点 A 和 B 都做了 IP 地址的 NAT 转换，且接点 A 已经做了电信的静态路由规则，即 A 点可以实现访问网通和电信的分流，在 A 点不需要在做任何设置。B 点就需要指定通过 AB 两点间的 PPTP 隧道到电信的线路，他指定的网关为 A 点的 PPTP 的 IP 地址（192.168.100.1）

设置电信访问的网关：



	Destination	Gateway	Pref....	Distance	Interface	Routi
AS	0.0.0.0/0	202.112.12.1			ether1	
AS	58.20.0.0/16	192.168.100.1			pptp-out1	
AS	58.22.0.0/15	192.168.100.1			pptp-out1	
AS	58.24.0.0/15	192.168.100.1			pptp-out1	
AS	58.30.0.0/15	192.168.100.1			pptp-out1	
AS	58.32.0.0/13	192.168.100.1			pptp-out1	
AS	58.40.0.0/15	192.168.100.1			pptp-out1	
AS	58.42.0.0/16	192.168.100.1			pptp-out1	
AS	58.44.0.0/14	192.168.100.1			pptp-out1	
AS	58.48.0.0/13	192.168.100.1			pptp-out1	
AS	58.66.0.0/15	192.168.100.1			pptp-out1	
AS	58.82.0.0/15	192.168.100.1			pptp-out1	
AS	58.87.64.0/18	192.168.100.1			pptp-out1	
AS	58.100.0.0/15	192.168.100.1			pptp-out1	
AS	58.116.0.0/14	192.168.100.1			pptp-out1	

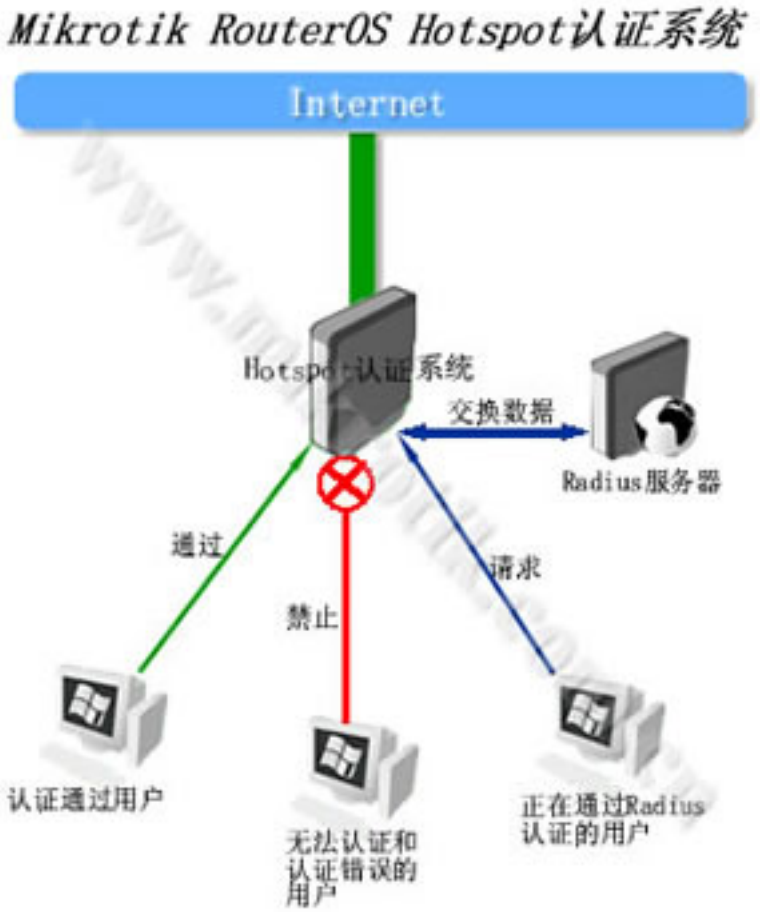
通过编辑电信的路由脚本，并导入路由表中，则实现了通过 PPTP 隧道使用 A 接入点的电信线路，完成了借线功能。

Mikrotik RouterOS HotSpot 介绍

Hotspot热点服务认证是一种友好的web方式的认证系统，在此种认证方式中，系统将自动要求未认证用户打开认证网页，验证通过后，便可连接到因特网，未认证用户无论输入任何一个网站地址，都会被强制到一个认证界面，要求用户进行认证。

基于web认证的接入网关要维护一个IP地址或MAC列表，依照这个表对所有收到的每个数据包进行检查，查看该数据包是否在允许通过之列，凡是开机后第一次进行www浏览的而没有通过认证的数据包，不被允许通过，接入网关会将一个web的认证界面推给用户，让用户进行认证，认证通过后，就把该用户的IP地址加入到IP地址列表中，如果不是有权用户HTTP包文就丢弃，如果收到的包允许通过，就进行地址转换或直接使用公网地址替换原地址，而后送出。基于web认证的接入网关也可以通过Radius进行认证，此时Hotspot认证账号管理作为Radius的client。将接收到的要求认证的数据转发给Radius服务器，Radius服务器会在自己的数据内查询用户资料，并判断是否能通过。

下面是一个HotSpot认证系统情况结构：



上面是用HotSpot做为认证网关，内网防火墙用户阻止用户的一些非法数据，保证认证网关的安全，过滤用户向外发出的相应病毒端口，控制用户对外的访问端口、数据、服务等。在内网设置防火墙是考虑到更多的病毒攻击和非法访问，以及过大的数据流量大多来至内网，当然你完全可以选择在认证网关前面增设一台对外的防火墙以保证网络更高的安全和稳定性。

与PPPoE认证对比

与PPPoE方式相比，基于Hotspot认证方式的好处是对网络设备无要求，它可以穿透三层设备，对本地三层交换和路由业务无影响，在使用家庭网关或机顶盒等有三层路由设备的网络使用Web方式比较适合。

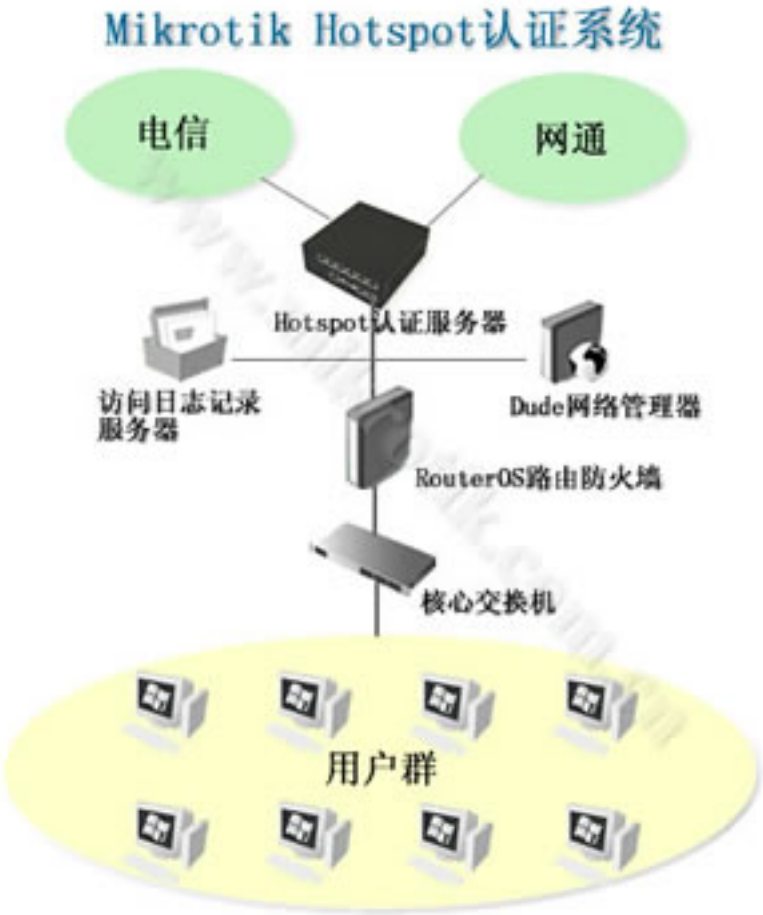
应用范围

Hotspot认证无需安装客户端软件，可以节省运维成本。而且支持Web页面的修改提供增值业务操作，为运营商提供更多的赢利模式。Hotspot认证在酒店、宾馆、ISP运营商、社区等接入场合有许多应用。热点认证服务系统应用形式主要是针对一些小型企业用户，如酒店、单位、家属区、中小规模的校园网，认证系统放在网络出口处，对用户上网进行控制，并对其流量进行控制。

多线路功能

同样在RouterOS启用Hotspot认证也能实现多线路的接入，如电信和网通的双线方案，这样也解决了网络访问的互联互通的问题。

如下面的示意图：



VRRP虚拟冗余路由协议

热点认证系统是提供一套完整的容错硬件和软件解决方案。为了保证网络永不中断，采用了以下双重方式：提供VRRP的虚拟冗余路由协议，保证一台认证系统出现故障，另外一台马上接替工作。

支持各种主流的网络接入

热点认证系统支持PSTN、ISDN、DDN、ADSL、CABLE。支持高达1000M的网络接口，而不会降低网络出口的速度。

支持多种账户管理

面向用户的网管计费系统，支持用户上网的唯一性控制， 无论用户通过哪一台计算机访问Internet，可以对他们进行管理和计费，支持动态地址分配、静态地址 、MAC地址与帐号绑定，MAC与IP绑定，帐号与IP绑定。

支持多种用户类型，并支持新用户类型的扩展

A类帐号用户：不能访问国外站点

B类帐号用户：能够访问国外站点

方便的使用

无论您是管理员还是用户，您都能感觉到热点认证系统给您带来的方便，对于管理员，我们在提供友好的管理客户端的同时，添加了WEB管理和查询。对于用户的使用，我们不强迫您安装客户端（您也可以安装客户端，方便您的上网），只要您有浏览器，简单的通过浏览器验证就可以使用网络，并且，用户可以通过浏览器查询使用的时间，流量，费用等相关情况。

支持用户分组管理

热点认证系统支持对不同的用户类别进行分组管理，管理更加方便。

提供了灵活的控制手段

热点认证系统提供了许多设置选项对上网用户进行管理。比如包月未缴费，上网时段，每日上网时间等选项，当有一个条件满足后，用户帐号就可以被自动查封。支持时间表，用户只可以在规定的时间内访问Internet。

高效的带宽管理

热点认证系统提供高效的带宽管理功能，您可以自由的划分带宽，为每个用户定义带宽属性，从1bps-100Mbps的带宽设置，为特定的的用户或帐号定义保留带宽或不进行登陆认证，并且可以根据用户的使用情况动态调整带宽。

浏览技术，无需安装客户端软件

上网用户使用浏览器(Browser)进行身份认证登陆等操作，不需要特殊的客户端软件，减少了网络管理员的维护工作量。支持Web登录方式 提供日志记录，跟踪操作人员的操作情况，杜绝操作人员违纪现象。

报表生成系统，提供用户的上网行为分析。

支持完整的用户上网记录日志，包括用户上网时间、流量、网站等

主要特征：

* 用户通过时间与流量认证计费
* Cookie (存储用户的账号和密码)
* 带宽控制功能
* 定额控制（连接超时时间, 下载/上传传输限制）
* 实时用户状态信息显示
* 自定义认证HTML页(可以由你自己设计认证页)
* DHCP服务器分配IP地址
* 简单的RAIUS客户端配置
* MikroTik RouterOS Hotspot能与PPTP隧道、IPsec以及其它的一些功能配合使用。
* 可以通过Access Point与以太网接入用户。
* 定时广播指定的URL链接

认证方法：

* 用户的账号与密码
* MAC和IP地址

认证过程：

热点认证网关工作原理是，一个客户端必须通过网络提供者的认证注册处理，即在试图打开一个网页时，弹出一个登陆web窗口，输入账号和密码（如下面用户的帐号和密码同为test）：

HotSpot会在用户数据库中查询用户信息，如果存在用户的相关信息，便会弹出一个认证通过的web窗口。

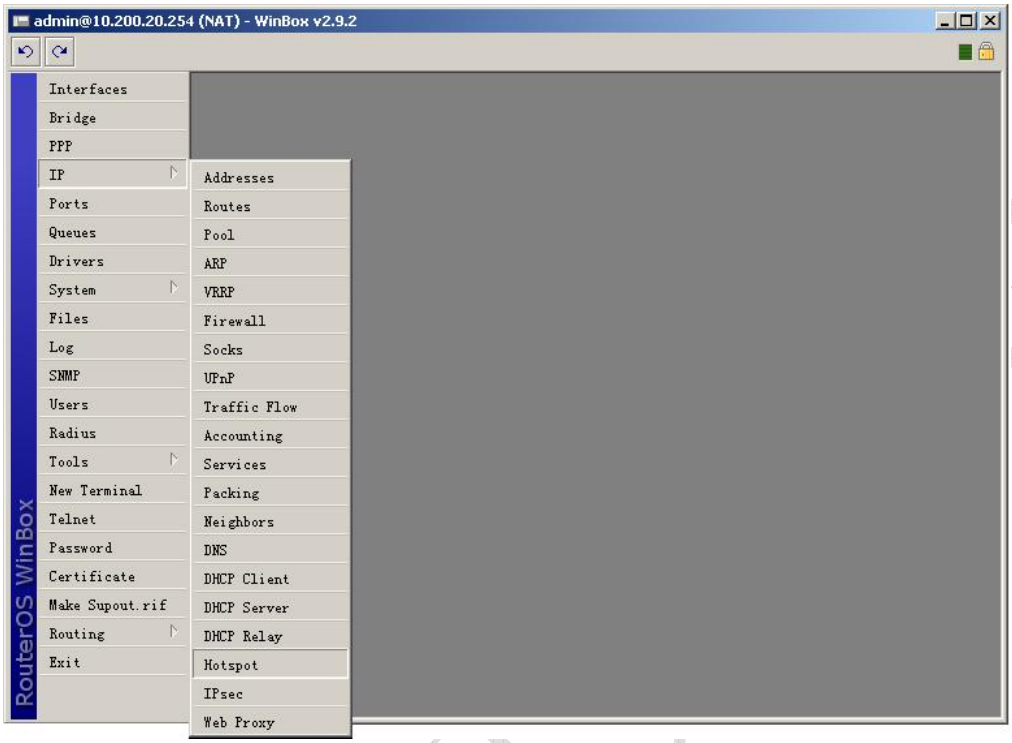
当用户离线是可以打开状态页点击log off（注销），退出认证。

以上是HotSpot认证上网的几个基本过程，对于用户来说是非常直观、简洁、方便。通过修改html格式的认证系统文件，可以提供设计你自己的认证页面。

Winbox管理

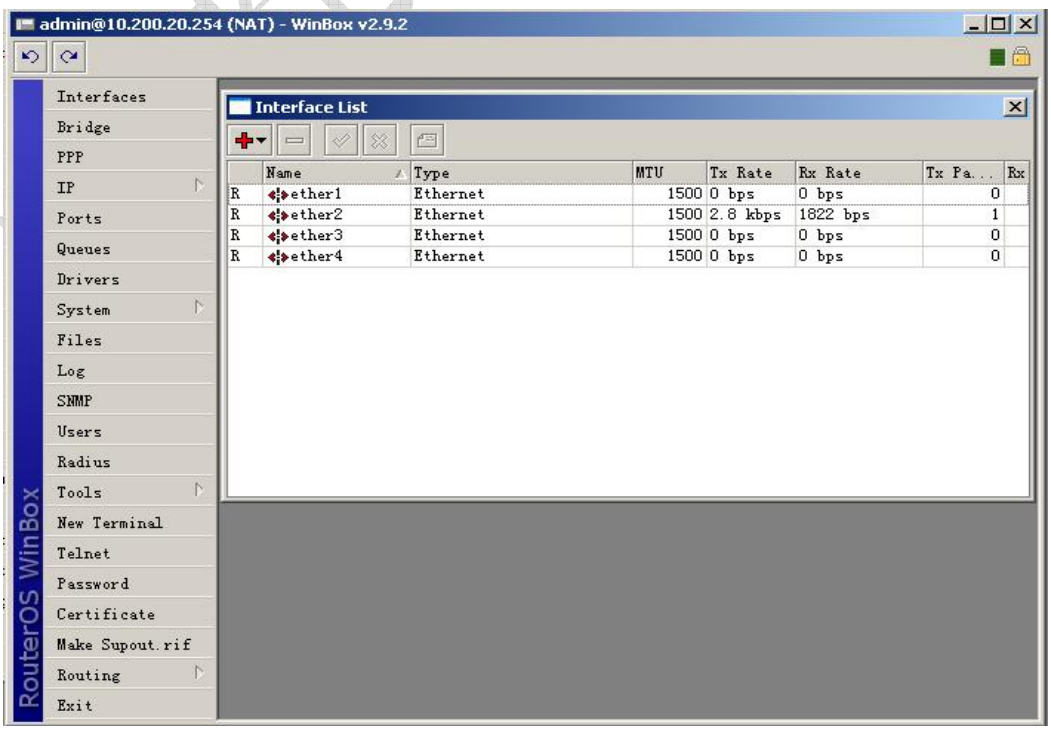
HotSpot网关认证系统提供了一个基于windows平台的图形化远程控制软件winbox，让用户能轻松管理这套认证系统。

下面介绍一下HotSpot在winbox中的基本操作流程以及相应的功能。
当在RouterOS本机通过命令操作设置完网卡和IP后，即可使用winbox与RouterOS连接了。这是一个winbox的操作图像界面：

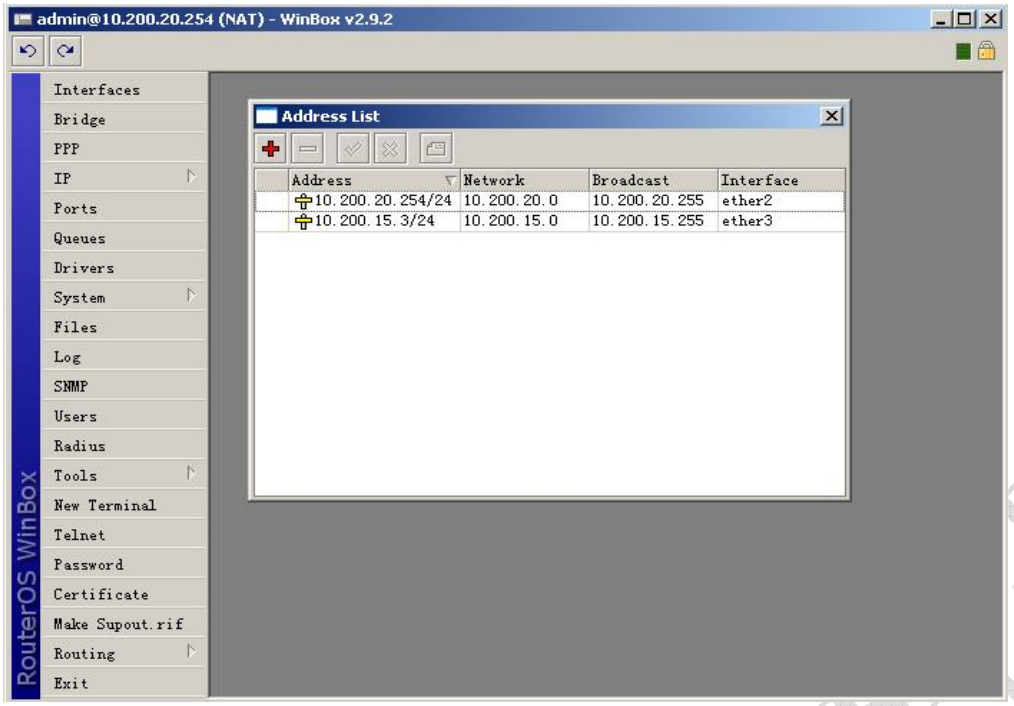


在图形界面的左边为根目录选项，在根目录中包括如：Interfaces、Bridge、IP、PPP 等一些基本的网络设置。如在IP目录中又分为了Address、Routers、ARP、Firewall、DNS、DHCP、Hotspot等功能设置选项。
在图形界面的左上角有两个方向相反的箭头，分别是撤销和恢复，用于当用户操作失误时反回之前的操作。
右上角有一个绿色的图标为CPU占用率。

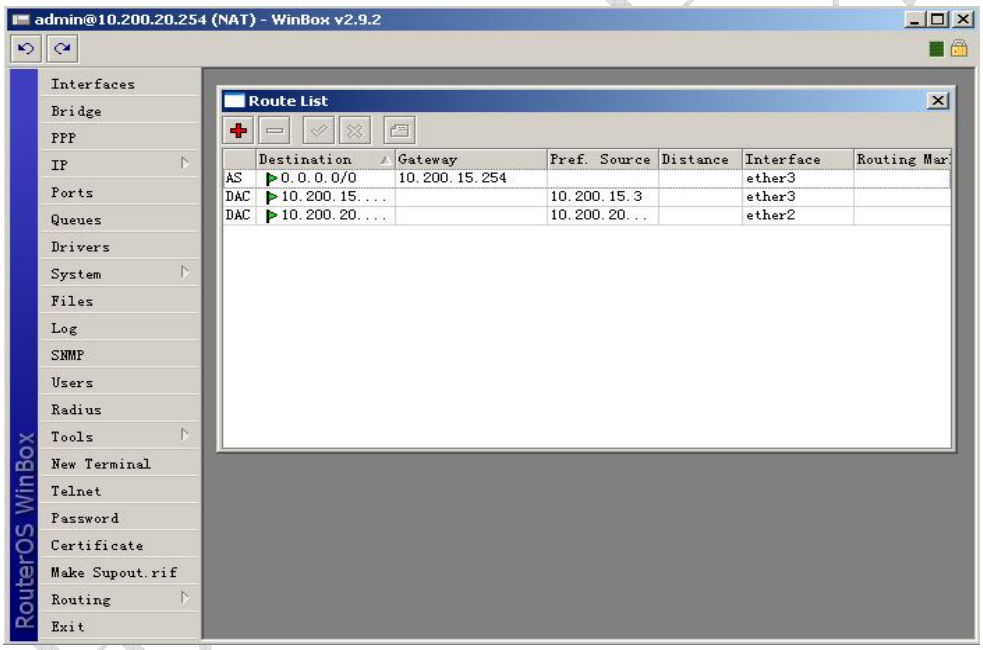
在winbox打开根目录的Interfaces选项，观测网卡的流量情况：



如果需要添加IP或查看IP地址设置，点开winbox中的IP中Addresses



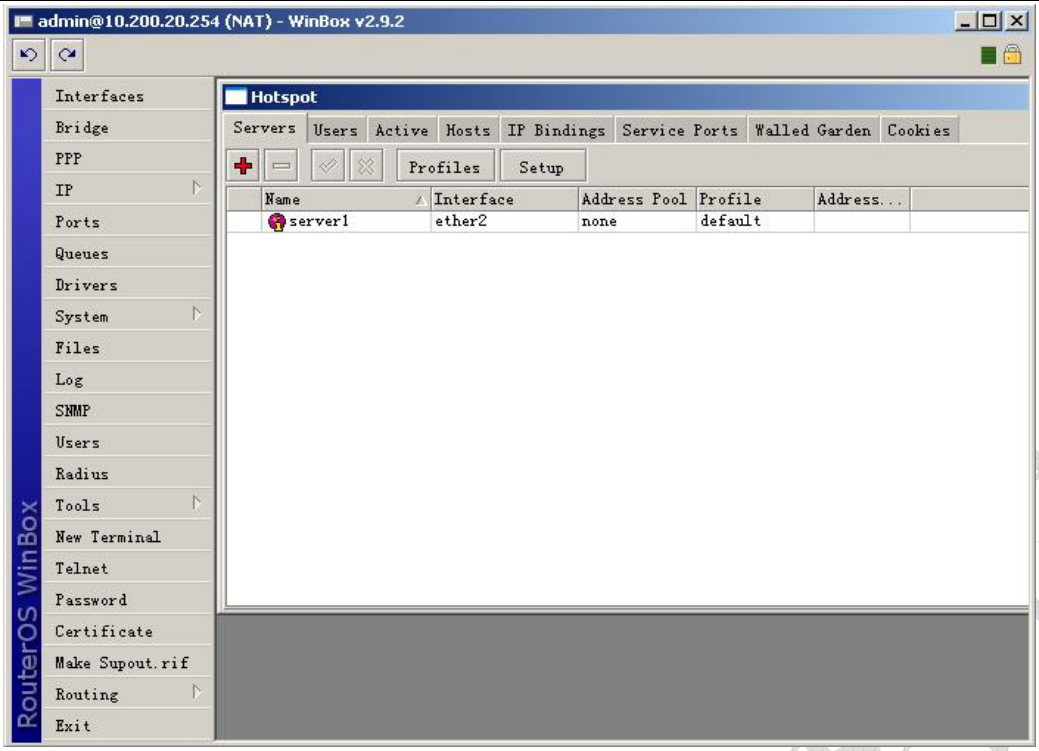
查看和添加路由在IP中的Routers：



以上的IP和Router选项都最基本的网络连接参数，在这些基本参数设置完成后，设置好NAT即可以正常连接网络。

HotSpot的功能介绍

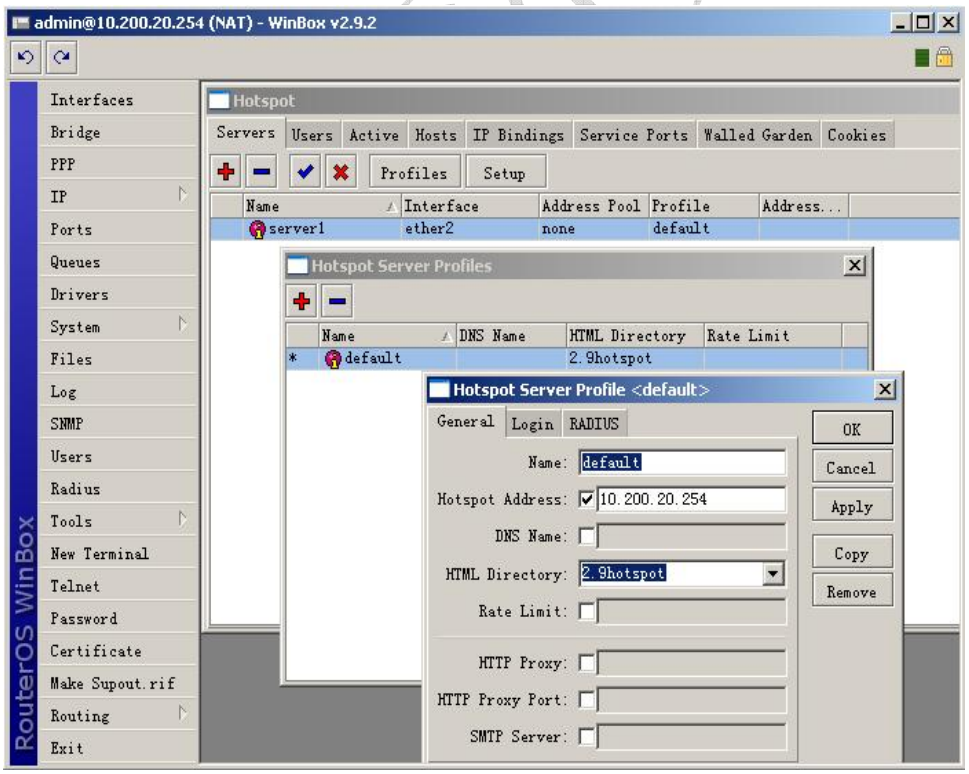
HotSpot的管理和设置在IP中的Hotspot里，下面是HotSpot选项控制界面：



打开HotSpot后里面可以看到上面的标签栏分别有Servers、Users、Active、Hosts、IP-bindings、Service-ports、Walled-garden、Cookies。

在上面的解图中我们可以看到现在HotSpot中选中的为Servers的标签，在里面可以看到增添了一个server1的服务，指向的是ether2的网卡，在address-pool中显示的为none，说明DHCP没有启用，这个服务使用的profile为默认的。RouterOS HotSpot允许添加多个认证服务接口，实现多接口认证。

在下面简单介绍以下几种主要功能：



在服务的profile中，有一条默认的策略（default），在这条策略中我们可以看到在general中设置内容包括

Name – profile的名称

Hotspot Address – 认证网关的地址

DNS Name – 认证网关的域名

HTML Directory – 指定认证页的路径

Rate Limit – 速率限制

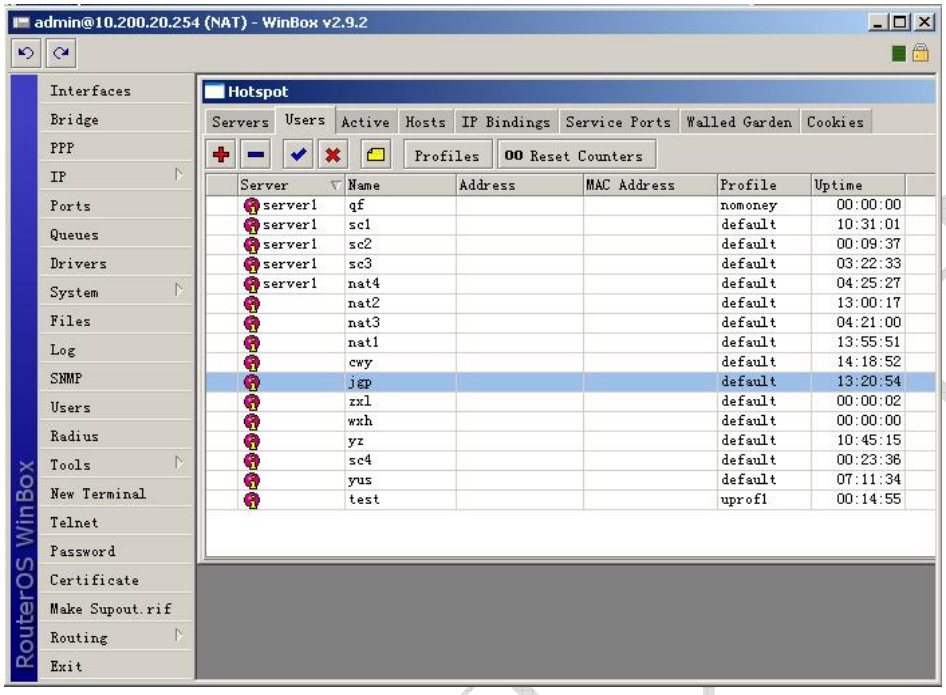
HTTP Proxy – HTTP代理

HTTP Proxy Port – HTTP 代理端口

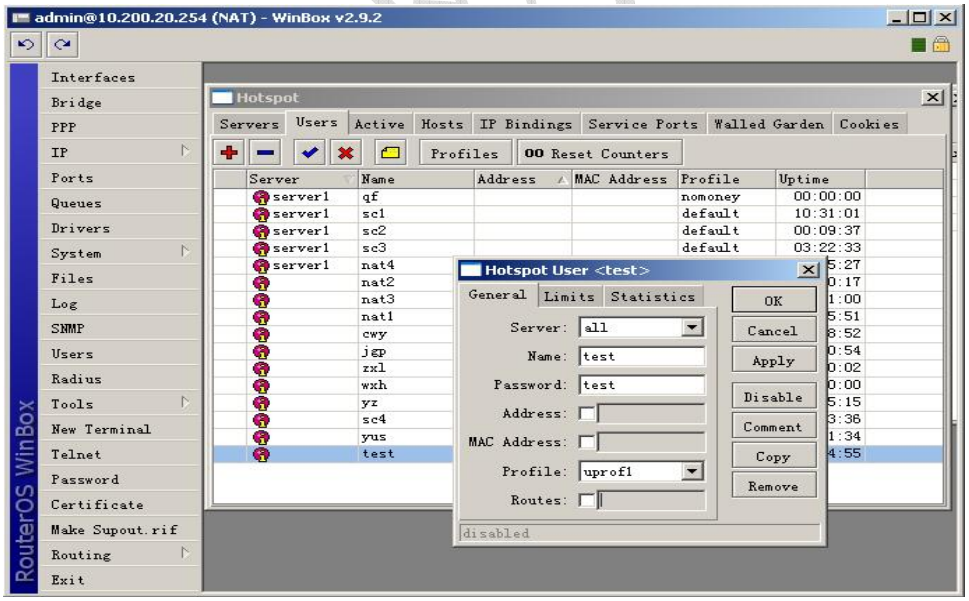
SMTP Server – 邮件服务器

在profile最后一个标签中的Radius，是用于与Radius计费服务器连接设置。

在Users选项中添加和删除掉用户帐号，并通过profiles设置用户类型。



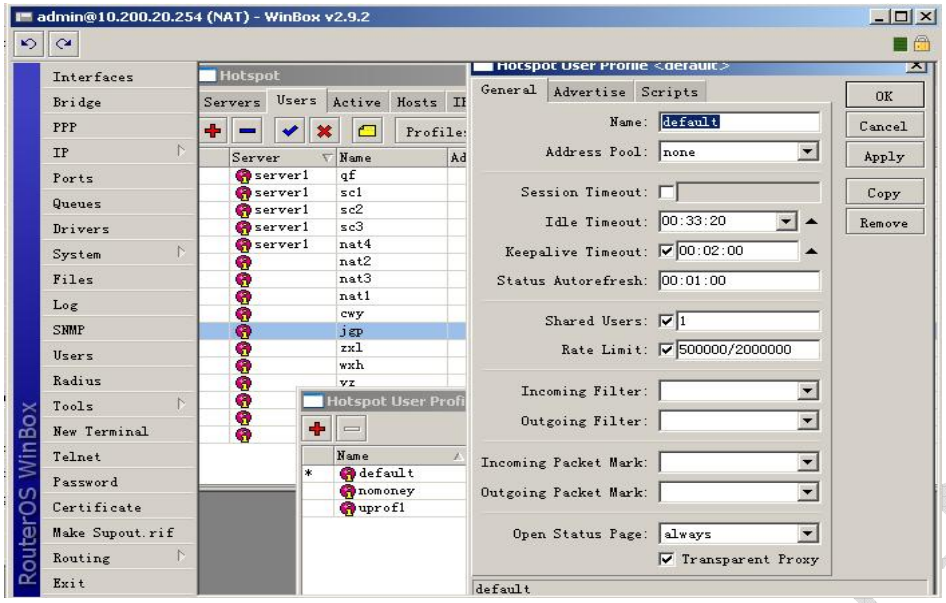
上面的截图可以看到用户帐号管理的设置界面，在这里用于管理用户的帐号和类型。



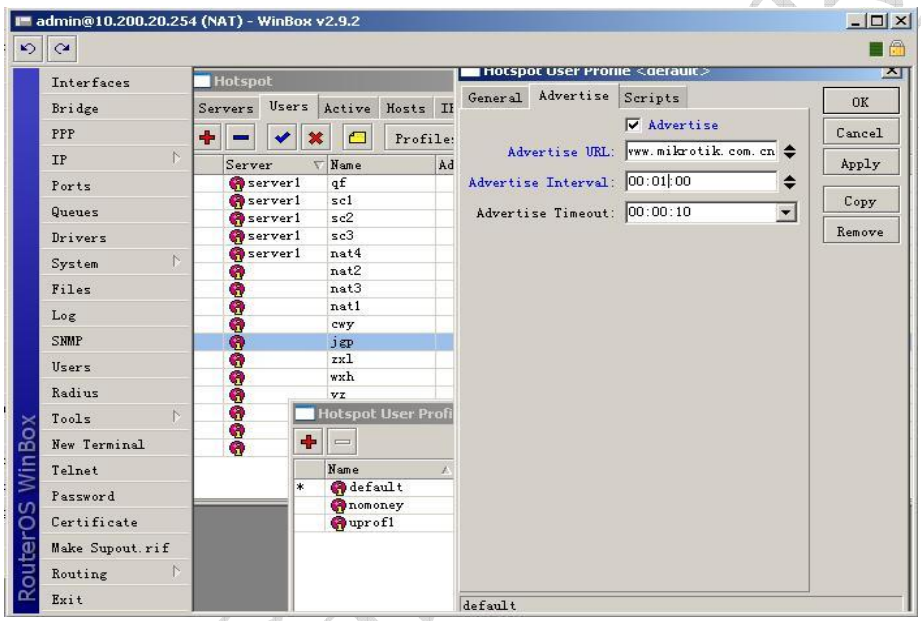
上面可以看到添加一个test帐号的设置，在添加帐号时可以看到在里面有Address和MAC Address，这两个设置是用于绑定用户的IP和MAC地址，下面的Routers是用于帐号的路由选择。Profile是用户帐号的类型，通过在profiles中定义参数。

在后面的Limits标签中是用于设置用户的计时和流量限制参数。

下面的截图是关于用户的profiles的定义：



在这里面可以看到对用户帐号类型的各种定义，包括IP地址池的分配，连接超时时间，帐号共享数，账号的带宽设置等等。



上面的截图是设置该类型的帐号的定时广播功能，即向该类型定时打开指定的URL连接。这样可以向用户提醒一些紧急通知、广告宣传和缴付通知等。

在标签Hosts用于查看到登陆用户的IP、MAC、连接状态、流量等信息，为管理者提供用户连接的实时状态情况，能更好的管理访问用户。

在HotSpot通过设置IP-Bindings可以设置特殊用户，这些特殊用户可以分类为阻止认证或是绕过认证等。

不仅在认证方面的各种功能，还提供了DNS缓存、Web缓存、负载均衡、策略路由等等各种网络功能。同样提供了完善的日志记录功能，并且能对CPU、内存、流量、硬盘等的天、周、月、年的记录，达到管理者能完全的了解认证系统的运转情况，通过对日志数据的分析，急时对网络或服务器做出修改和调整。

HotSpot在对用户认证和管理方面都非常的完善，并具备自己所有的网关特色，能对用户的上网线路进行优化，加快用户上网的访问速度。而其在价格上也是非常具有竞争力的一款路由认证软件，有极高的性价比。

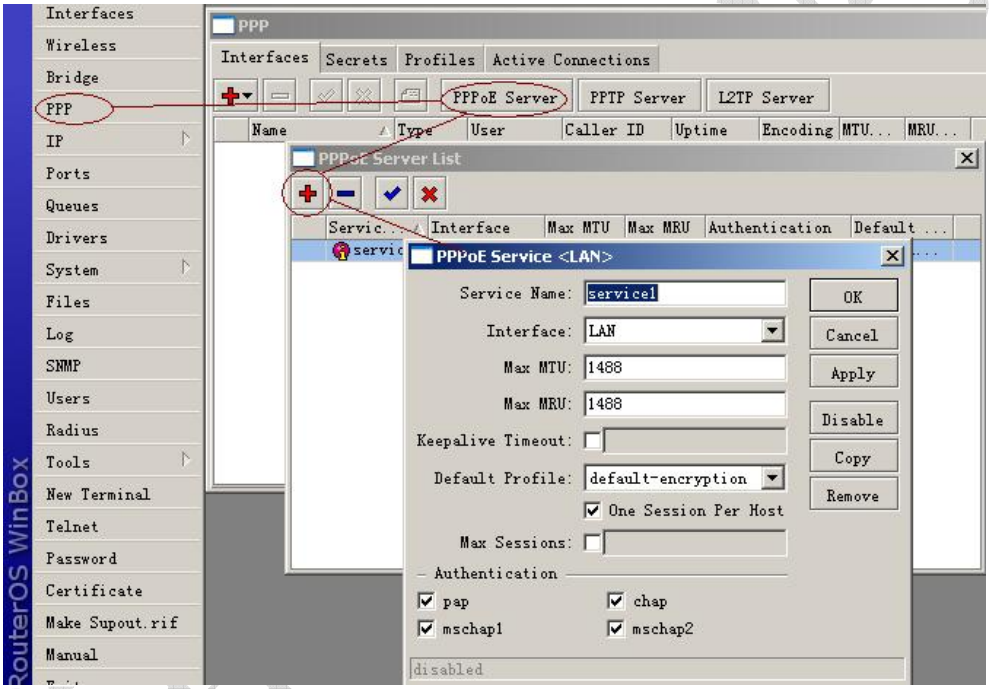
PPPoE Server 配置

PPPoE 基于以太网的点对点协议(Point to Point Protocol over Ethernet)当前的 PPPOE 主要被 ISP 商用于 xDSL 和 cable modems 与用户端的连接，他们几乎与以太网一样。 PPPoE 是一种标准的点对点协议(PPP) 他们之间只是传输上的差异：PPPoE 使用 modem 连接来代替普通的以太网。一般来说, PPPoE 是基于与用户认证和通过分发 IP 地址给客户端

一个 PPPoE 连接由客户端和一个访问集线服务器组成，客户端可以是一个安装了 PPPoE 协议的 windows 电脑。 PPPoE 客户端和服务器能工作在任何以太网等级的路由器接口（interface） - wireless 802.11 (Aironet, Cisco, WaveLan, Prism, Atheros), 10/100/1000 Mbit/s Ethernet, RadioLan 和 EoIP (Ethernet over IP tunnel)都支持。

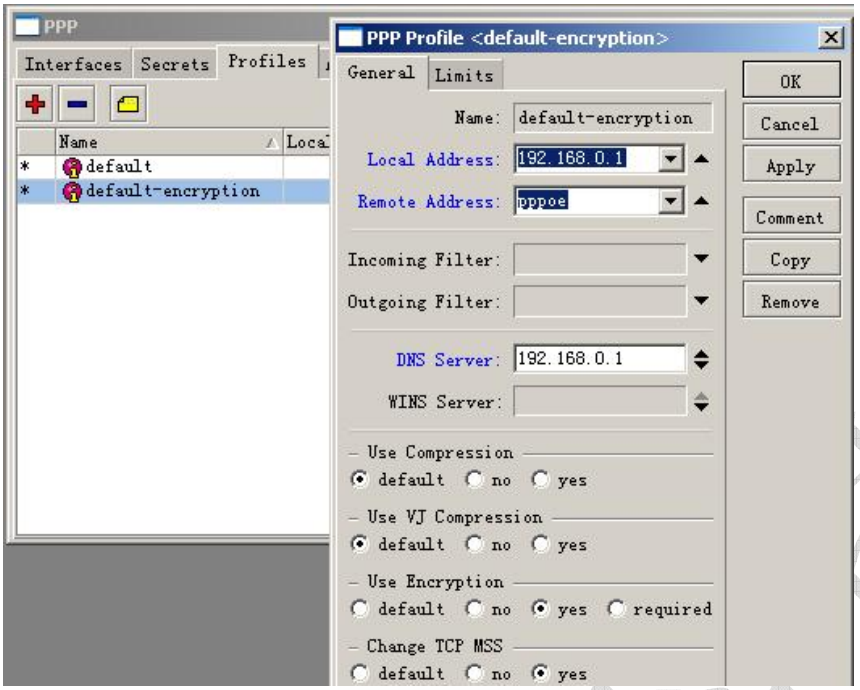
需要等级： *Level1 (限制 1 个连接)，Level3 (限制 200 个连接)，Level4 (限制 200 个连接)，Level5 (限制 500 个连接)，Level6 (无限制)*

现在我们建立一个 PPPoE-Server，首先我们进入 winbox 的 ppp 目录：

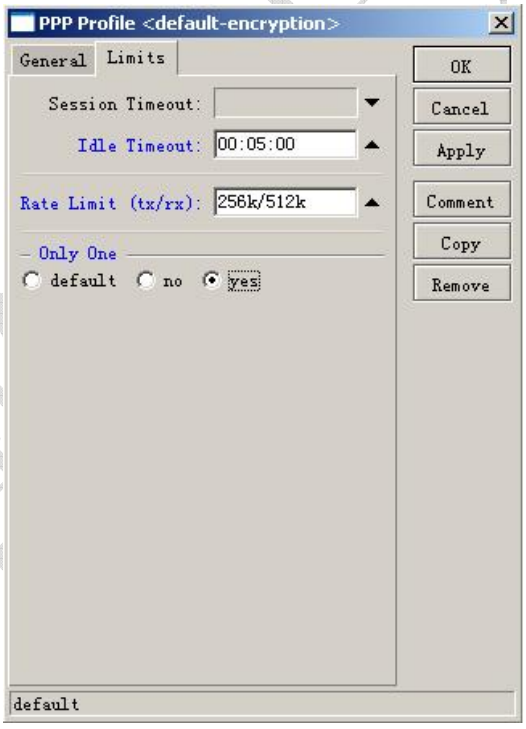


注： **keepalive-timeout** 值通常情况下设置为 **10**。如果你设置为 **0**，路由器将不会断开客户端，直到他们自己注销或是路由器重启该用户帐号才会断开。解决这个问题，**one-session-per-host** 属性需启用

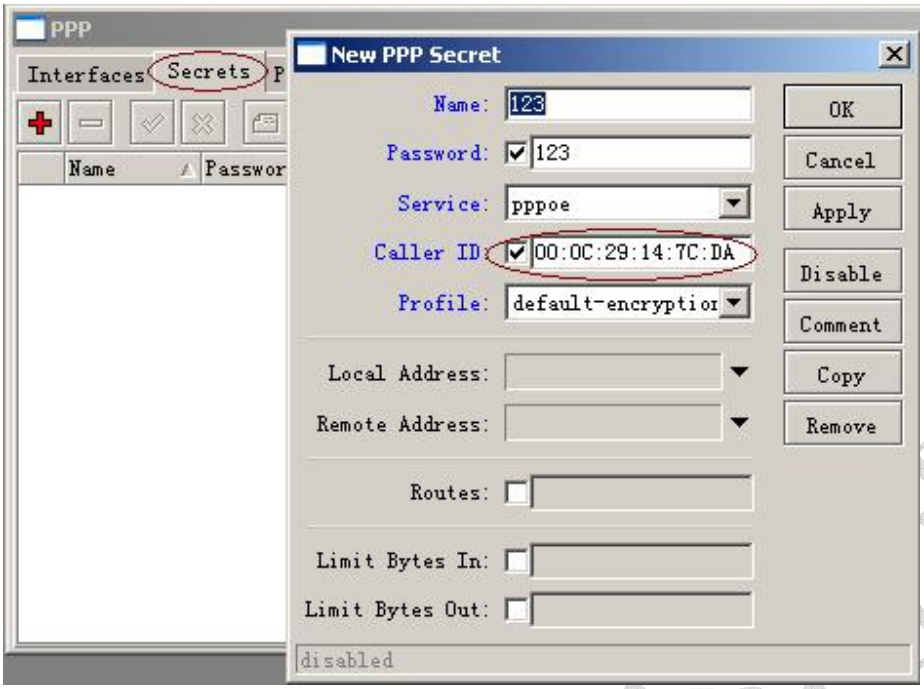
接下来我们建立 PPPoE Server 的 profile，定义客户的类型我们选用 default-encryption（数据加密方式传输）：



在 Limits 中是配置账户的相关限制参数：



配置客户账号和相关信息：



这样 PPPoE Server 基本配置完成

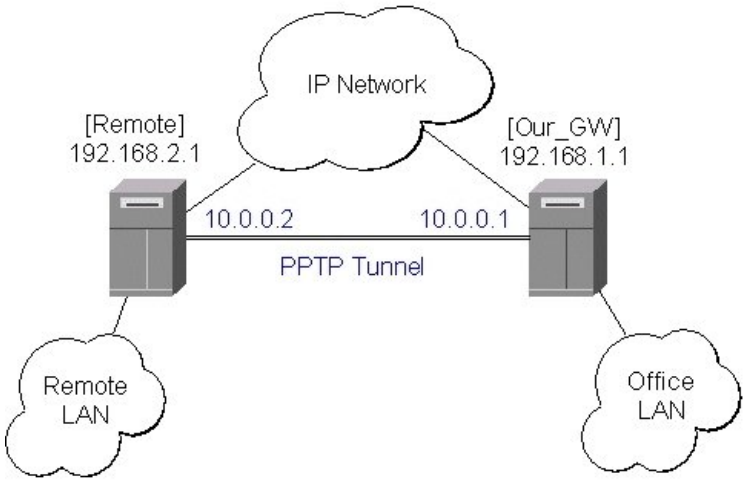
EoIP 应用实例

描述

我们假设要桥接两个网络：'Office LAN'和'Remote LAN'。网络通过路由器[Our_GW]以及[Remote]连接到一个 IP 网络。IP 网络可以是私有企业网或者因特网。这两个路由器通过这个 IP 网络通信。

实例

我们的目标是创建在路由器和桥之间且两个网络都通过它的一个安全频道。



为了在两个路由器之间创建一个安全的以太网桥，你应该

1. 在他们之间创建一个 PPTP 隧道。Our_GW 将成为 PPTP 服务器：

```
[admin@Our_GW] interface pptp-server> /ppp secret add name=joe service=pptp \
\... password=top_s3 local-address=10.0.0.1 remote-address=10.0.0.2
[admin@Our_GW] interface pptp-server> add name=from_remote user=joe
[admin@Our_GW] interface pptp-server> server set enable=yes
[admin@Our_GW] interface pptp-server> print
Flags: X - disabled, D - dynamic, R - running
  #    NAME              USER       MTU   CLIENT-ADDRESS  UPTIME   ENC...
  0    from_remote        joe
[admin@Our_GW] interface pptp-server>

The Remote router will be the pptp client:

[admin@Remote] interface pptp-client> add name=pptp user=joe \
\... connect-to=192.168.1.1 password=top_s3 mtu=1500 mru=1500
[admin@Remote] interface pptp-client> enable pptp
[admin@Remote] interface pptp-client> print
Flags: X - disabled, R - running
  0  R name="pptp" mtu=1500 mru=1500 connect-to=192.168.1.1 user="joe"
       password="top_s2" profile=default add-default-route=no

[admin@Remote] interface pptp-client> monitor pptp
     status: "connected"
     uptime: 39m46s
     encoding: "none"

[admin@Remote] interface pptp-client>
```

查阅 PPTP 接口手册获得更多关于设置加密频道的细节。

2. 通过在两个路由器添加 EoIP 隧道接口配置 EoIP 隧道。当对 EoIP 隧道指定变量值时，使用 PPTP 隧道接口的 IP 地址：

```
[admin@Our_GW] interface eoip> add name="eoip-remote" tunnel-id=0 \
\... remote-address=10.0.0.2
[admin@Our_GW] interface eoip> enable eoip-remote
[admin@Our_GW] interface eoip> print
Flags: X - disabled, R - running
 0   name=eoip-remote mtu=1500 arp=enabled remote-address=10.0.0.2 tunnel-id=0
[admin@Our_GW] interface eoip>

[admin@Remote] interface eoip> add name="eoip" tunnel-id=0 \
\... remote-address=10.0.0.1
[admin@Remote] interface eoip> enable eoip-main
[admin@Remote] interface eoip> print
Flags: X - disabled, R - running
name=eoip mtu=1500 arp=enabled remote-address=10.0.0.1 tunnel-id=0
[Remote] interface eoip>
```

3. 在两个路由器上的 EoIP 和以太网接口之间启用桥接：

在 Our_GW 上：

```
[admin@Our_GW] interface bridge> add
[admin@Our_GW] interface bridge> print
Flags: X - disabled, R - running
 0  R name="bridge1" mtu=1500 arp=enabled mac-address=00:00:00:00:00:00 stp=no
    priority=32768 ageing-time=5m forward-delay=15s
    garbage-collection-interval=4s hello-time=2s max-message-age=20s

[admin@Our_GW] interface bridge> add bridge=bridge1 interface=eoip-remote
[admin@Our_GW] interface bridge> add bridge=bridge1 interface=office-eth
[admin@Our_GW] interface bridge> port print
Flags: X - disabled, I - inactive, D - dynamic
 #   INTERFACE     BRIDGE  PRIORITY PATH-COST
 0   eoip-remote   bridge1 128      10
 1   office-eth    bridge1 128      10
[admin@Our_GW] interface bridge>
```

同理，对 Remote：

```
[admin@Remote] interface bridge> add
[admin@Remote] interface bridge> print
Flags: X - disabled, R - running
 0  R name="bridge1" mtu=1500 arp=enabled mac-address=00:00:00:00:00:00 stp=no
    priority=32768 ageing-time=5m forward-delay=15s
    garbage-collection-interval=4s hello-time=2s max-message-age=20s
```

```
[admin@Remote] interface bridge> add bridge=bridge1 interface=ether
[admin@Remote] interface bridge> add bridge=bridge1 interface=eoip-main
[admin@Remote] interface bridge> port print
Flags: X - disabled, I - inactive, D - dynamic
 #    INTERFACE     BRIDGE  PRIORITY PATH-COST
 0    ether         bridge1 128      10
 1    eoip-main     bridge1 128      10
[admin@Remote] interface bridge> port print
```

4. 来自同一网络的地址既可以用于 Office LAN 又可以用于 Remote LAN。

故障分析

- **路由器可以相互之间 ping 通但 EoIP 隧道依然不能正常工作！**

  检查 EoIP 接口的 MAC 地址——它们不应该一样！