RouterOS

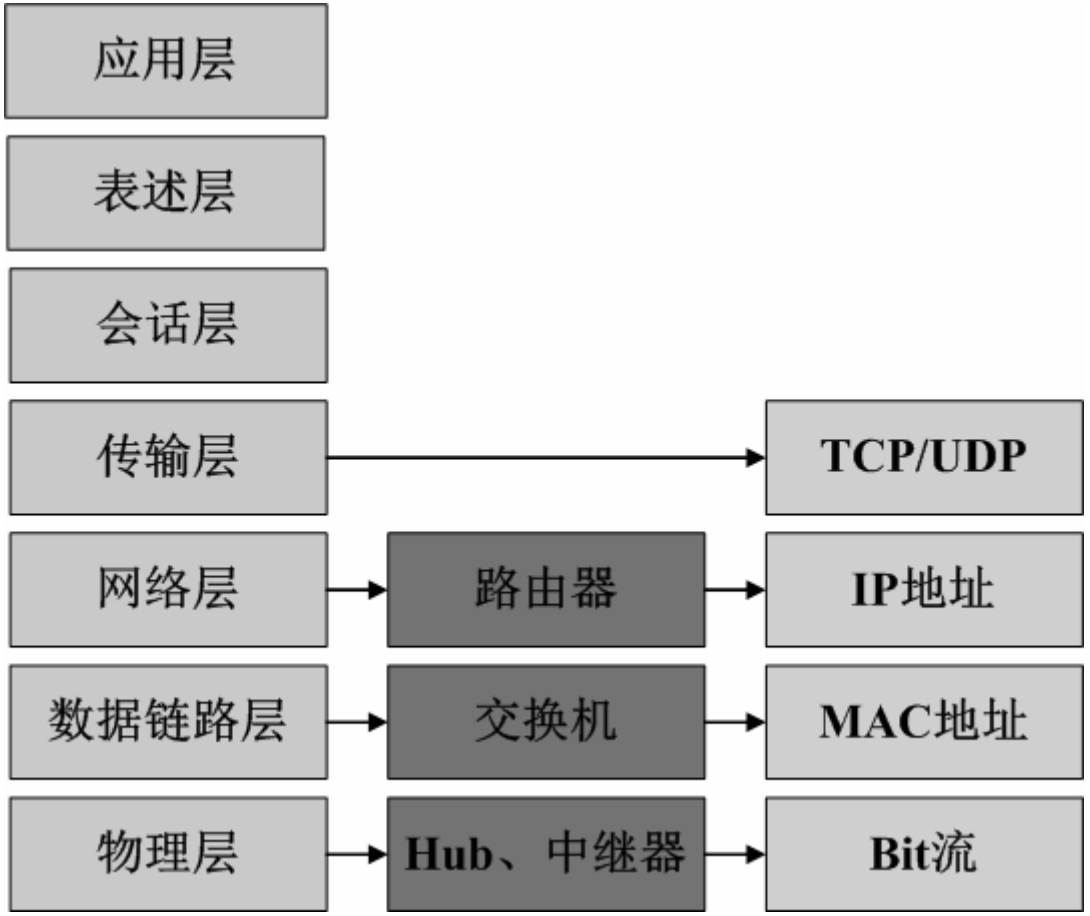
2010 培训

余松

培训要求

- 有一定的局域网实践经验
- 熟悉OSI 七层参考模型
- 需要了解基本的TCP/IP知识
- 了解路由器的基本工作原理

七层模型


应用层
表述层
会话层
传输层
TCP/UDP
网络层
路由器
IP地址
数据链路层
交换机
MAC地址
物理层
Hub、中继器
Bit流

交换机

- 二层交换机属数据链路层设备，可以识别数据包中的MAC地
址信息，根据MAC地址进行转发，并将这些MAC地址与对应的端口记录在自己内部的一个地址表中。

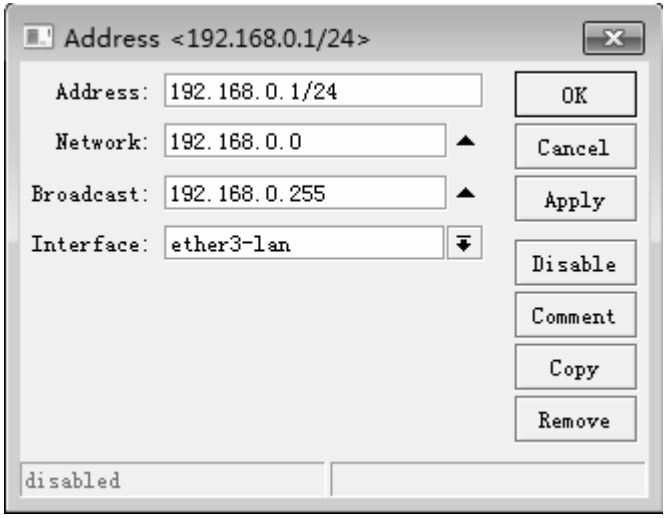
- 交换机的工作原理是学习、存储和转发，从各个网口上学习MAC地址，并存储到交换机的列表中，如果有数据发到交换机，则查找列表中的目标mac转发数据

IP地址

- IP地址组成是由主机地址和子网掩码组成
- 主机IP地址规定了主机在网络中的具体IP，即名称
- 子网掩码规则了这个IP段在网络中的范围
- IP地址是192.168.0.1，子网掩码255.255.255.0
- 即代表了IP地址是192.168.0.1，局域网中可以通信的范围是192.168.0.1-192.168.0.254

IP地址

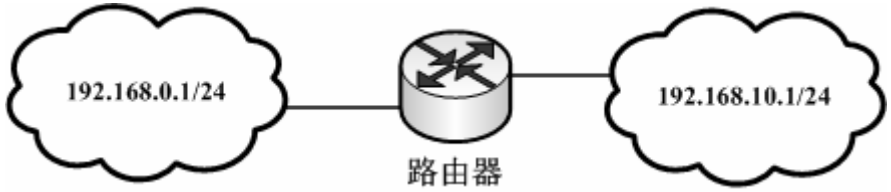
- 255.255.255.0如用二级制表示每个255都是2的8次方（0-255）256
- 我们可以用多少位代替十进制表示为24（3个255，3*8=24）如右图的winbox添加IP地址
- Address：IP地址/子网掩码
- Network：网络地址192.168.0.0
- Broadcast：广播地址192.168.0.255
- Interface：将这个IP设置到那一个网卡上
- 可以理解为Network是起始地址，Broadcast为结束地址，但这个两个地址被保留，只能是中间范围的IP地址可以被使用



路由Route

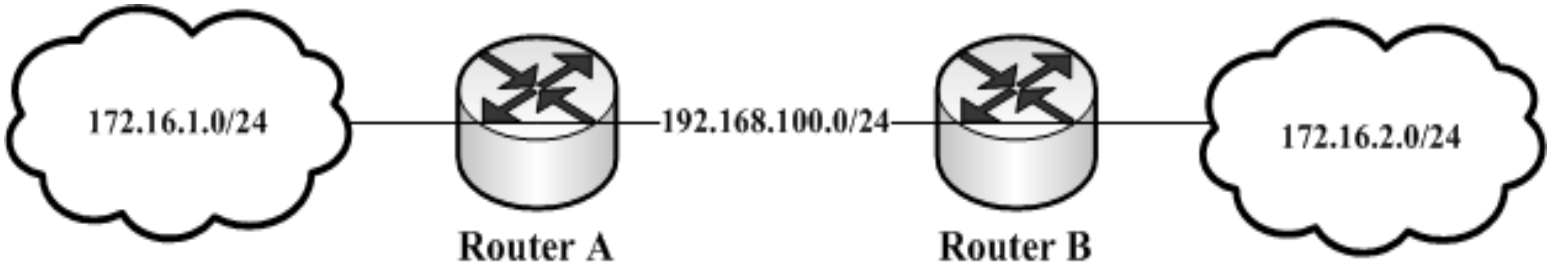
- 同一IP地址段我们连接可以使用交换机，但如果两个IP地址不同一个广播域内，如192.168.0.1/24和192.168.10.1/24那么他们之间是不能直接通过二层网络转发通信的

- 解决的方法是通过三层的路由器连接



- **路由器：**连接两个或两个以上不同IP网段的设备

路由 Route



- **路由表：**是路由器中路由表，是路由器根据IP数据包的源和目标地址进行路径选择的依据。
- IP数据的传输类似于接力，每个路由器都是一个接点，根据IP数据包的目的地将数据通过一个接一个的路由器放送到指定的目的地。

传输协议

- IP数据包需要在网络中传输，我们需要通过一个运载工具，这个就是TCP协议，包含TCP和UDP两种，通过TCP和UDP将IP地址传递到我们指定的目的地

- TCP是有链接的传输，可靠性高，要求的资源要高，大部分都采用TCP协议，如IE、FTP、Telnet

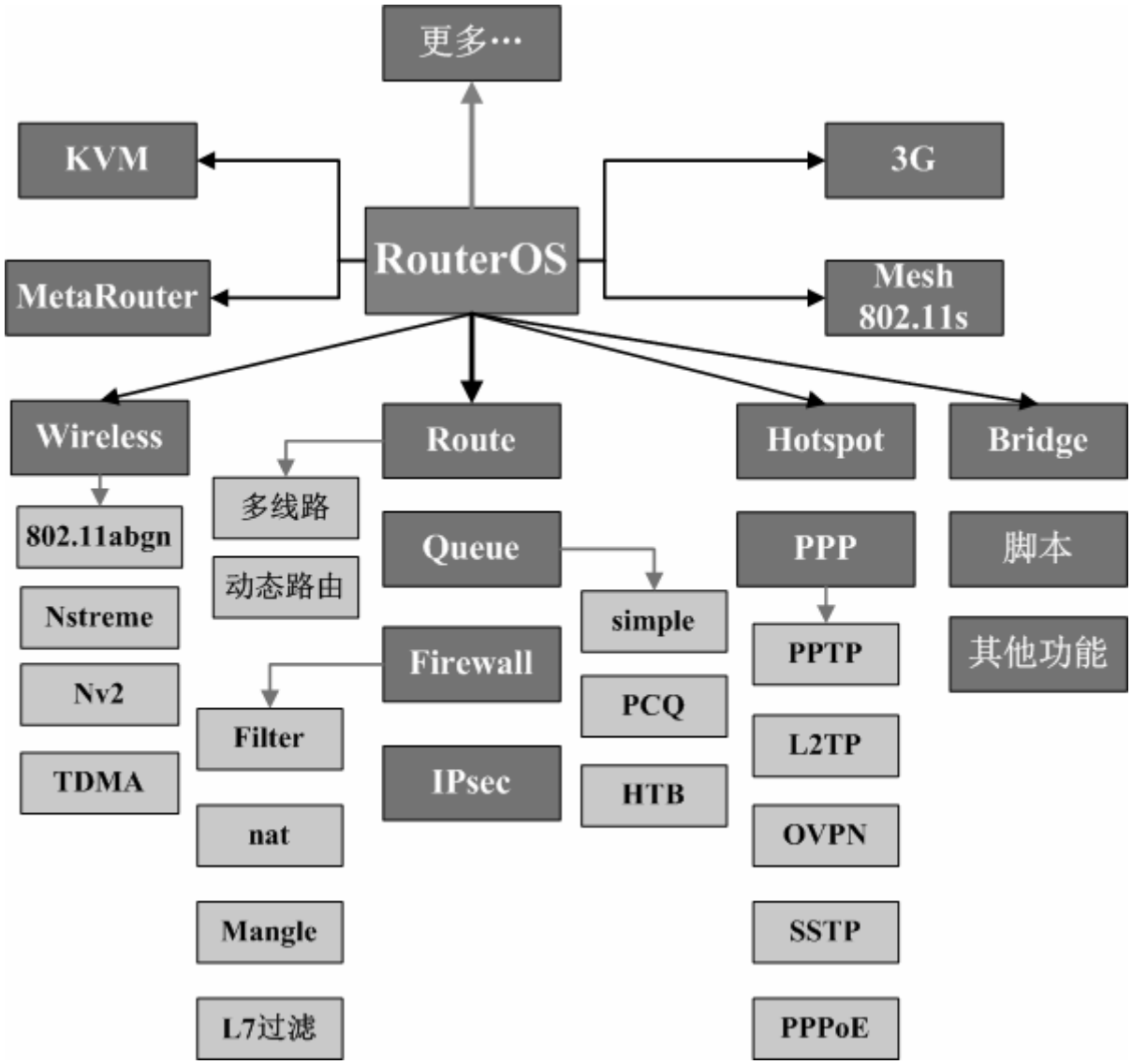
- UDP是无连接传输，不要求对方建立连接，资源开销低，如果DNS和QQ

什么是RouterOS

- 一种基于Linux核心的独立路由操作系统，不需要依附在其他的操作系统安装
- 支持大部分主流的网络协议和功能，支持操作系统的特性
- 具备脚本的编辑功能，实现系统智能化运行
- 兼容当前的所有x86平台的硬件，如Intel和AMD等，支持嵌入的平台RouterBOARD系列产品

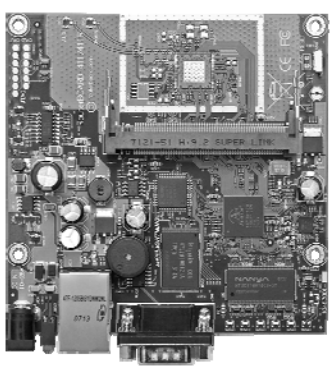
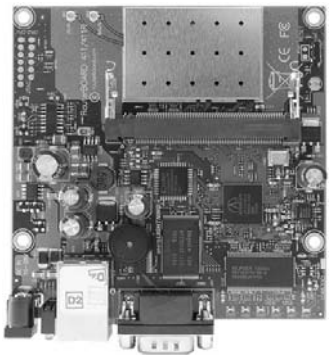
RouterOS应用的范围

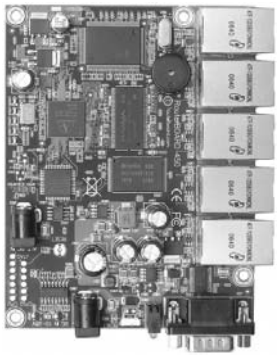
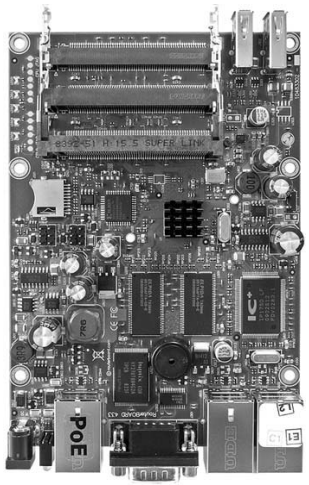
- 企业网络办公与网络安全管理
- 网吧多线接入与流量控制
- ISP认证与运营
- VPN互联与加密通信网络
- Wlan无线网络传输与覆盖

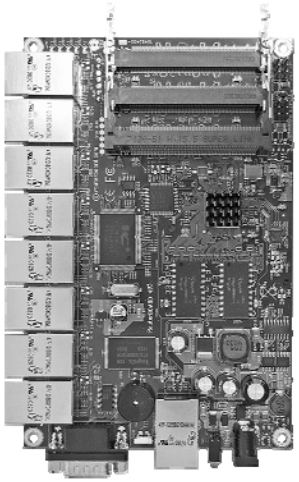
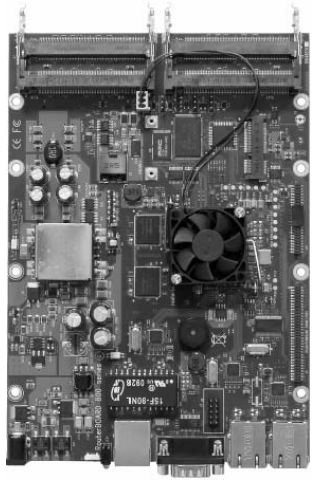

更多…
KVM
3G
RouterOS
MetaRouter
Mesh
802.11s
Wireless
Route
Hotspot
Bridge
802.11abgn
多线路
Queue
PPP
脚本
Nstreme
动态路由
simple
PPTP
其他功能
Nv2
Firewall
PCQ
L2TP
TDMA
Filter
IPsec
HTB
OVPN
nat
Mangle
SSTP
L7过滤
PPPoE

RouterOS平台

RouterBOARD平台

配套工具

- Winbox – 图像操作管理工具
- The Dude – 网络监控软件，并能生成拓扑结构
- NATBox – 中文管理界面，简化RouterOS操作
- NAT Radius - 基于中文的Radius计费软件，操作简单实用。

RouterOS等级区别

等级 / 功能	Level 0	Level3	Level 4	Level 5	Level 6
升级	24小时	4.x	4.x	5.x	6.x
无线AP	24小时	不支持	支持	支持	支持
无线桥接和客户端	24小时	支持	支持	支持	支持
RIP，OSPF，BGP协议	24小时	支持	支持	支持	支持
EoIP隧道在线用户	24小时	1条	无限制	无限制	无限制
PPTP隧道在线用户	24小时	1条	200	无限制	无限制
SSTP隧道在线用户	24小时	1条	200	无限制	无限制
PPPoE隧道在线用户	24小时	1条	200	500	无限制
L2TP隧道在线用户	24小时	1条	200	无限制	无限制
Hotspot认证在线用户	24小时	1条	200	500	无限制
VLAN	24小时	1条	无限制	无限制	无限制
P2P防火墙规则	24小时	1条	无限制	无限制	无限制
NAT规则	24小时	无限制	无限制	无限制	无限制
Radius客户端	24小时	支持	支持	支持	支持
Queue流量控制规则	24小时	无限制	无限制	无限制	无限制
Web代理	24小时	支持	支持	支持	支持
User Manager在线用户	24小时	10	20	50	无限制

RouterOS功能

- **RouterOS**现在所具备的功能,已经远远超出我们对一般路由器的理解。

- **RouterOS**是基于**Linux2.6**内核开发，通过编译后，实现各种功能的快速安装和操作

- 可以看成是一个路由化的操作系统，他与普通路由器的区别：

  ▲ 多功能平台 – 基本的路由功能外，还包含防火墙、 QoS流控、认证系统、web代理、VPN、WLAN、3G等等；

  ▲ 脚本编辑 – 使路由器应用更加方便和智能；

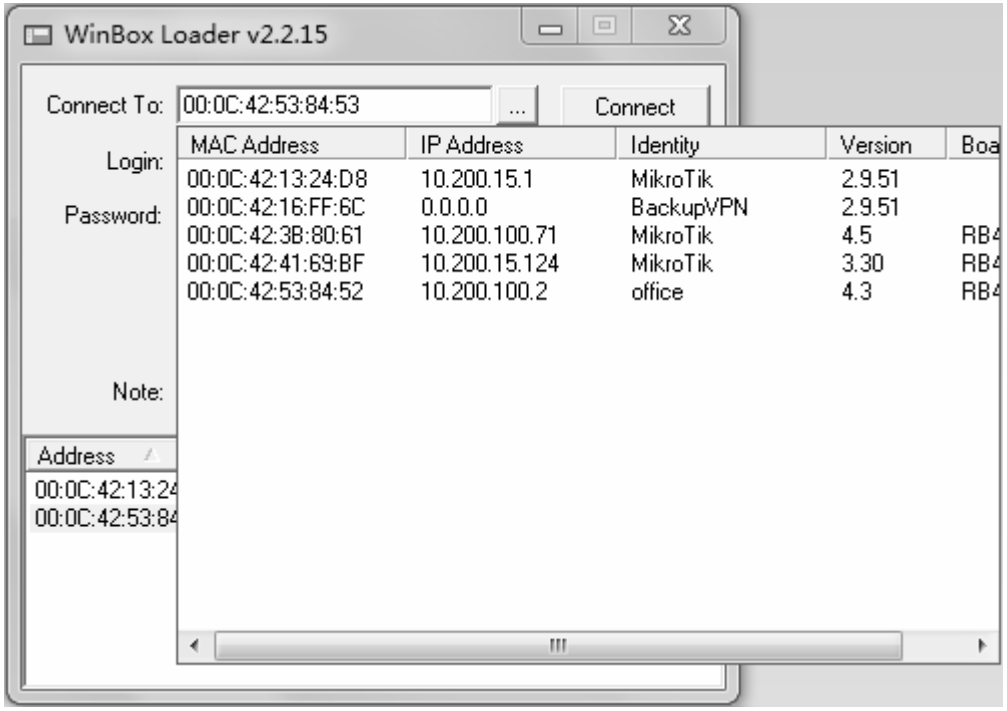
  ▲ 虚拟化技术 – 多操作系统的兼容，达到一机多用的功能。

RouterOS的配置工具

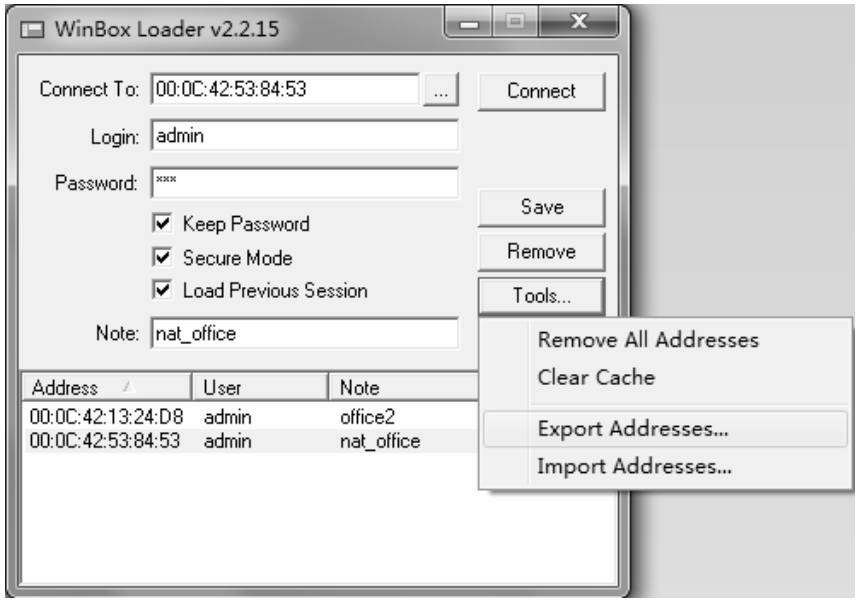
- RouterOS的配置工具主要是winbox软件，同样具备CLI命令行、Web配置接口
- 登陆RouterOS可以通过IP地址和MAC地址
- 在同一局域网内可以通过MAC地址登陆，不要修改电脑的IP和路由器同一IP段，只需要使用winbox软件。
- 远程连接，通输入对方的公网IP地址登陆

Winbox操作1

- 搜索和显示MNDP (MikroTik Neighbor Discovery Protocol) MikroTik邻居探测协议
- 可以通过该功能键搜索同一子网内MikroTik。并能通过MAC地址登陆到MikroTik RouterOS进行操作。
- 在winbox2.2.12后增加了可选择的MAC登陆或者IP登陆的功能

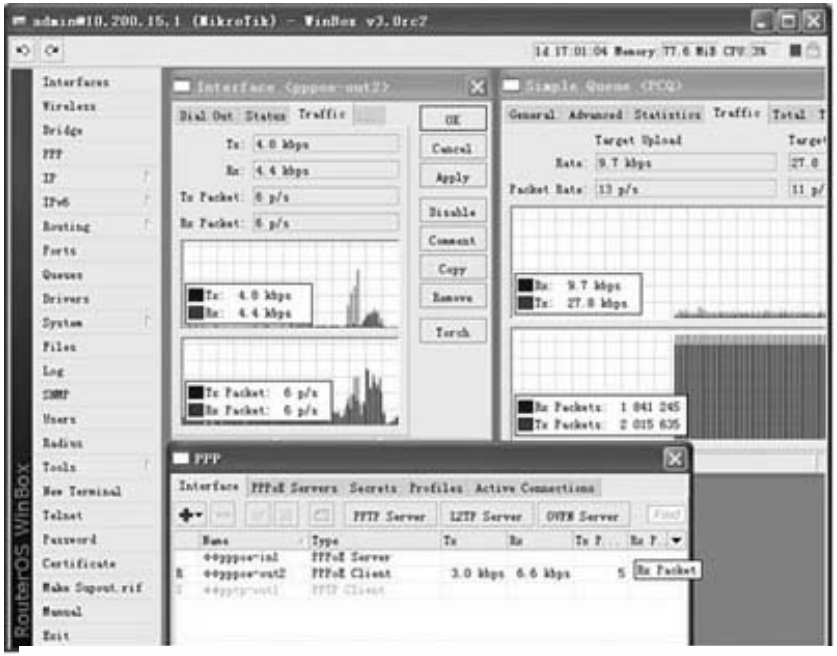


Winbox操作2



- Tools按钮：能删除所有列表中的项目，清除在本地的缓存。
- 能从wbx文件导入地址或导出为wbx文件
- Secure Mode（安全模式）：提供保密并在winbox和RouterOS之间使用TLS（Transport Layer Security）协议
- Keep Password（保存密码）：保存密码到本地磁盘的文本文件中

Winbox操作3

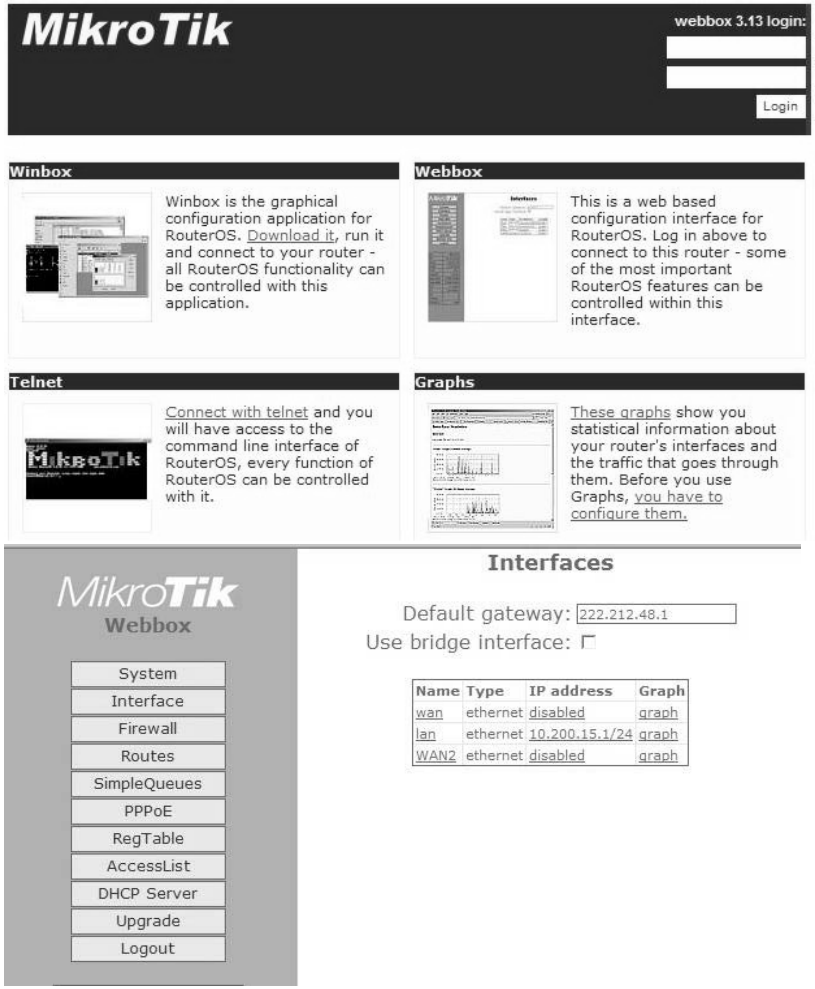


- Winbox连接使用TCP端口：8291
- RouterOS 默认登陆帐号为admin，密码为空

图标	功能		图标	功能
	添加一条项目			定义或编辑一个注释
	删除一条存在项目			查询关键字
	启用一个项目			撤销操作
	禁用一条项目			恢复操作

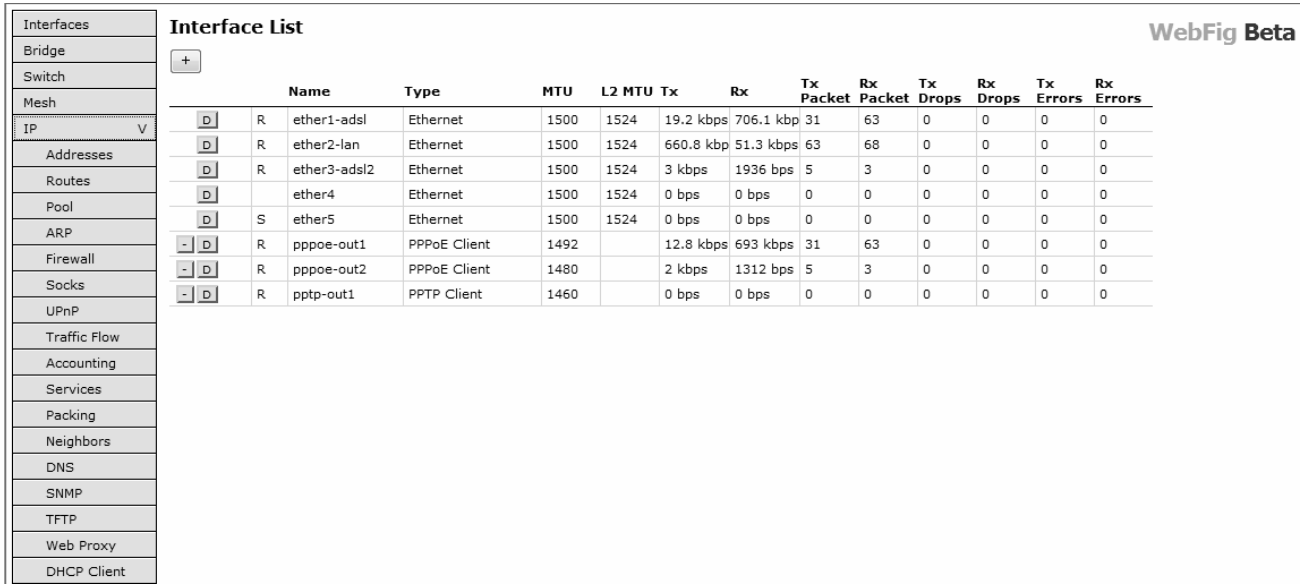
命令行与web接口

```
  MMM      MMM       KKK                          TTTTTTTTTTT      KKK
  MMMM    MMMM       KKK                          TTTTTTTTTTT      KKK
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     000000      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  000  000     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    000  000     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   000000      TTT     III  KKK  KKK

oct/10/2006 04:56:26 system,error,critical login failure for user root from 218
.1.65.233 via ssh
oct/10/2006 04:56:27 system,error,critical login failure for user test from 218
.1.65.233 via ssh
Terminal vt102 detected, using multiline input mode
[admin@MikroTik] > ip ad
[admin@MikroTik] ip address> prin
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         BROADCAST       INTERFACE
 0   10.200.15.1/24     10.200.15.0     10.200.15.255   lan
 1   10.200.16.1/24     10.200.16.0     10.200.16.255   lan
 2 D 222.212.51.143/32  222.212.48.1    0.0.0.0         pppoe-out1
[admin@MikroTik] ip address> /
[admin@MikroTik] > ip rout
[admin@MikroTik] ip route> prin
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf
 #     DST-ADDRESS        PREF-SRC        G GATEWAY         DIS INTERFACE
 0 ADC 10.200.15.0/24     10.200.15.1                         lan
 1 ADC 10.200.16.0/24     10.200.16.1                         lan
```



WebFig配置

- 最新的RouterOS v5.0版本支持网页的配置，90%是从winbox移植过来



RouterOS安装

- MikroTik RouterOS安装方式：

  * 使用带**ISO**的镜像文件通过光盘引导安装（用于**x86**系统）；
    支持AMD、Intel、VIA以及其他X86系统
    支持IDE、SATA硬盘接口

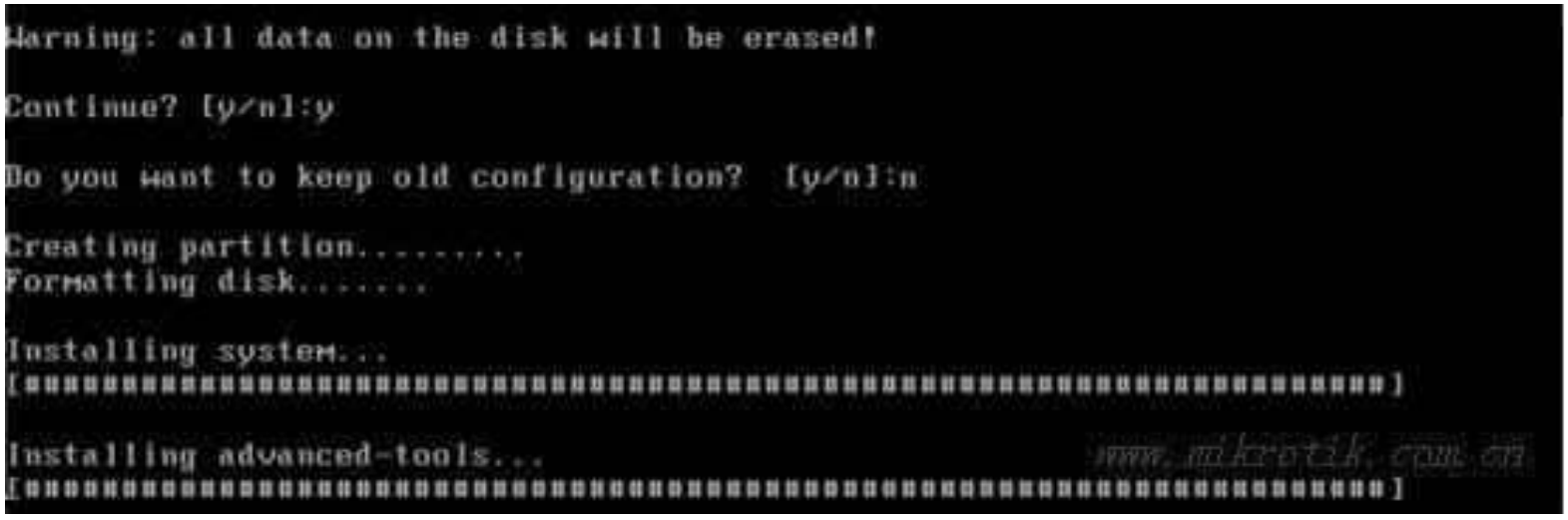
  * 使用**U**盘安装基于**X86**（限**3.0**版本）

  * 使用**netinstall**网络安装程序（主要用于**RouterBOARD**）；
    RB100、RB300、RB500、RB400、RB600、RB700、RB800、RB1000系列

RouterOS软件下载**http://www.mikrotik.com/download.html**

光盘安装

- BIOS设置光盘引导
- 自动进入安装页面，选择需要使用的功能包（system包是必须默认安



U盘安装 步骤

- U盘安装需要使用3.0以上版本netinstall软件
- 将U盘插入一台Windows电脑的USB接口
- 启动Netinstall软件，选择RouterOS-X86安装包
- 通过Netinstall安装RouterOS到U盘上
- 然后取出U盘，插入需要使用U盘的PC上，并设置PC通过USB引导启动

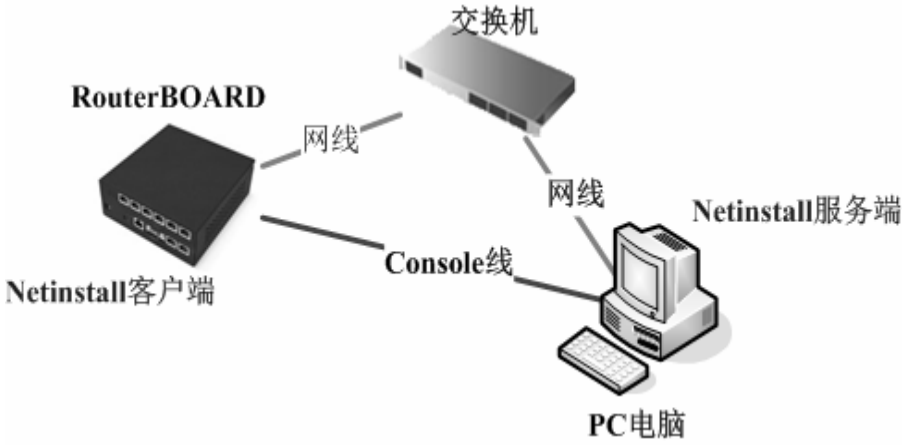
U盘安装

在**Netinstall**软件中可以找到**U**盘的盘符，并选择**x86**的安装包

安装时会格式化**U**盘，**RouterOS**会向里面写入引导程序

Netinstall网络安装

- Netintall 应用程序，通过连接RouterBOARD的串口和ether1的以太网口，安装RouterOS ；
- 在下面的情况下使用重新安装：
  * 忘记密码；
  * 文件损坏，重装RouterOS；
  * 导入配置脚本



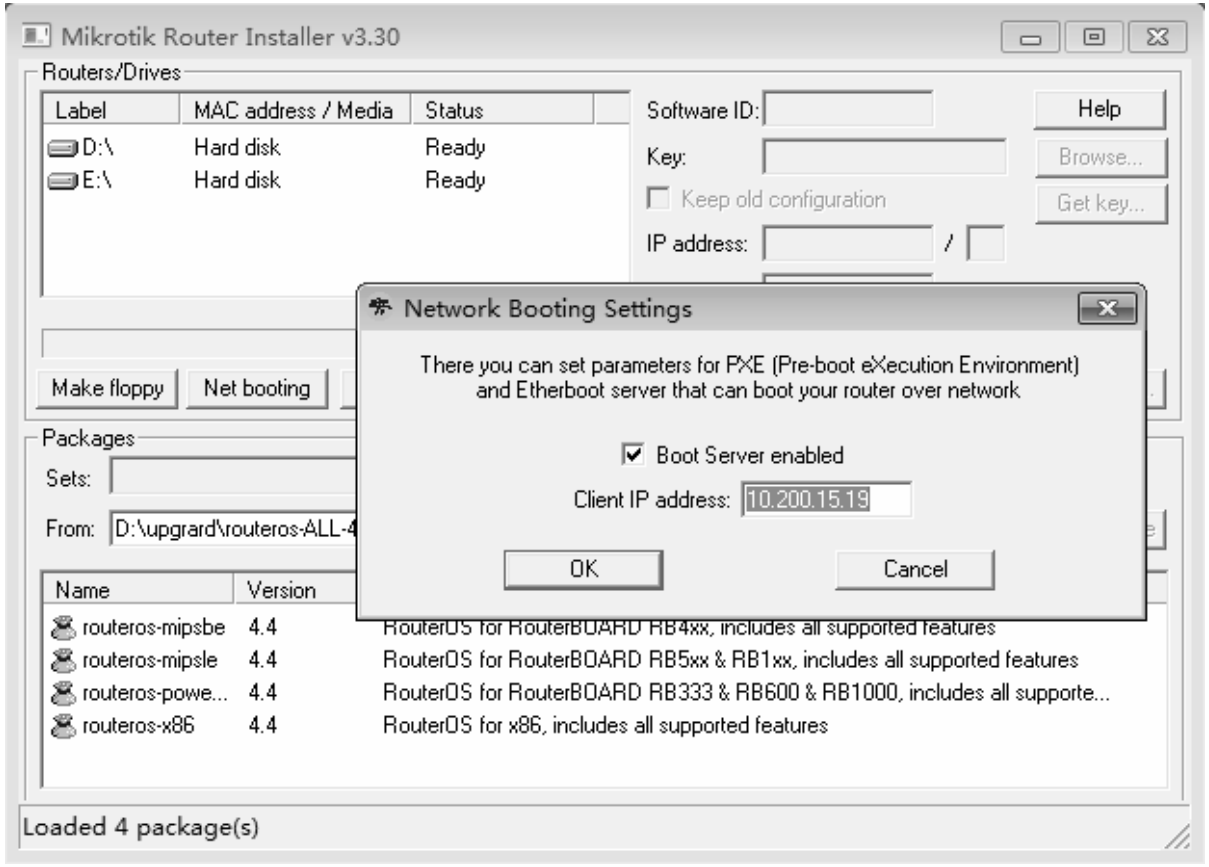
RouterOS串口线序

- RB系列和PC的串口线序如下：

DB9f	Function	DB9f	DB25f
1 + 4 + 6	CD + DTR + DSR	N/C	N/C
N/C	CD + DTR + DSR	1 + 4 + 6	6 + 8 + 20
2	RxD	3	2
3	TxD	2	3
5	GND	5	7
7 + 8	RTS + CTS	7 + 8	4 + 5

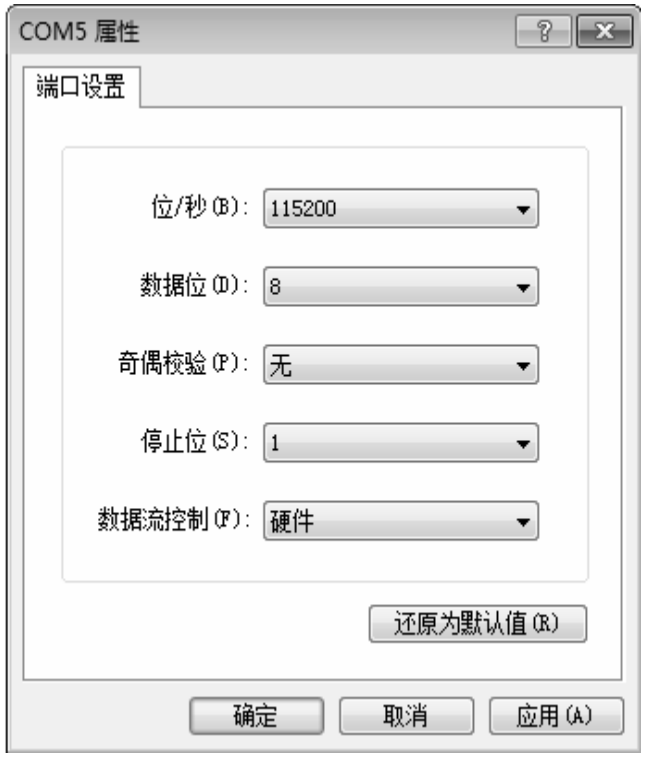
操作步骤1

- 在你的电脑上运行NetInstall 程序，确定软件包(*.npk文件)在你本地磁盘上。



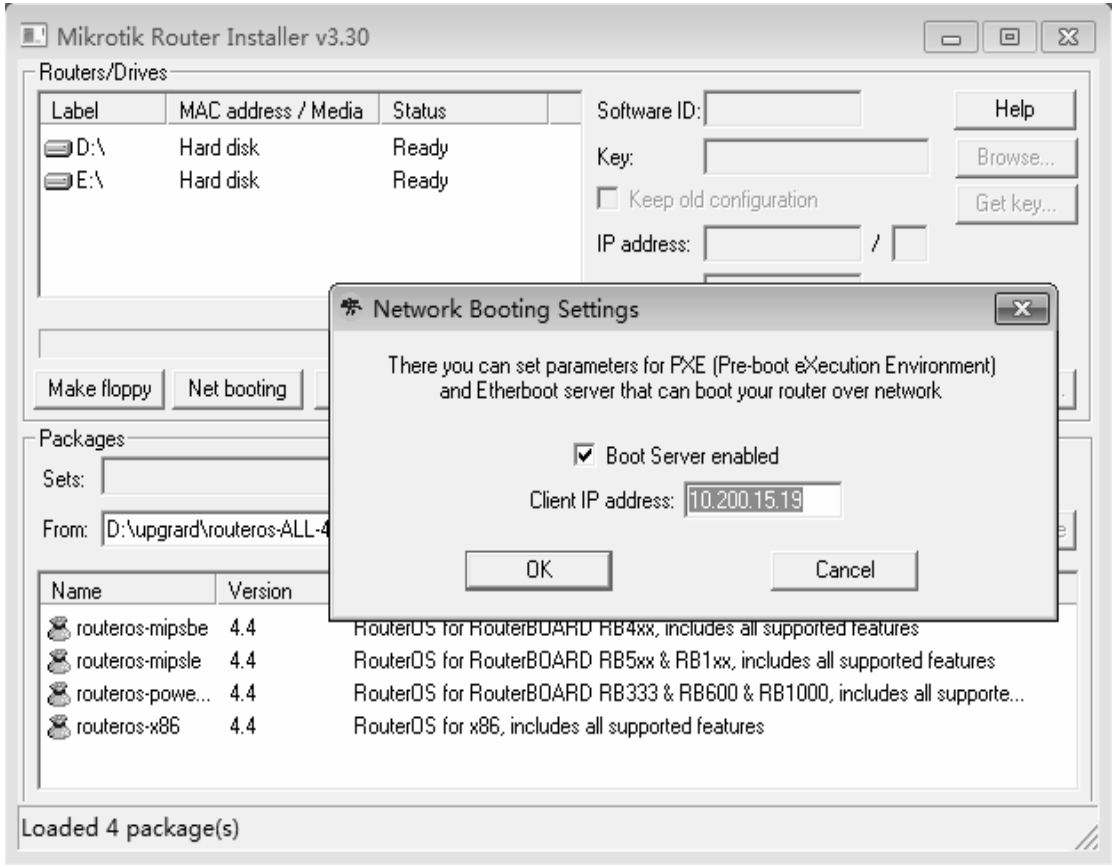
操作步骤2

- 设置好超级终端连接，每秒位数为115200，其他参数为系统默认值：



操作步骤3

- 输入Boot Server客户端的IP地址。临时分配给RouterBoard的IP地址（该事例的地址为**10.200.15.0/24**）



操作步骤4

- 设置**RouterBoard** 从以太网卡引导，进入**RouterBoard BIOS** (重起**RouterBOARD**后，在超级终端下出现提示时**press any key…**后按任意键进入**BIOS**设置)

RouterBOOT-1.13
What do you want to configure?
d - boot delay
k - boot key
s - serial console
o - boot device
u - cpu mode
f - try cpu frequency
c - keep cpu frequency
r - reset configuration
e - format nand
g - upgrade firmware
i - board info
p - boot protocol
t - do memory testing
x - exit setup
your choice:

操作步骤**5**

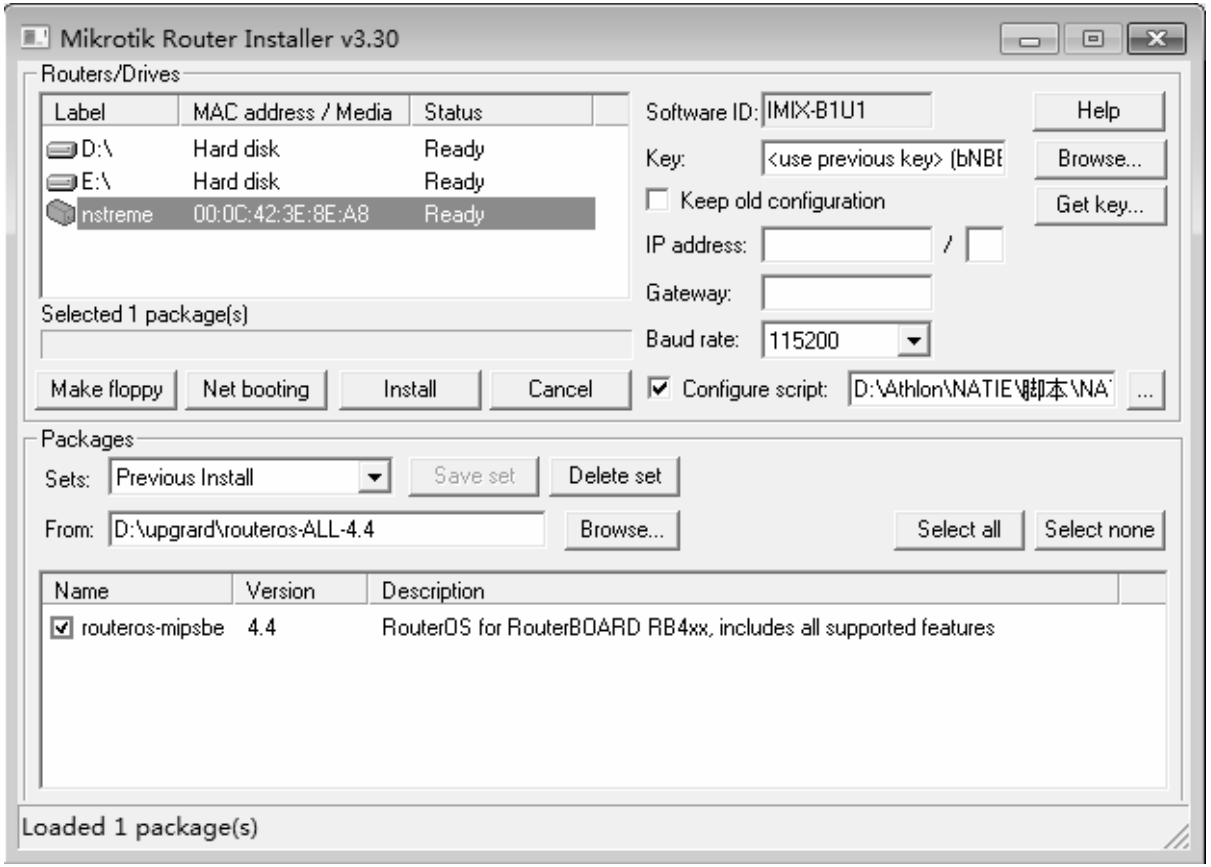
- 进入BIOS后设置引导设备，选择“boot device”，按“o”键可以进入
- 按“e”键，是选择从以太网卡引导RouterBoard：

Select boot device:
e - boot over Ethernet
*** n - boot from NAND, if fail then Ethernet**
c - boot from CF
1 - boot Ethernet once, then NAND
2 - boot Ethernet once, then CF
o - boot from NAND only
b - boot chosen device
your choice: e – Etherboot

返回RouterBoard BIOS首页，选择“x“，退出BIOS。

操作步骤6

- RouterBOARD启动完成后，在netinstall软件中，将会出现一个设备信息，显示当前连接的RouterBoard设备。



操作步骤7

- 设置给路由器新的IP地址和网关。
- 传输的波特率选择115200。
- 开始安装或复位RouterBOARD上的RouterOS
- 安装完成后进入RouterBoard BIOS设置boot from NAND only（仅从RouterBoard的闪存引导）

RouterOS基本配置

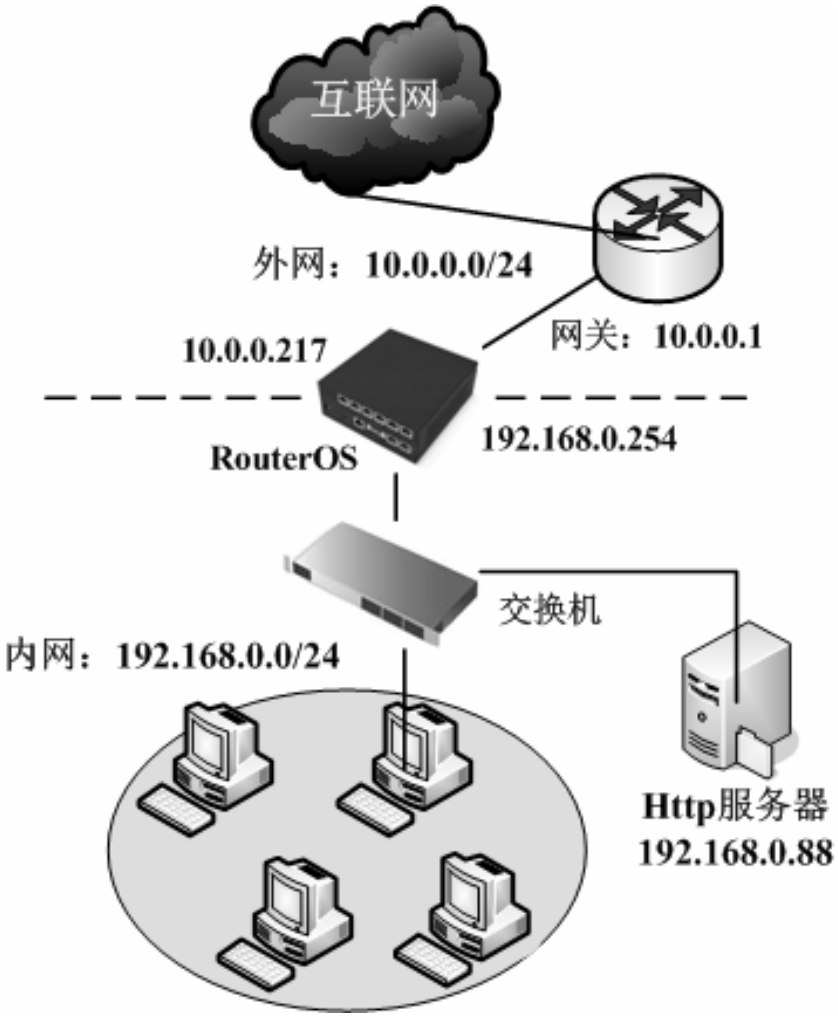
RouterOS的基本配置

- **首先：**启动设备后，检查interface接口网口连接是否正常，并定义网口名称
- **第二：**将对应网口的IP地址配置好，根据情况配置PPPoE拨号功能
- **第三：**配置路由，根据单线和双线配置相应的策略路由规则
- **第四：**配置流量控制规则，分配每台主机流量
- **第五：**配置nat网络地址转换与DNS
- **第六：**绑定ARP列表中的MAC地址

常用的功能

- Winbox登陆常用的功能菜单

RouterOS简单配置实例



- 单线网络的配置
- 配置内网和外网的IP地址，以及路由
- 设置nat的伪装规则
- 设置流量控制规则
- 做http服务器的端口映射

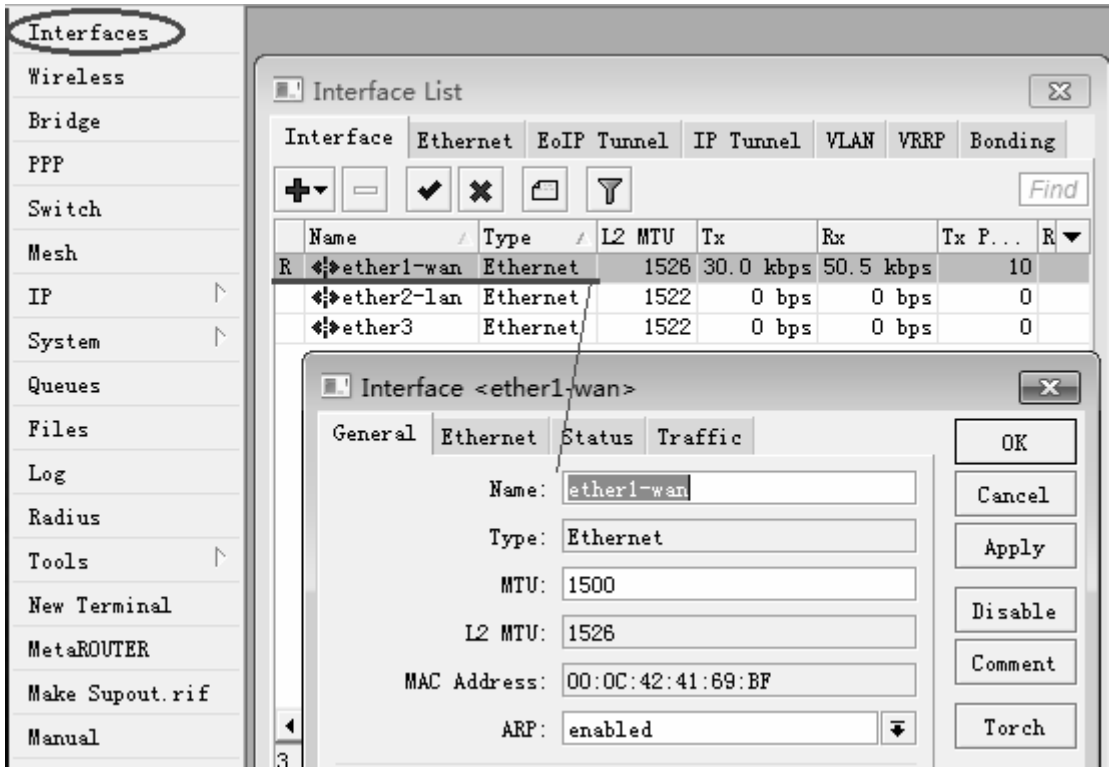
网络环境

在当前的事例中我们使用到两个网络（外网和内网）：

- 内网使用地址为：192.168.0.0子网淹码24-bit（255.255.255.0）。路由器的地址在这个网络中为192.168.0.254
- ISP的网络为10.0.0.0 子网淹码24-bit（255.255.255.0）。路由器的地址是在网络中为10.0.0.217
- 外网DNS为61.139.2.69，202.98.68.96

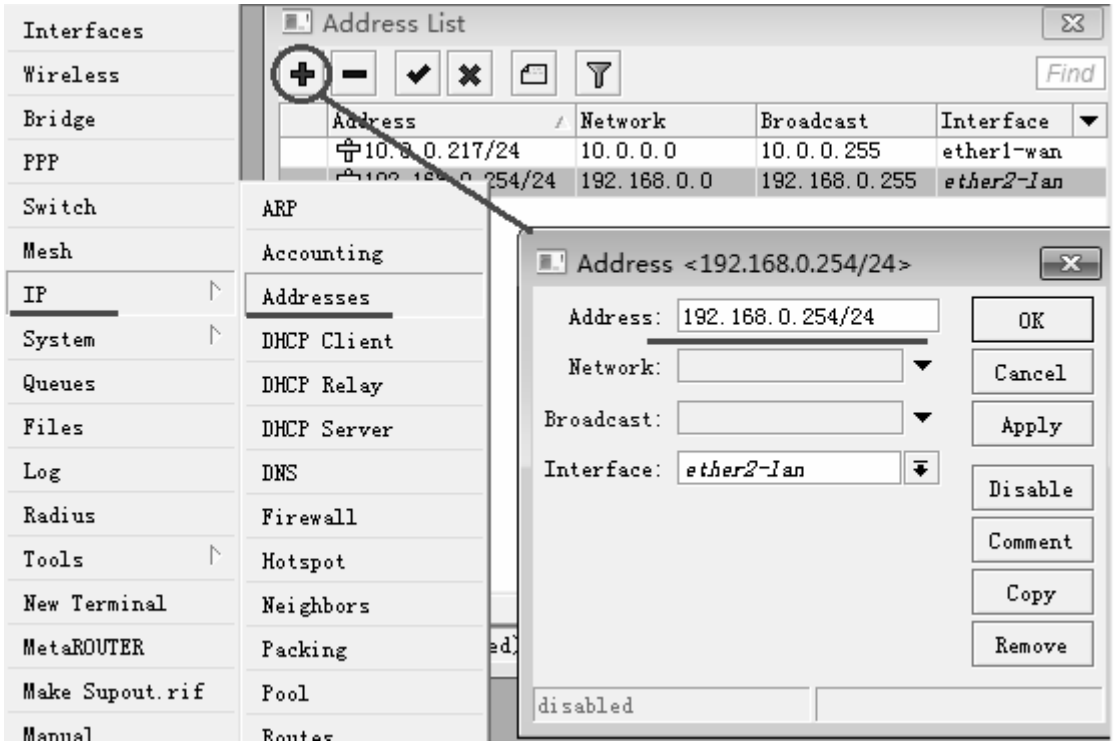
第一步：网络接口配置

- 在/interfaces列表中修改ether1为ether1-wan，定义为外网接口；修改ether2为ether2-lan定义为内网接口，如图：



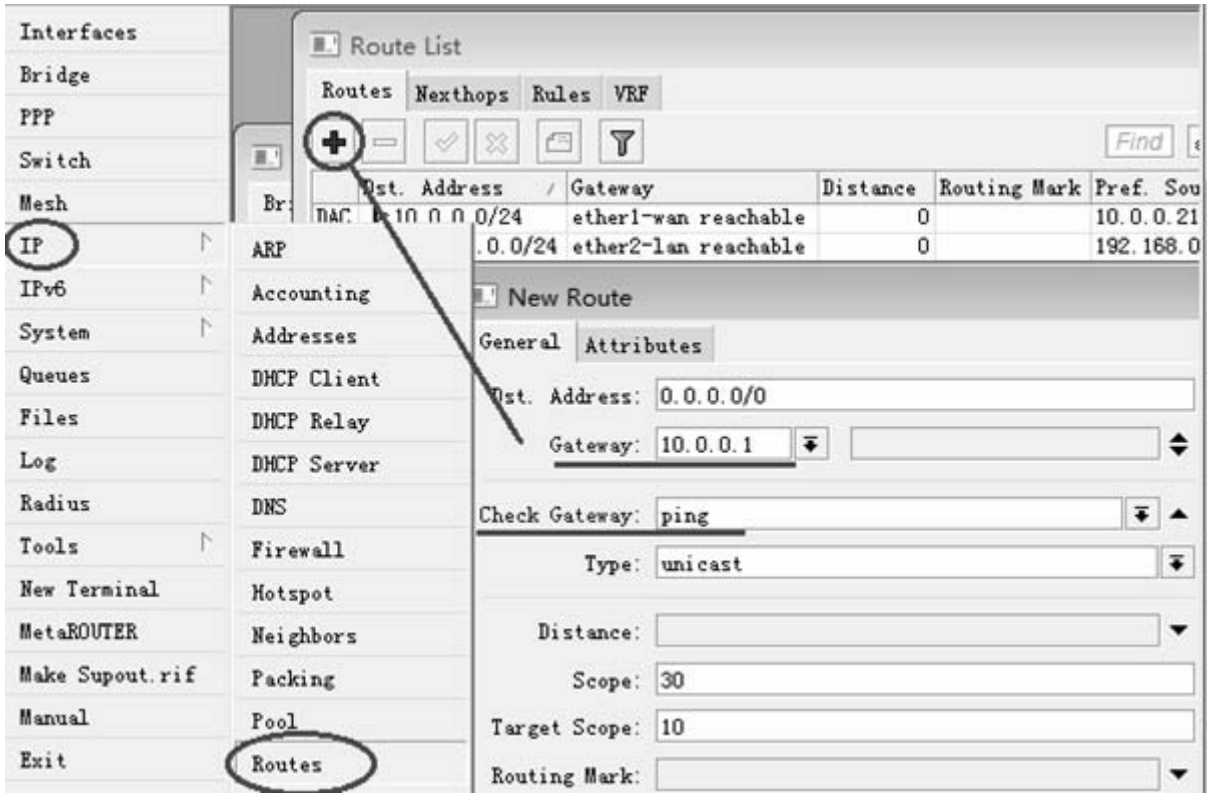
第二步：添加IP地址

- 在/ip address中添加IP地址和选择网卡接口，添加内网和外围的IP地址如图：



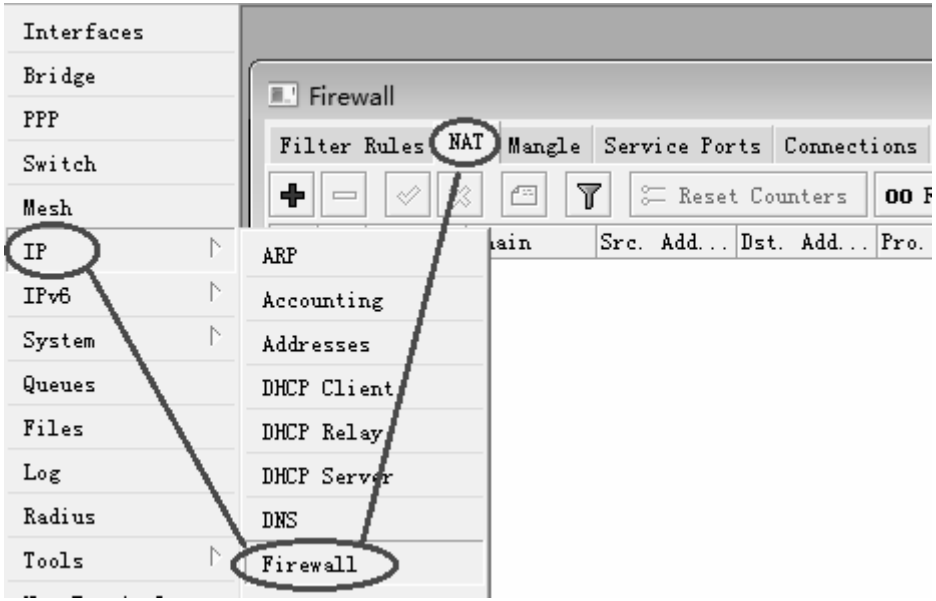
第三步：添加默认网关

- 在/ip routes里添加默认网关10.0.0.1，开启check-gateway=ping（网关ping监测）如图：



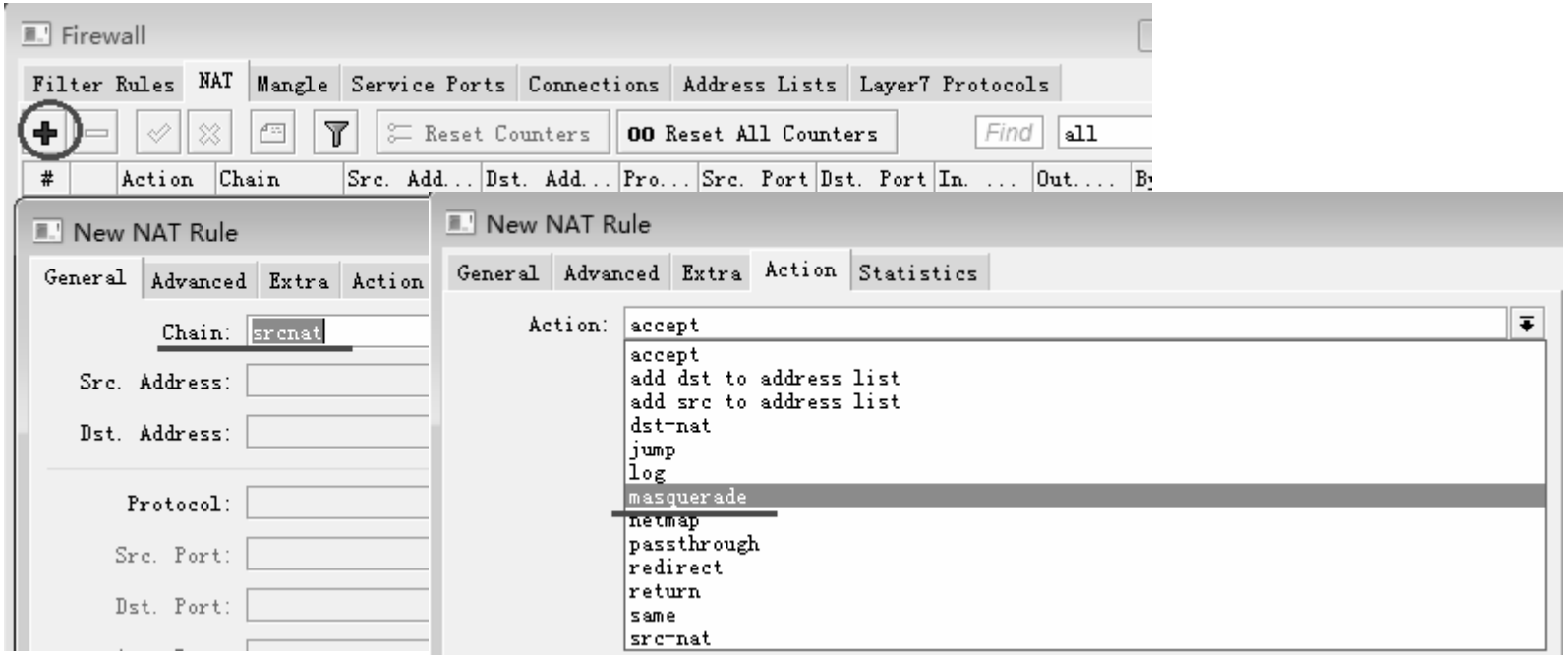
第四步：NAT地址转换 （1）

- 在/ip firewall nat 里点击“+”添加伪装规则，



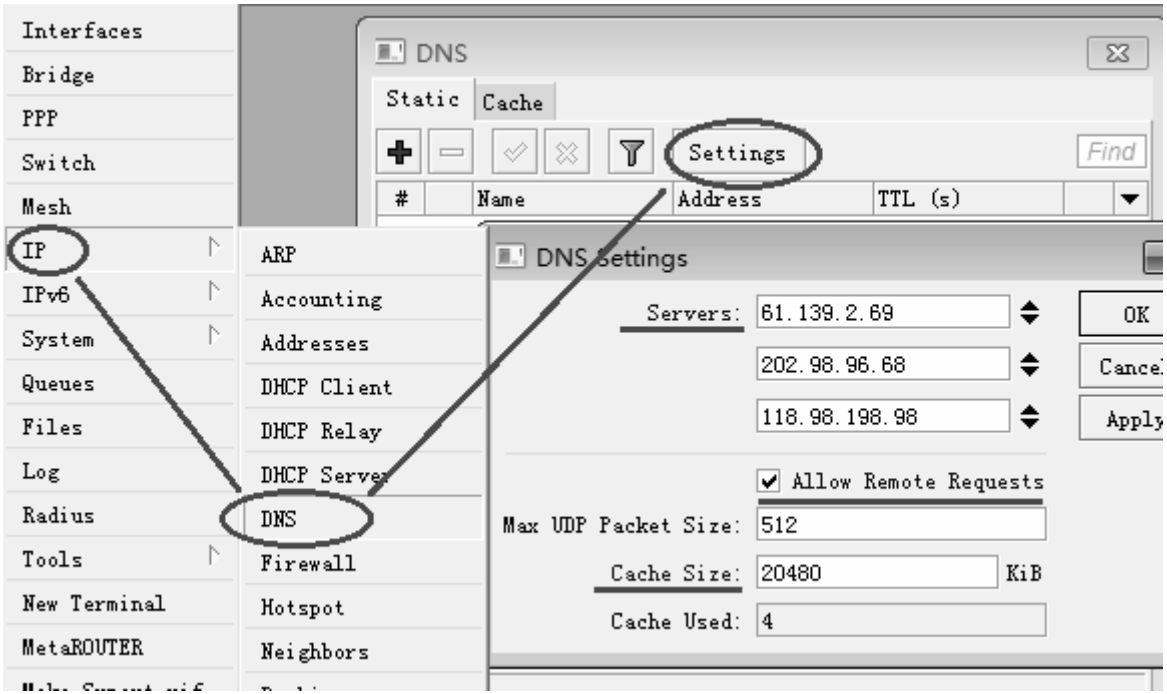
第四步：NAT地址转换（2）

- 在NAT里添加新的规则，在chain里选择srcnat链表，在选择action里的action=masquerade规则：



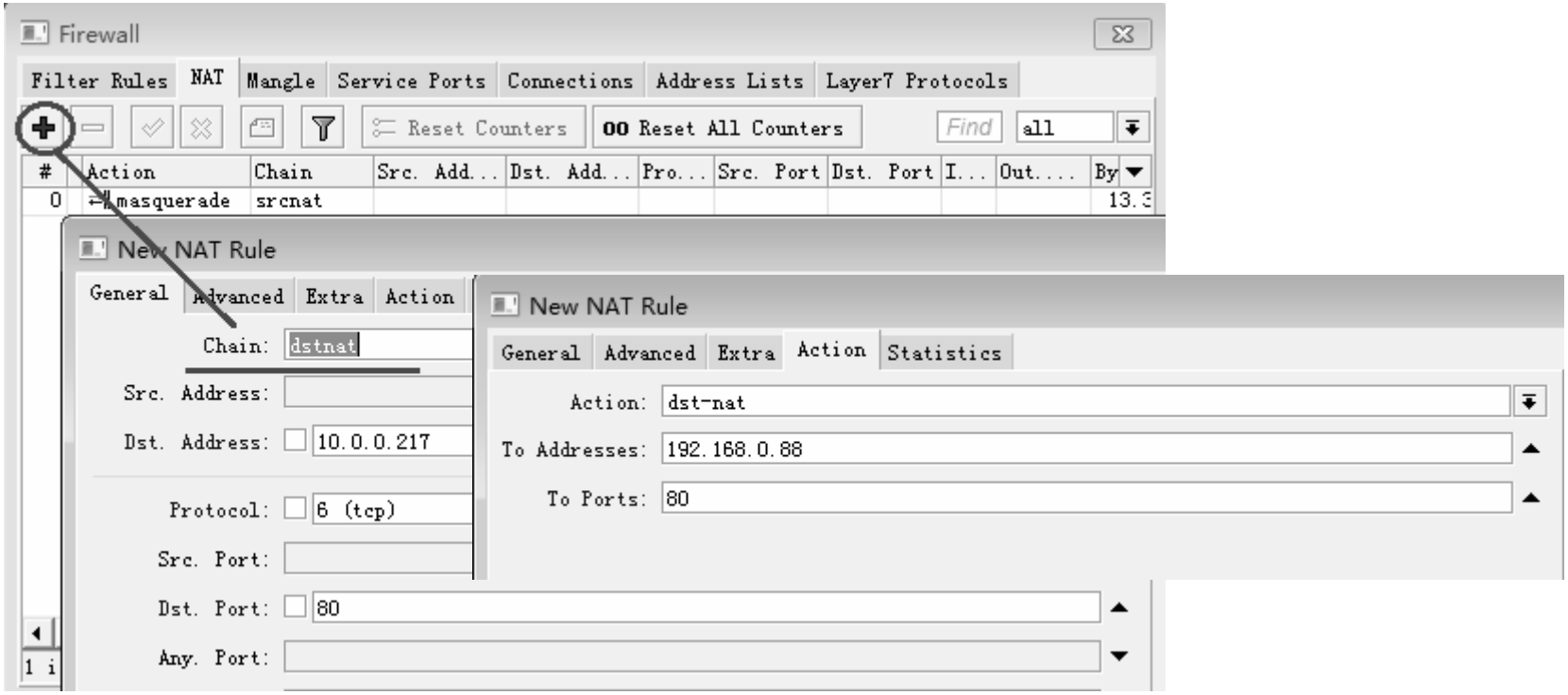
第五步：DNS配置

- 在/ip dns的settings中添加多个DNS服务器地址，根据需要启用DNS缓存（allow remote requests）：



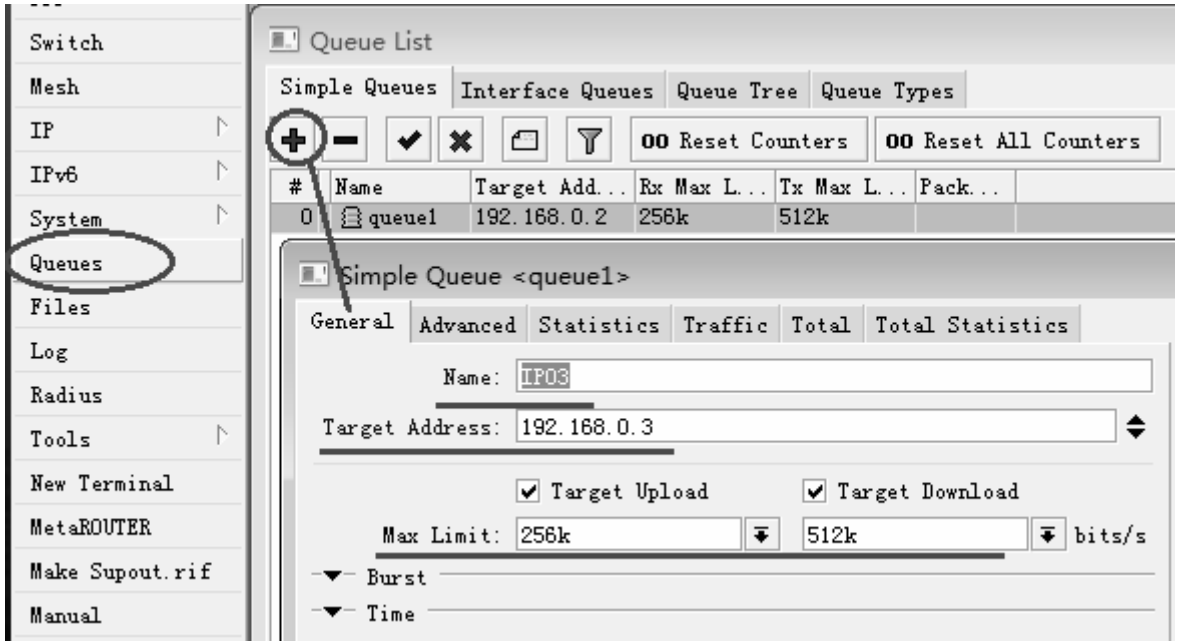
端口映射

- 将内网的http服务器发布到外网，内网的http服务器IP地址192.168.0.88，进入ip firewall nat里，选择chain=dstnat，外网IP地址是10.0.0.217配置到dst-address，dst-port为tcp协议80端口，下图：



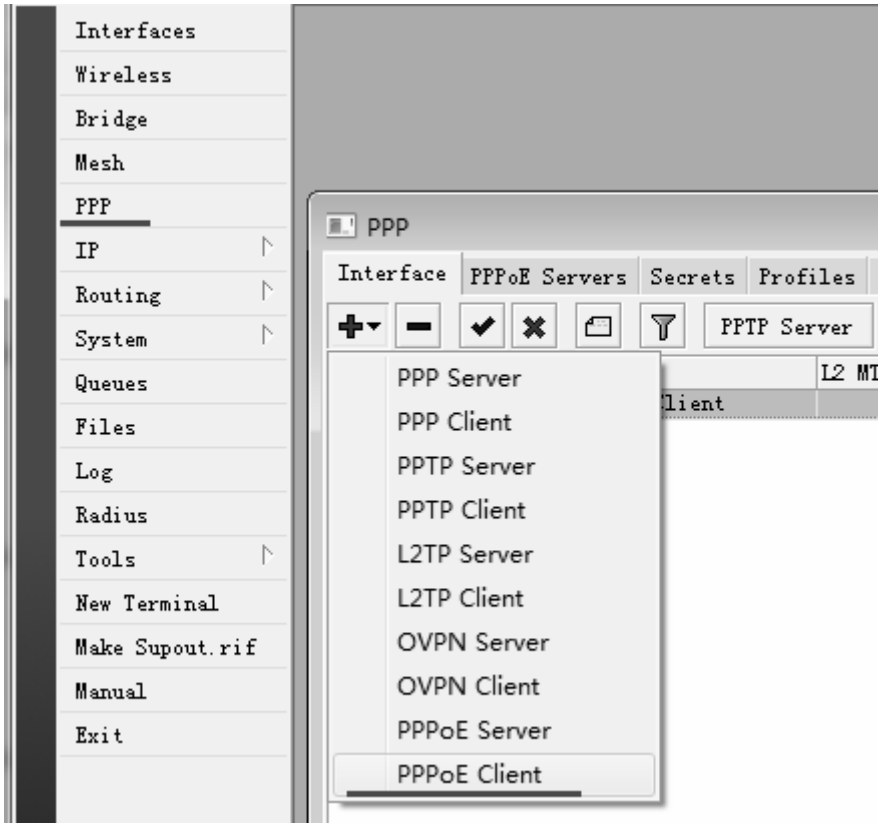
单机带宽控制

- 进入Queue添加带宽控制规则，选择simple queue，添加主机IP是192.168.0.3，并取名为IP03，设置带宽为上行(upload)256kbps，下行(download)512kbps



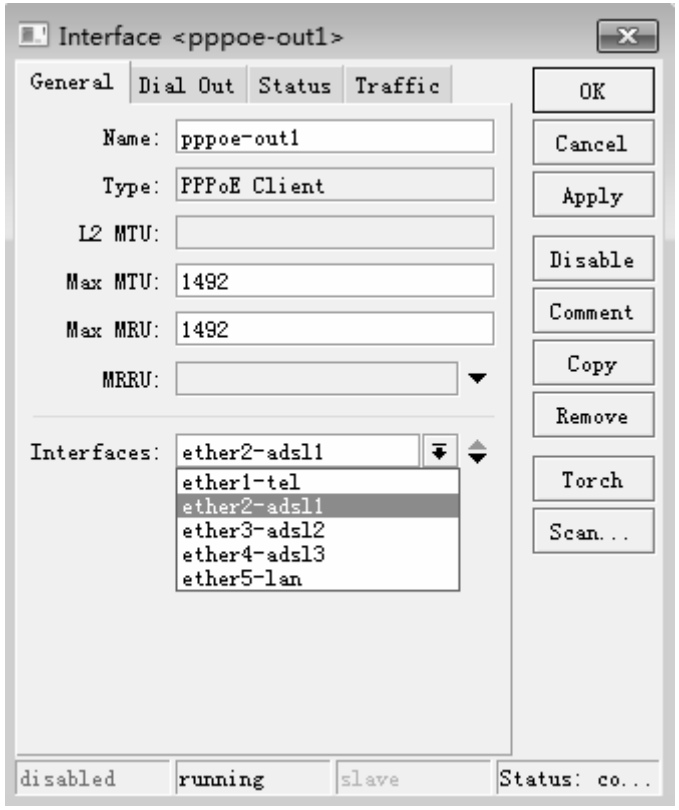
ADSL拨号配置 1

- 进入PPP目录下，添加选择PPPoE-Client，添加ADSL拨号



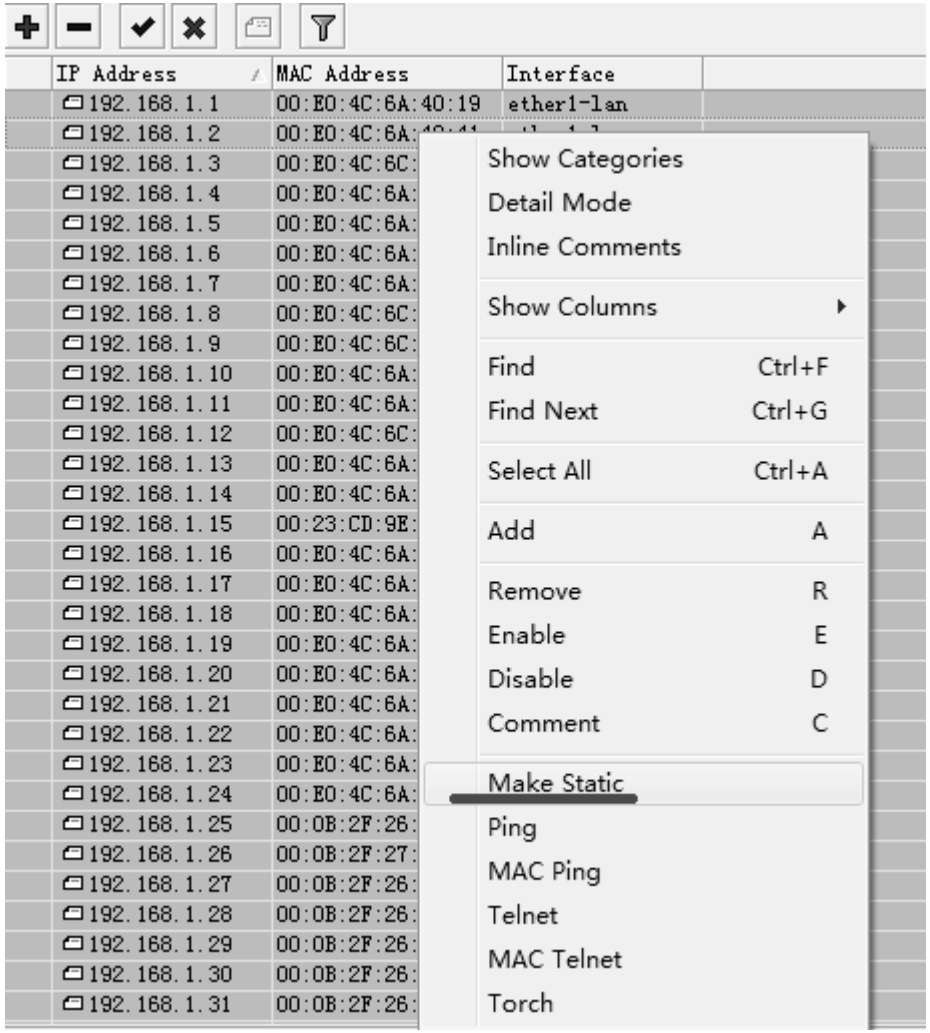
ADSL拨号配置 2

- 在PPPoE拨号中选择拨号网卡（interface），并配置账号（user），密码（Password），其他配置参照下图：



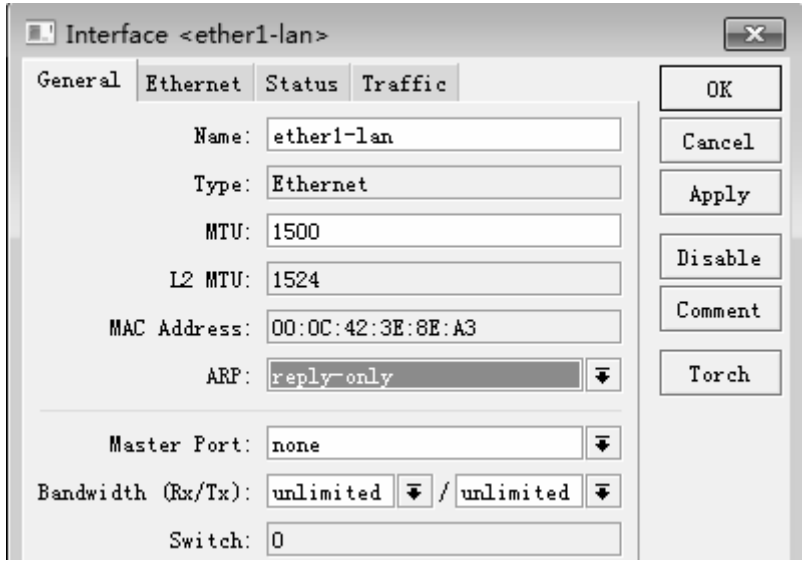
配置MAC地址绑定 1

- 进入ip arp配置MAC地址列表，确定所有地址进入ARP列表后，点右键，选择Make Static。
- 也可以点加号，手动添加。



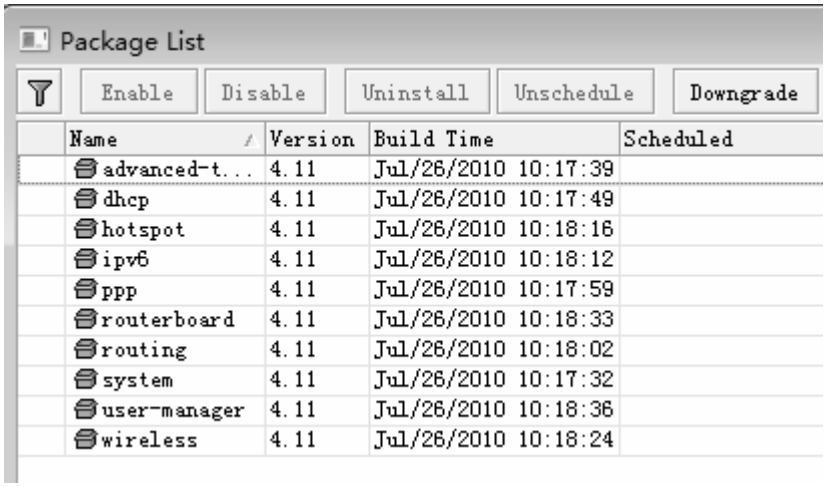
配置MAC地址绑定 2

- 进入interface下，选择ether1-lan的接口，将ARP属性选择为reply-only，仅回应ARP列表中静态的MAC地址.



软件升级和功能包管理

- 进入winbox中的”system>packages”管理功能包；
- RouterOS能禁用和启用功能包来控制相应的RouterOS功能；
- 安装和卸载功能包，升级最新的软件版本，或者降级老的版本；



RouterOS升级与降级

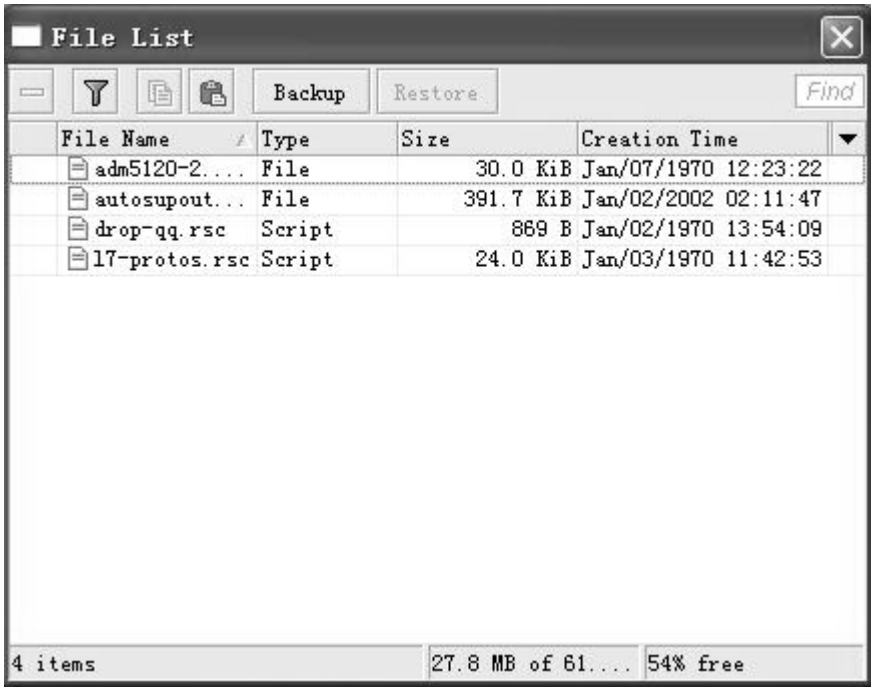
升级**RouterOS**

- 在- 在“system>license”中检查你的License（许可）是否允许升级
- 上传最新版本的RouterOS软件功能包
  - FTP方式
  - winbox中files窗口中，拖-拉-放的方式
- 通过system reboot命令正常重启路由器

降级**RouterOS**

- 上传一个较早RouterOS版本功能包
- 进入“system>packages”，并点击“downgrade”重启路由器安装老的功能包
  - 注意，仅选择“system>reboot”是不会安装老的功能包，你需要先进入“packages”窗口点击“downgrade”

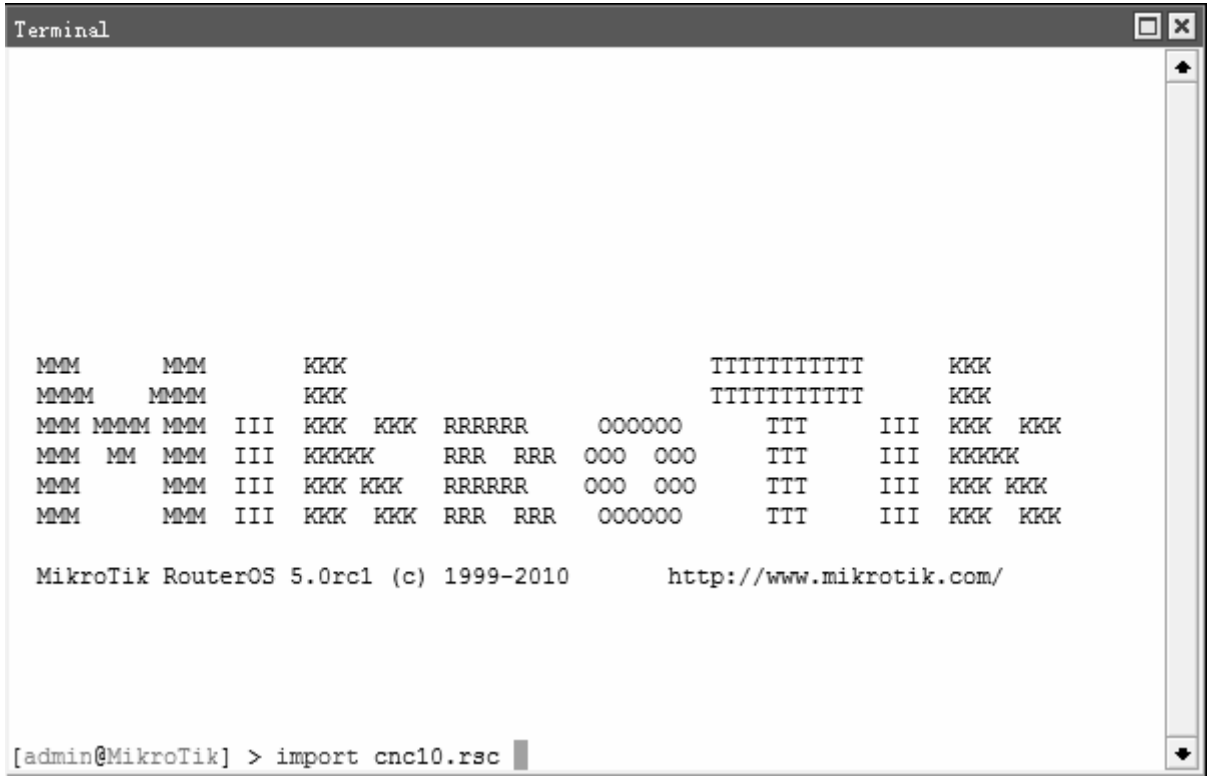
RouterOS文件操作



- 在files目录下可以对各种文件进行管理；
- 一般存放升级文件或者脚本文件，以及Hotspot文件等；
- 脚本和升级文件必须放在根目录下，否则不能运行；

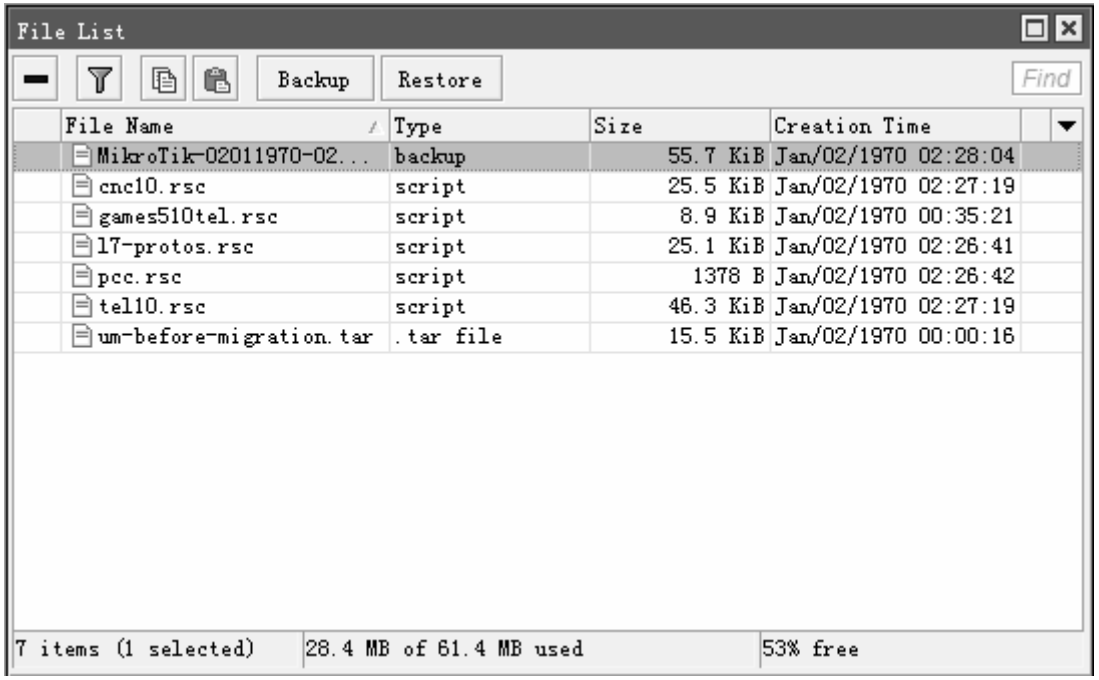
Import和export

- Export能将一部分功能从RouterOS导出，存放在Files目录下
- Import通过已有的或者编辑好的脚本导入RouterOS，文件同样存放在Files下



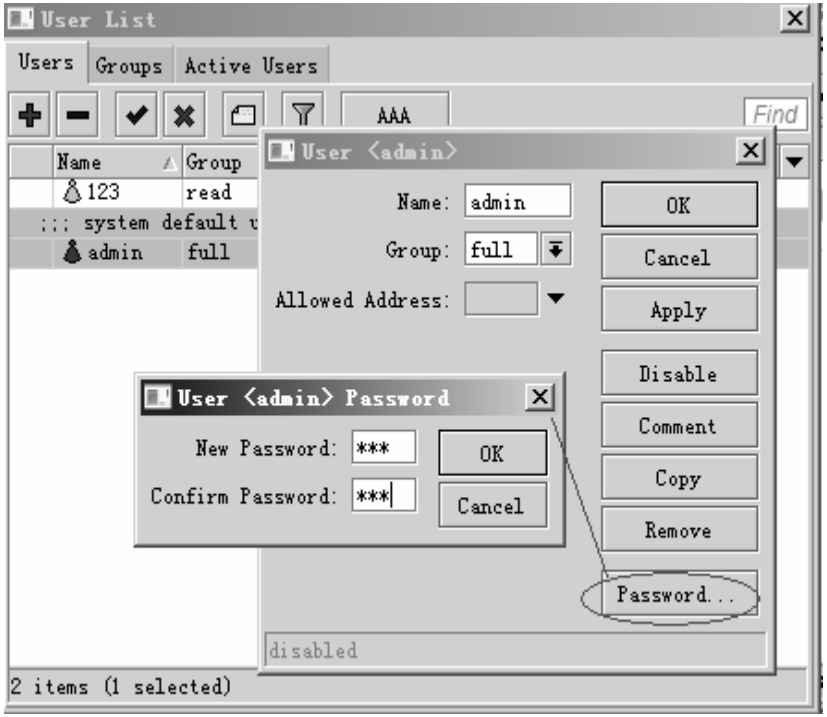
备份与恢复

- 将系统的配置参数备份，并恢复
- 进入/files中，使用backup 命令备份系统配置，使用Restore恢复配置
- Backup是导出系统的配置为二级制文件，无法编辑。而Export导出的是脚本，能通过管理员的需要编辑修改，然后通过import导入。



用户登录名管理

- 全新安装的RouterOS后,默认用户登陆名为”admin”
  - 默认登陆密码为空
  - “admin”属于“full”组
  - “full”组是最大的权限
- 安全管理路由器
  - 所有的管理用户都应设置密码
  - 添加新的管理用户
  - 根据需要设置新用户的分组
- 如果你忘记了密码，你只能重新安装系统

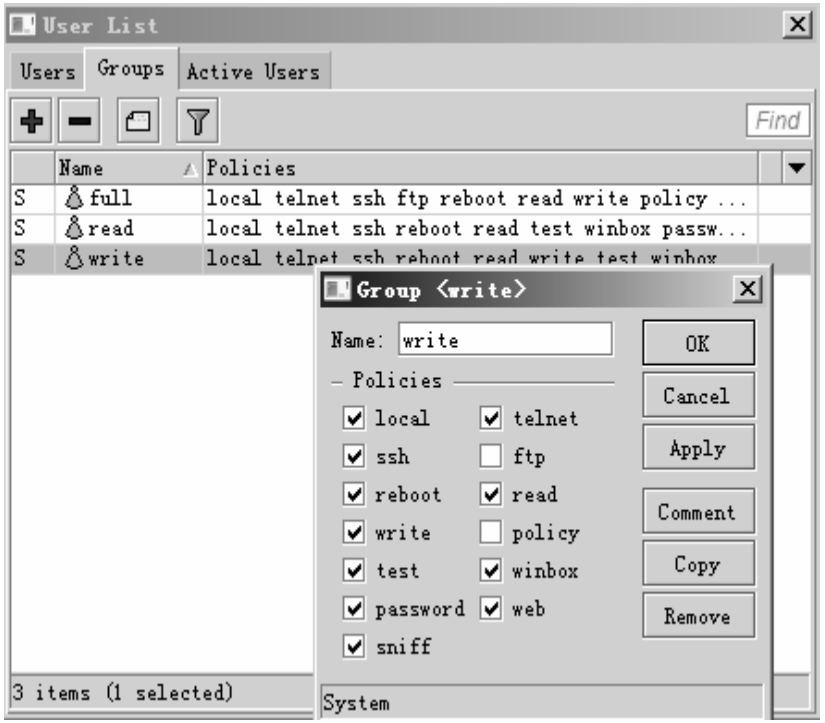


在/user中通过password修改密码

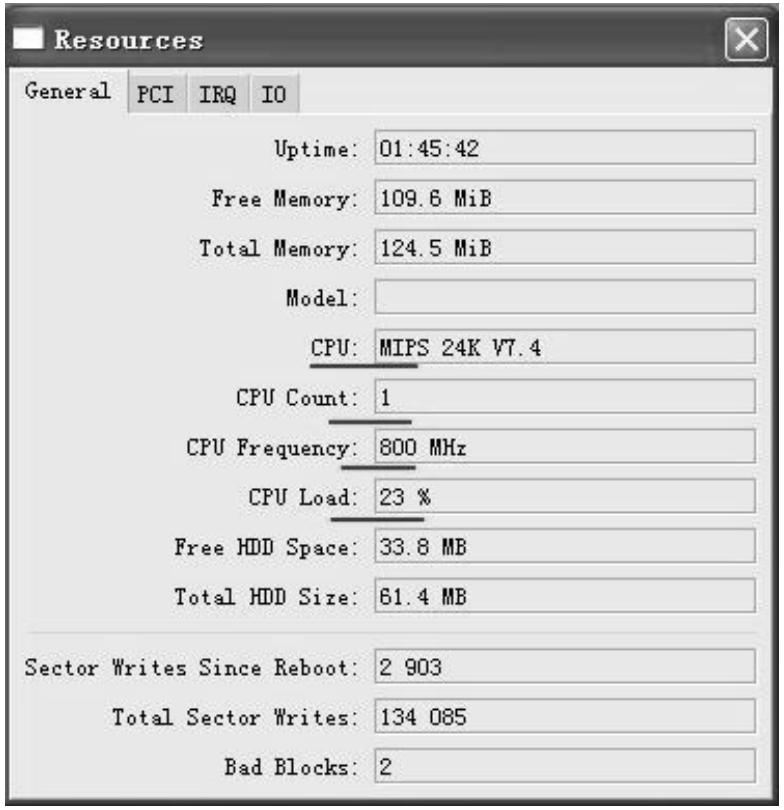
通过allowed address限制登陆IP地址

用户组设置（User Groups）

- 系统有三个默认的分组，并能修改他们，但不能删除
  - “full” 所以的权限
  - “read” 只读权限
  - “write” 读/写的权限，但不能修改用户帐号
- 你可以添加自己指定的用户分组，并可以设置他们的规则
- 你可以修改已经存在的用户分组。

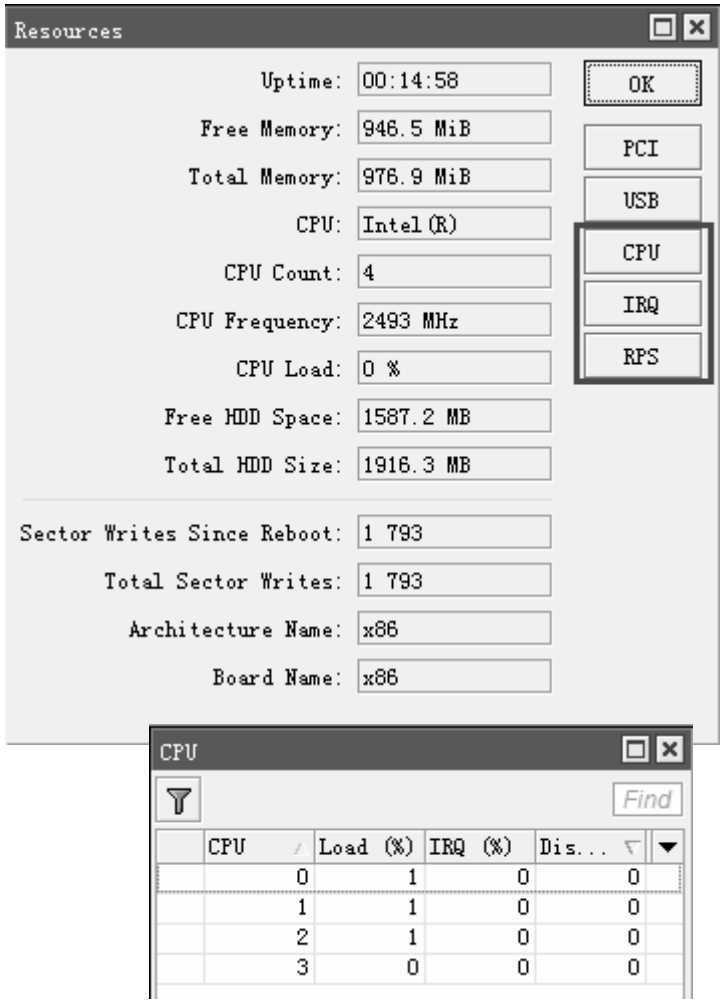


RouterOS资源管理



- /system resource可以查看系统信息和资源使用情况
- 查看路由器运行时间、内存、CPU、CPU频率、CPU占用情况和硬盘使用情况

RouterOS 5.0rc资源管理



- 从RouterOS v5.0beta6开始支持16核心的处理器，之前的版本限制为8核心，能分配不同的CPU给每个网卡的IRQ，提升性能和负载均衡
- 能查看每个CPU核心的负载情况
- 支持基于Linux新的NAPI技术和RPS控制
- 对多CPU核心的负载优化

RouterOS的CPU调用

- 一般的错觉是认为路由器在CPU 70%时工作缓慢（大延迟、高丢包率，低转发能力），其实只要负载不到100%的CPU有能力处理所有数据，因此1%, 17%, 50%或者98% - 这些处理计算不会存在延迟。

- 但要注意这个对于多核处理器系统除外，例如我们在使用普通单核心P4处理器时，我们看到的是正常的CPU使用率，但当我们使用core2双核心处理器时，我们可能看到的是双核心CPU的平均值（RouterOS 5.0前的版本），在多核心处理可能出现一个核心100%，另外一个核心没有使用的情况，这样我们看到CPU的使用率是50%左右，但结果是路由器已经出现不稳定情况，如掉包等情况。

RouterOS常用端口 1

- 20/tcp 文件传输协议 FTP [数据连接]
- 21/tcp 文件传输协议 FTP [控制连接]
- 22/tcp 安全命令行解释 SSH 远程登录协议 (仅与安全封装一起)
- 23/tcp 远程通信网络协议telnet
- 53/tcp 域名服务器 DNS
- 53/udp 域名服务器 DNS
- 80/tcp 万维网（WWW）HTTP
- 67/udp 自举协议 或 DHCP 服务器 (仅与 dhcp 功能包一起)
- 68/udp 自举协议 或 DHCP 客户 (仅与 dhcp 功能包一起)
- 1723/tcp 点对点隧道协议 PPTP (仅与ppp功能包一起)
- 5678/udp MikroTik Neighbor Discovery Protocol
- 2000/tcp 带宽测试端口
- 3986/tcp Winbox代理
- 8291/tcp Winbox
- 20561/udp MAC winbox

命令行接口(CLI)
command line interface

登陆后，在当前的操作目录中按[?]可以查看到所以命令
- [admin@MikroTik] > [?]

连续按[Tab]键两次，能显示当前目录下可以获得的命令
- [admin@MikroTik] > [Tab] [Tab]

命令输入可以不用完整，通过[Tab]键可以帮助你完整输入
- [admin@MikroTik] >ip add [Tab]
- [admin@MikroTik] >ip address

如果按一次[Tab]键，没有显示任何信息，再连续按两次，会看到可获取的选项提示
- [admin@MikroTik] > i [Tab][Tab]
  import interface ip
- [admin@MikroTik] > in [Tab]
- [admin@MikroTik] > interface

CLI路径操作

你可以一步一步的进入每个目录
- [admin@MikroTik] > ip [Enter]
- [admin@MikroTik] ip > address [Enter]
- [admin@MikroTik] ip address > print [Enter]

使用**[..]**返回到上一级目录
- [admin@MikroTik] ip address> .. [Enter]
- [admin@MikroTik] ip >.. [Enter]
- [admin@MikroTik] >

使用**[/]**返回到根目录
- [admin@MikroTik] ip address > / [Enter]
- [admin@MikroTik] >

Print和Monitor

在CLI中，Print命令是最常用的，用于输出一个项目列表和输出许多修改参数情况，例如：

- print “显示当前菜单下的信息”
- print status “显示当前菜单下的状态信息”
- print interval = 2s “每间隔2秒刷新显示当前菜单下的信息”

使用‘print ?’查看可以获得的显示参数

‘monitor’连续显示项目的状态

- /int eth monitor ether2 “在interface ethernet下监测ether2每秒流量状态”

add、set和remove

- 使用“add”命令创建一个新的项目，并设定项目中的参数

[admin@Office] /ip address> add address=192.168.10.1/24 interface=lan

- 可以通过”set” 命令修改以存在项目中的参数

[admin@MikroTik] interface> set ether1 name=Local; set ether2 name=Public

- 或者使用”remove”命令删除不需要的项目

[admin@Office] /ip firewall filter> remove 2

enable 、disable、move

- enable 启用命令：

[admin@MikroTik] interface> enable ether2

- disable 禁用命令：

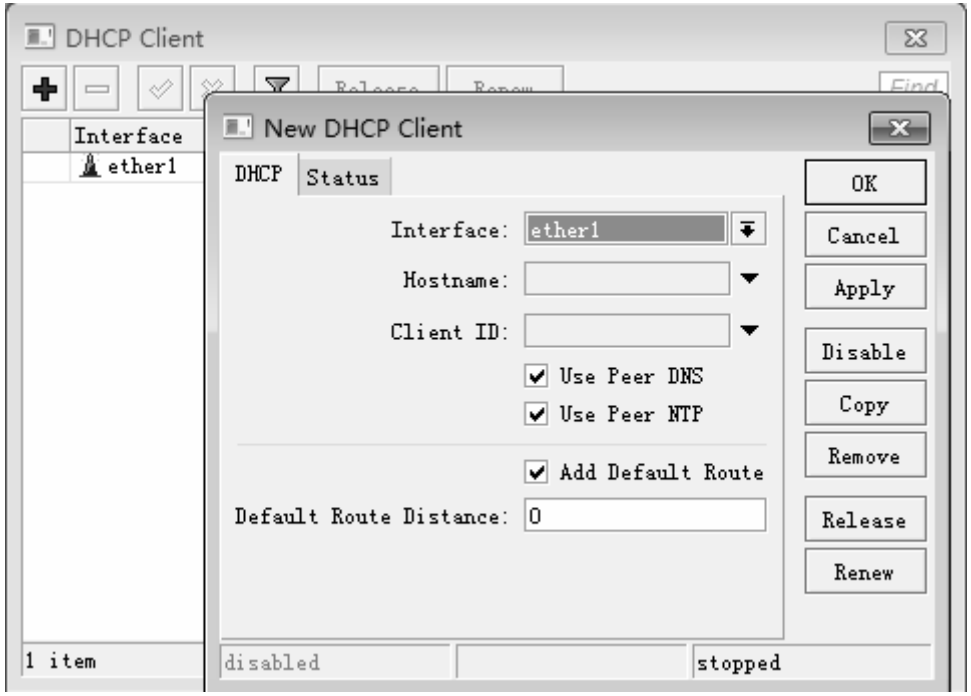
[admin@MikroTik] interface> disable 0

move命令，用于移动规则，如在防火墙里调整规则的顺序

/ip firewall filter move 10 0 “移动标号10的规则到0”

DHCP-Client

- 当我们上级网络设备分配了动态IP地址，我们需要用DHCP客户端自动获取；
- 我们只需要进入IP菜单下选择dhcp-client，这里我们只需要选择intercace到对应的网卡，其他参数默认



DHCP-Server

- 当我们需要将向局域网内动态分配IP地址，我们可以使用DHCP服务器

- 建立DHCP服务器的流程

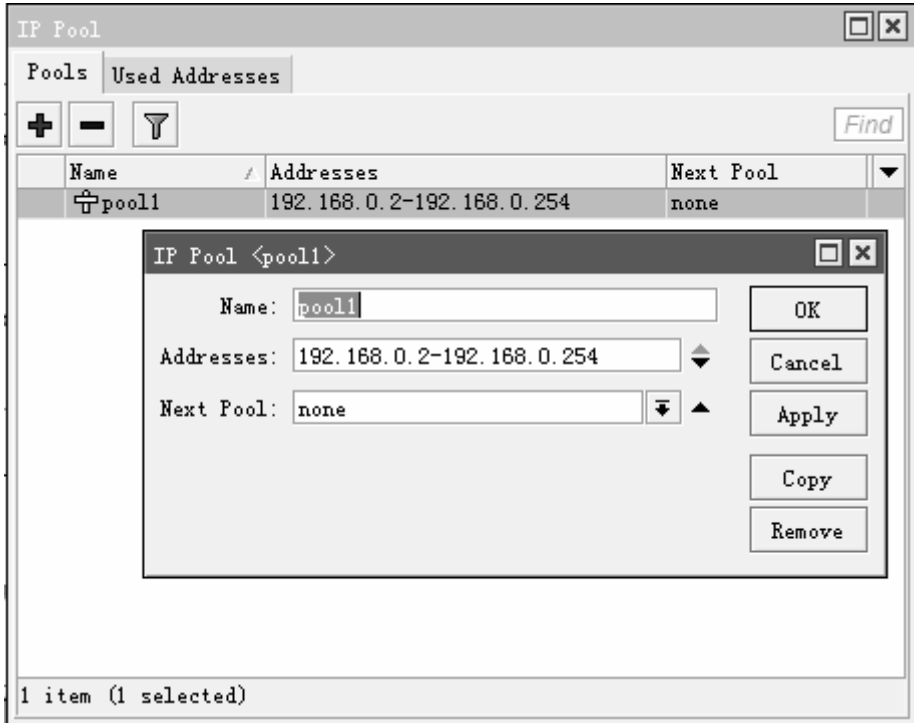
1、根据局域网的IP地址段，分配地址池，ip pool

2、进入ip dhcp-server添加dhcp服务器规则

3、建立对应的network网络参数

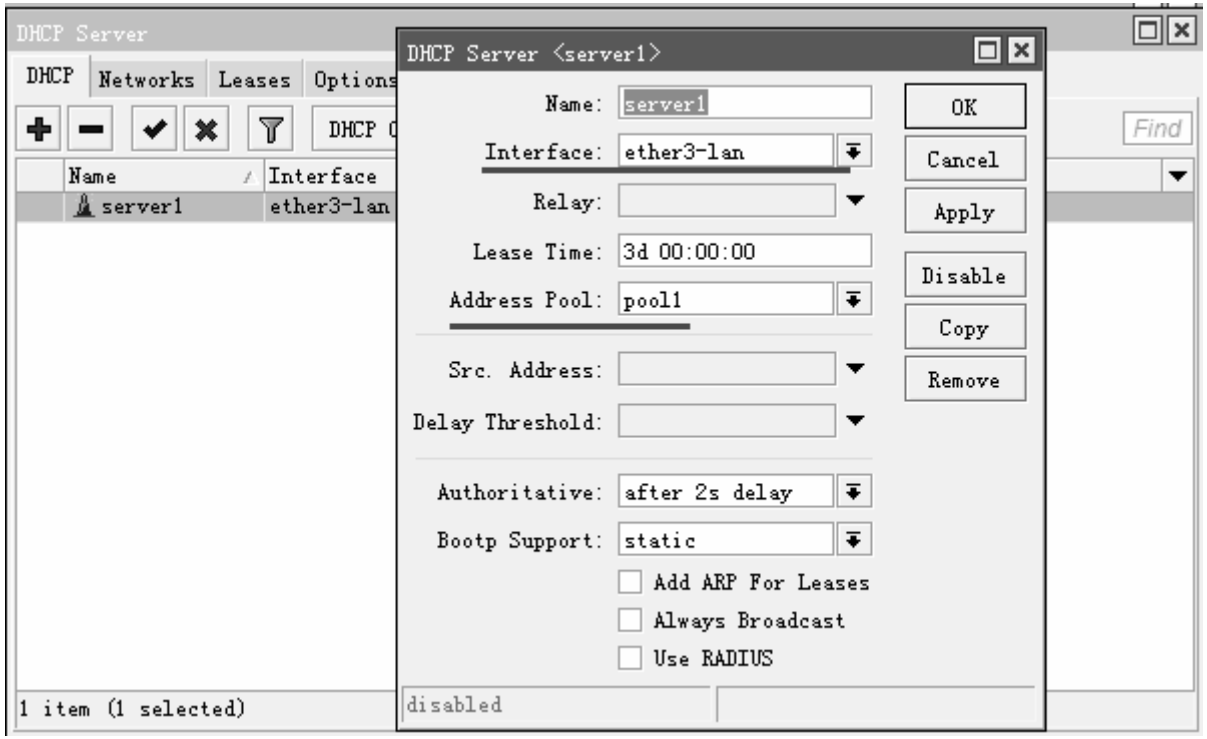
DHCP-Server设置 1

- 假设我们的路由器的内网地址是192.168.0.1/24
- 首先我们进入ip pool添加一个地址池，范围从192.168.0.2-192.168.0.254(排除路由器内网IP地址)



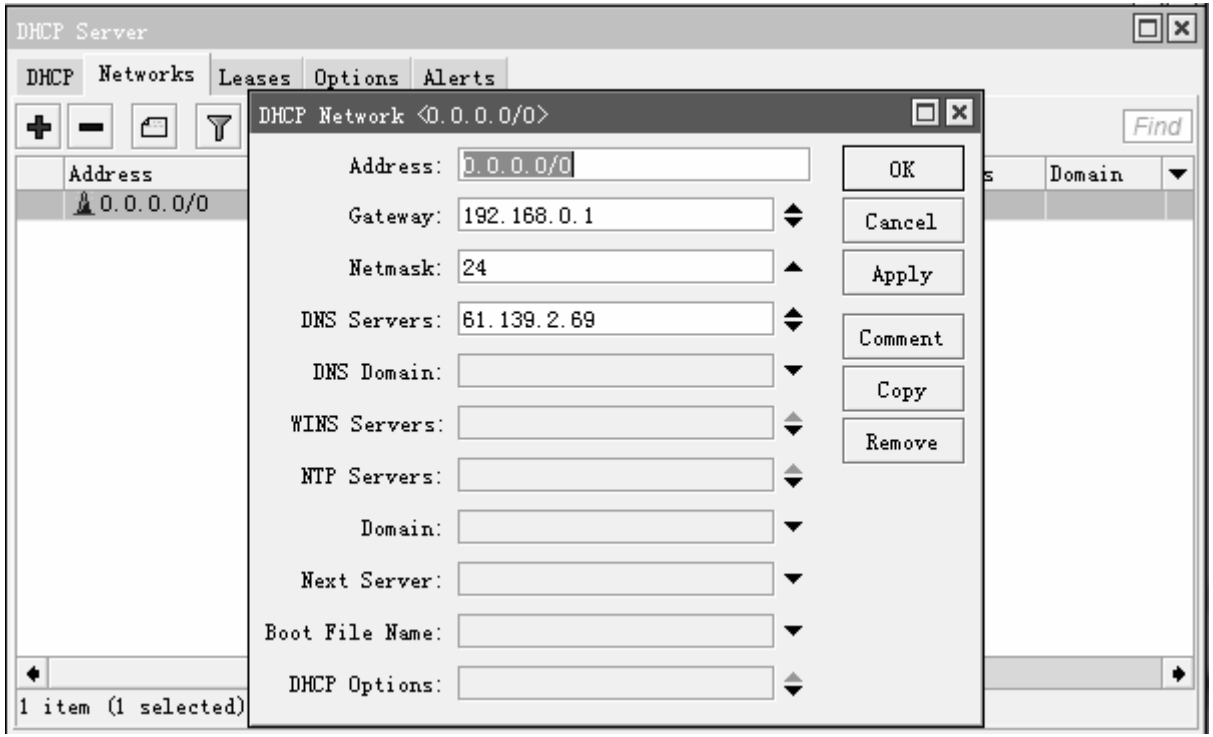
DHCP-Server设置 2

- 进入/ip dhcp-server添加DHCP服务器接口到ether3-lan和设置对应的地址池pool1

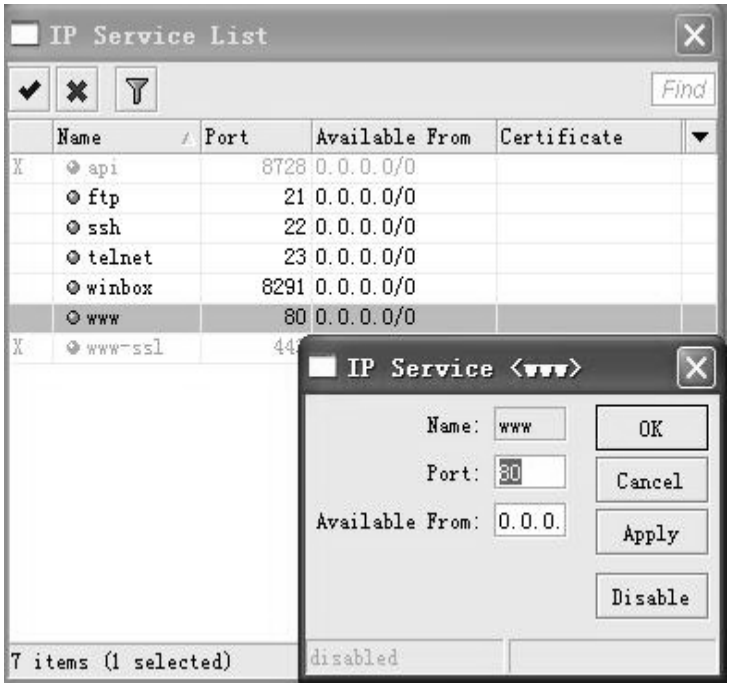


DHCP-Server设置 3

- 进入network标签，添加对应的dhcp服务器的网关、子网掩码值和DNS服务器



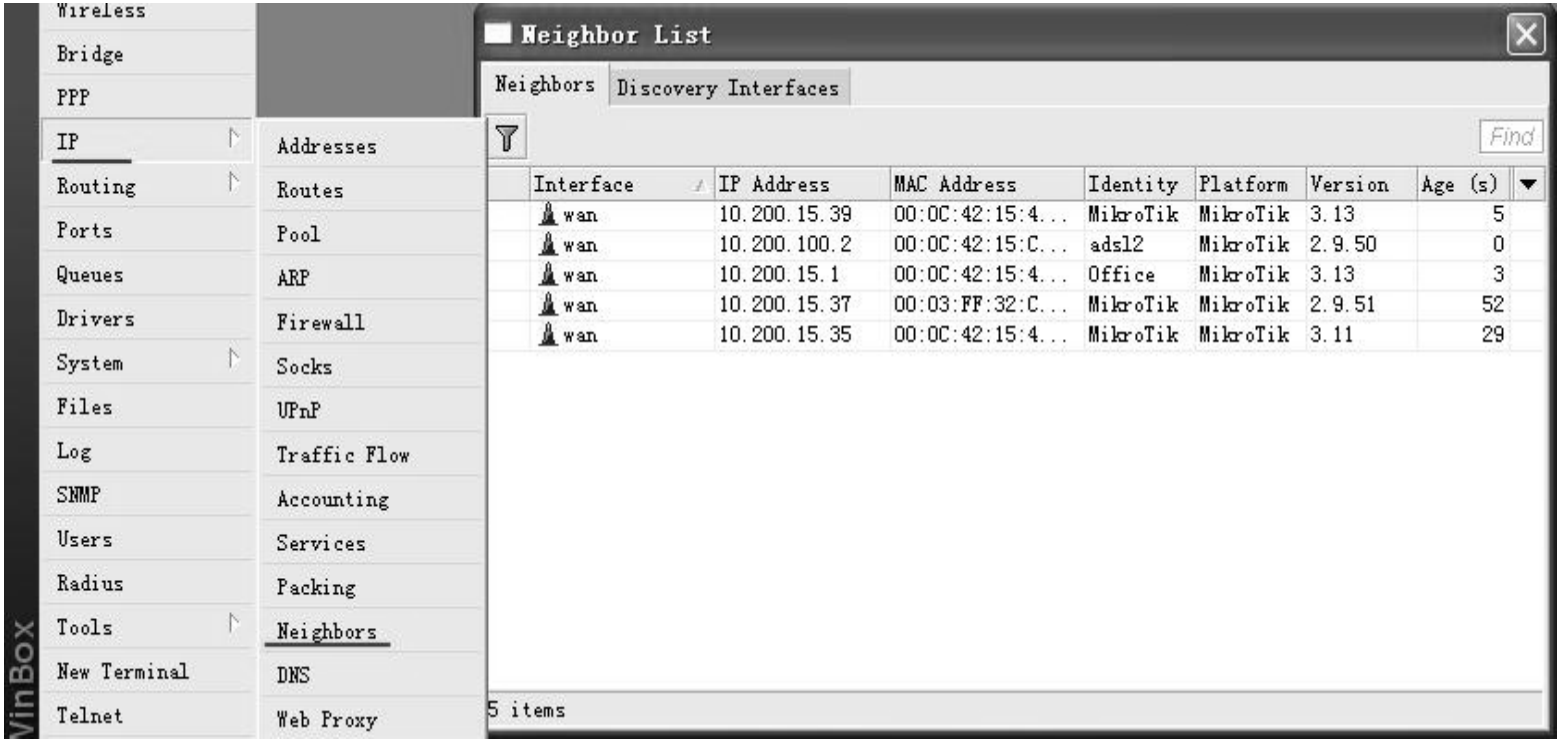
IP servcie



- 用于管理RouterOS的各种服务端口；
- 可以禁用启用相应端口，也能修改端口；
- 有效的管理端口，能提高路由器的安装性。

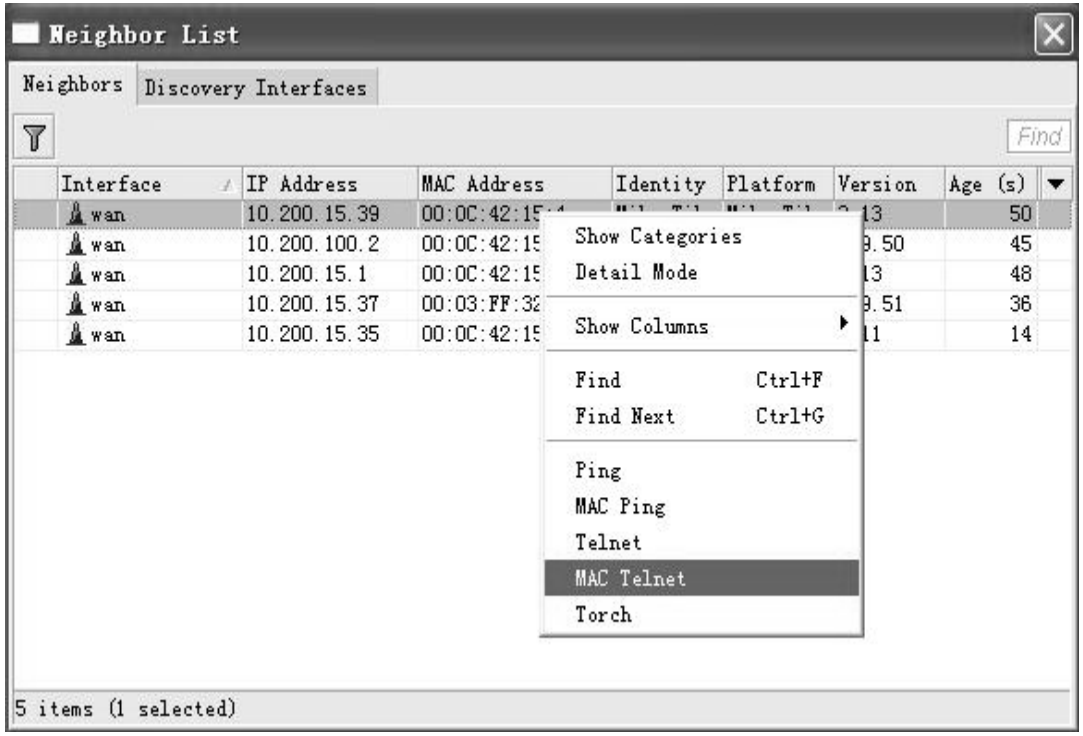
邻居 neighbor

- ip neighbor用于查看同一网络内的设备（显示MNDP (MikroTik Neighbor Discovery Protocol) 或CDP (Cisco Discovery Protocol) 设备 ）



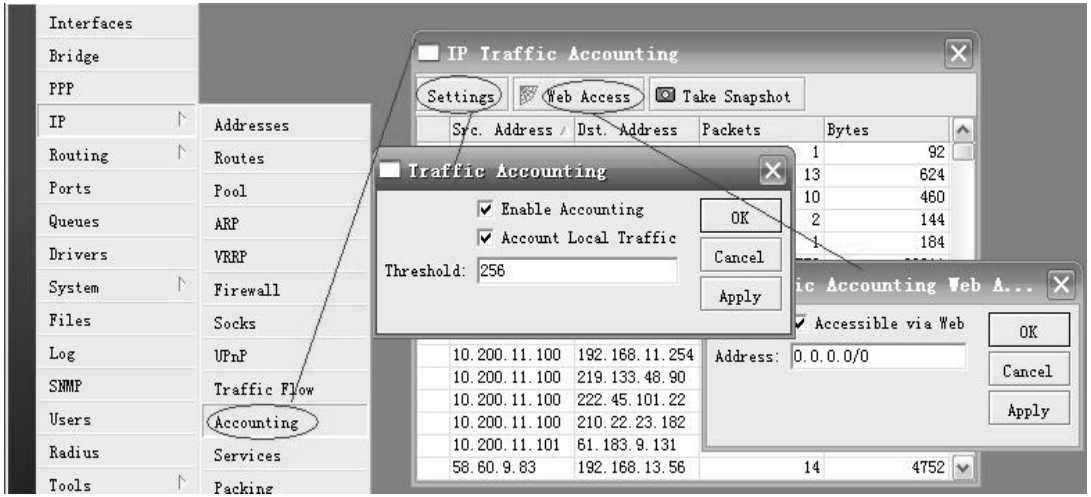
Neighbor 操作

- 在邻居列表中可以对每个设备进行管理，也可以通过MAC 或者IP地址的Telnet方式登陆邻居设备

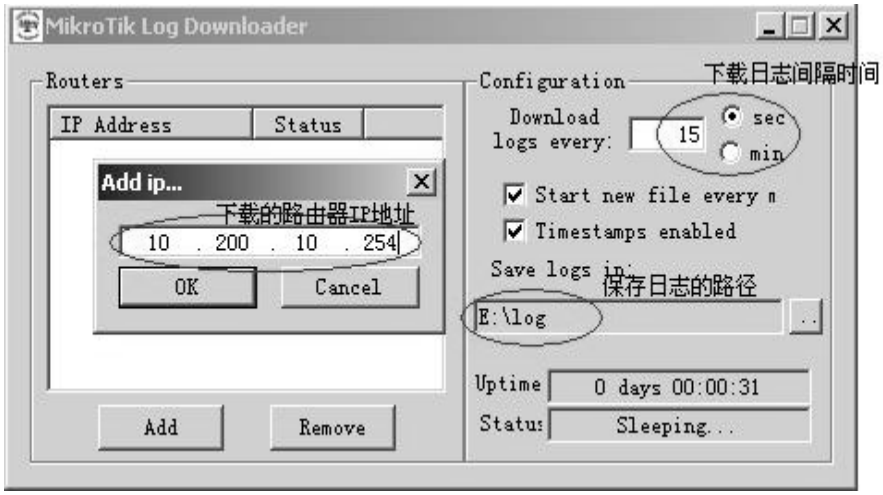


ip accounting日志记录

- 用于记录访问日志信息
- Web浏览器或wget可以连接到URL：**http://routerIP/accounting/ip.cgi**
- 能将所有的访问日志，通过Log Downloader软件下载到windows电脑上



在**ip accounting**中启用访问日志记录



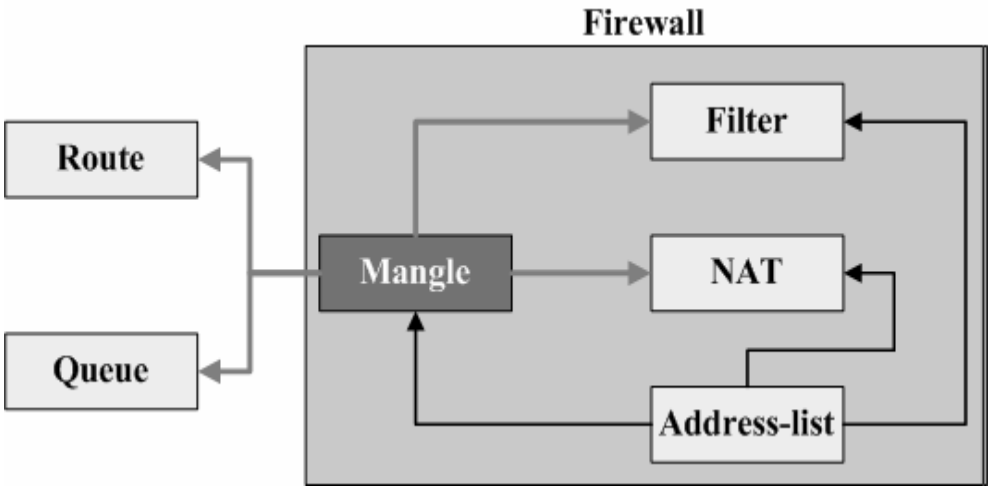
在**windows**中启用**log Downloader**工具，并配置

RouterOS IP工作流程
与防火墙

Mangle的作用

- RouterOS中的IP firewall主要由3个规则部分组成Mangle、Filter、NAT ， 而Address-list常用于地址列表分类。

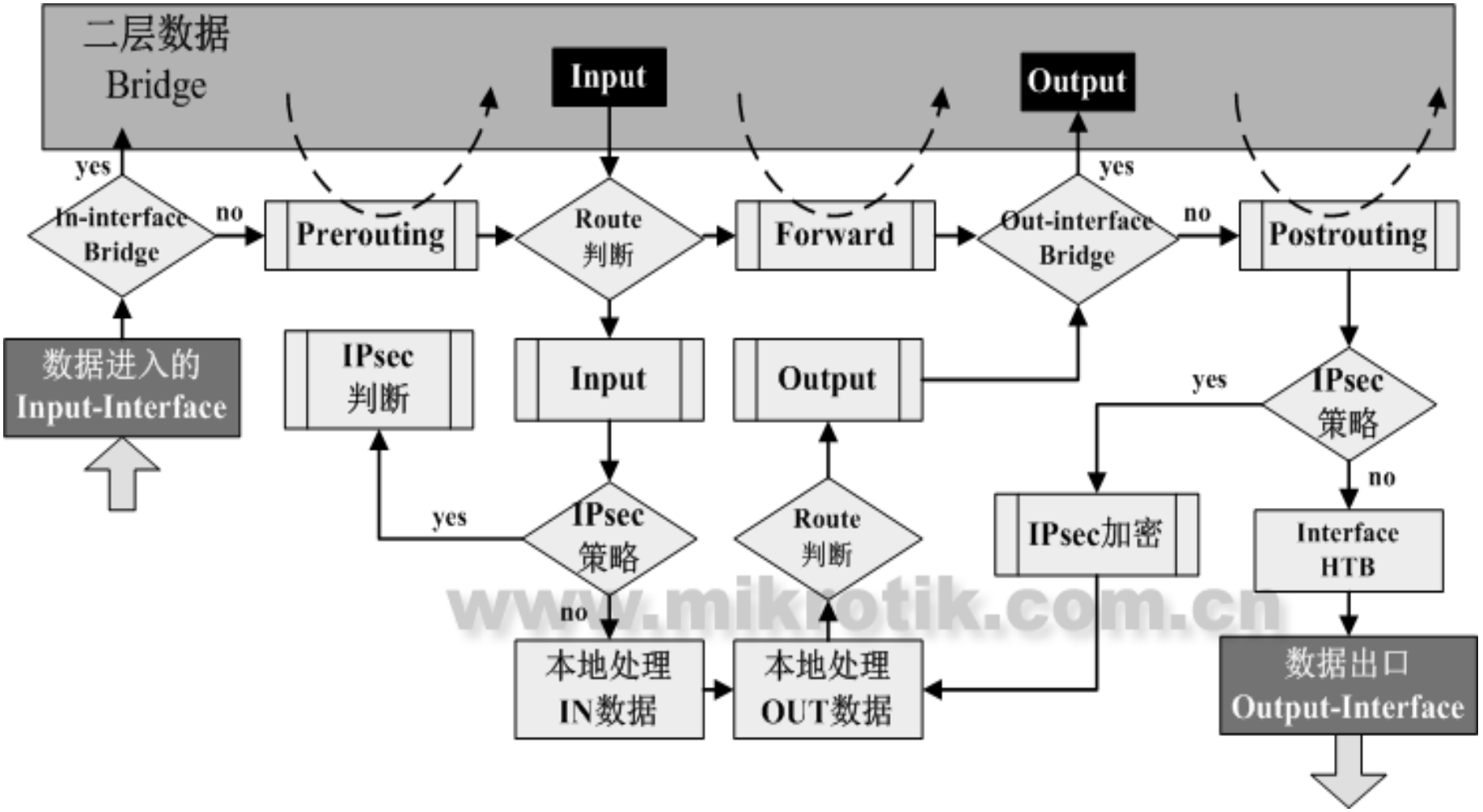
- Mangle通过标记特定的IP数据流后，为Filter、NAT和、路由、Queue提供标记后的IP数据流



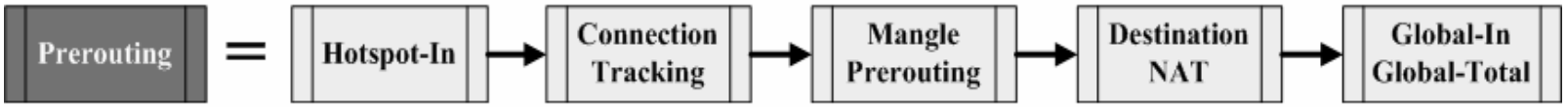
Mangle特点

- 不管是QoS、防火墙、nat规则和路由，在许多特殊的应用中都会使用Mangle标记（如routing-mark、connection-mark、packet-mark）
- Mangle在RouterOS中起到一个标记和分类的作用；
- 掌握RouterOS的高级应用，必须了解Mangle在RouterOS中的运用原理；
- 理解了Mangle，也就理解了路由、防火墙和QoS的应用

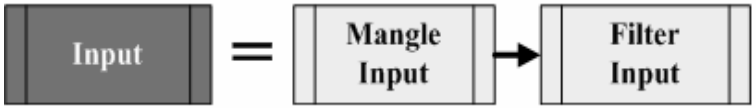
RouterOS IP数据处理流程



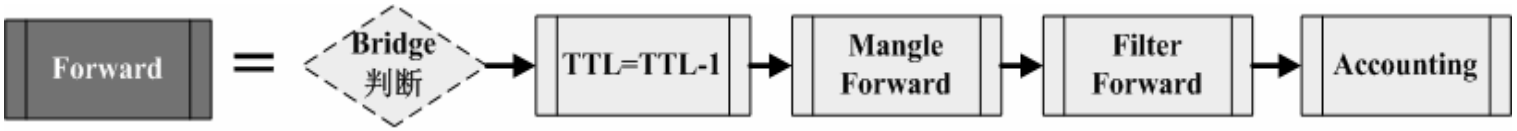
Prerouting – 路由之前



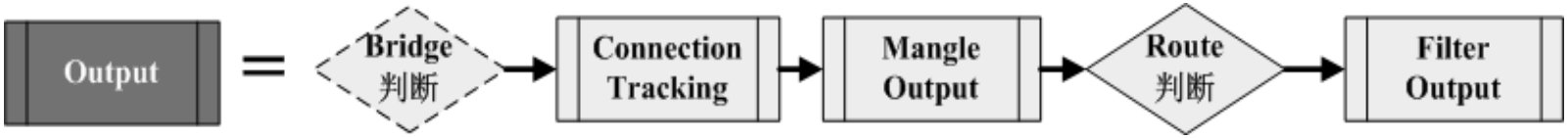
Input – 进入路由



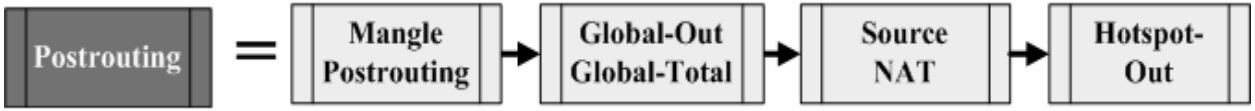
Forward – 转发路由



Output – 路由发出

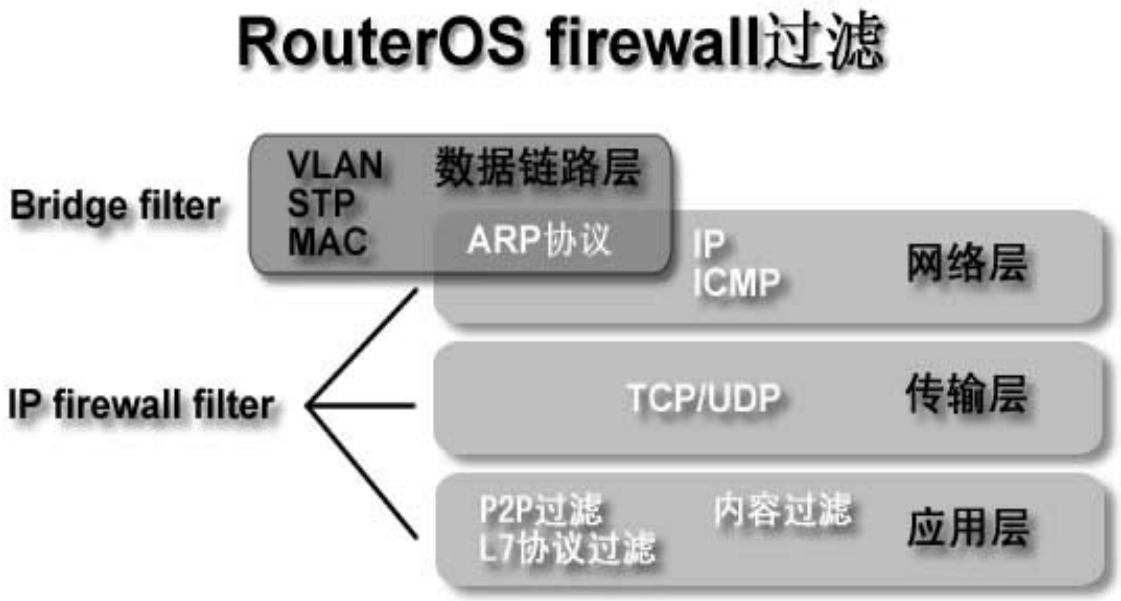


Postrouting – 路由之后



Firewall协议分类

分为二层过滤防火墙和三层与三层以上过滤防火墙，分别在bridge filter和ip firewall filter操作。



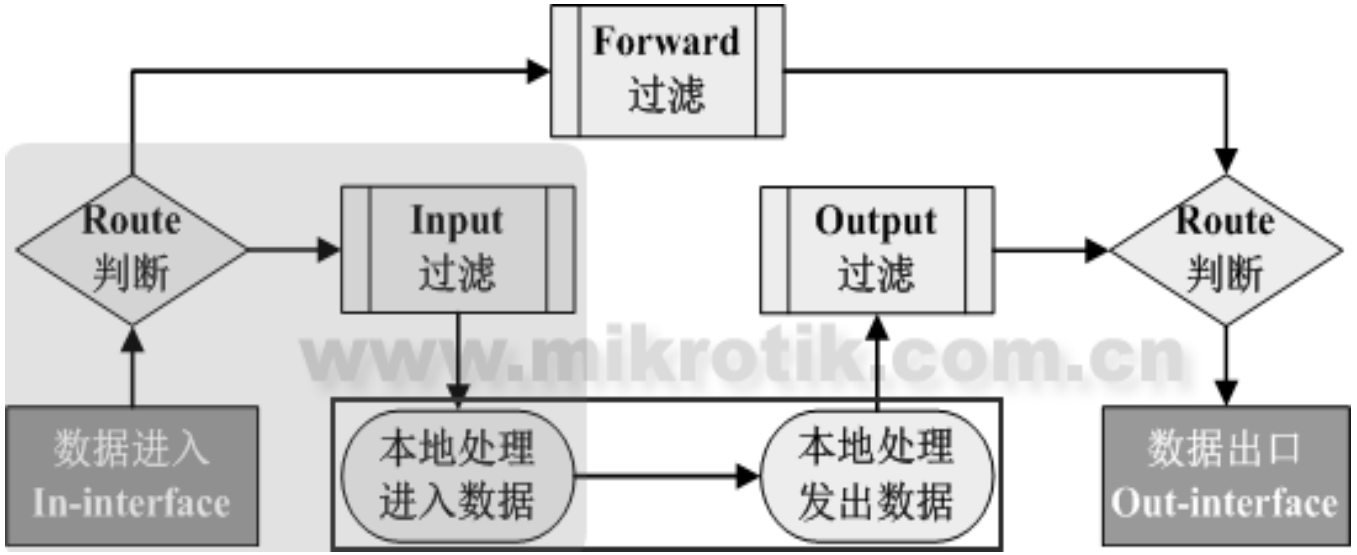
Firewall链表分类

RouterOS的链表组成主要包括：

- **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。**保护路由器**
- **forward –** 用于处理通过路由器的数据包。 **保护用户**
- **output –** 用于处理源于路由器并从其中一个接口出去的数据包。 **控制路由发出数据**
- 自定义链表 - RouterOS 中可以通过自定义建立其他链表，用于数据过滤分类。

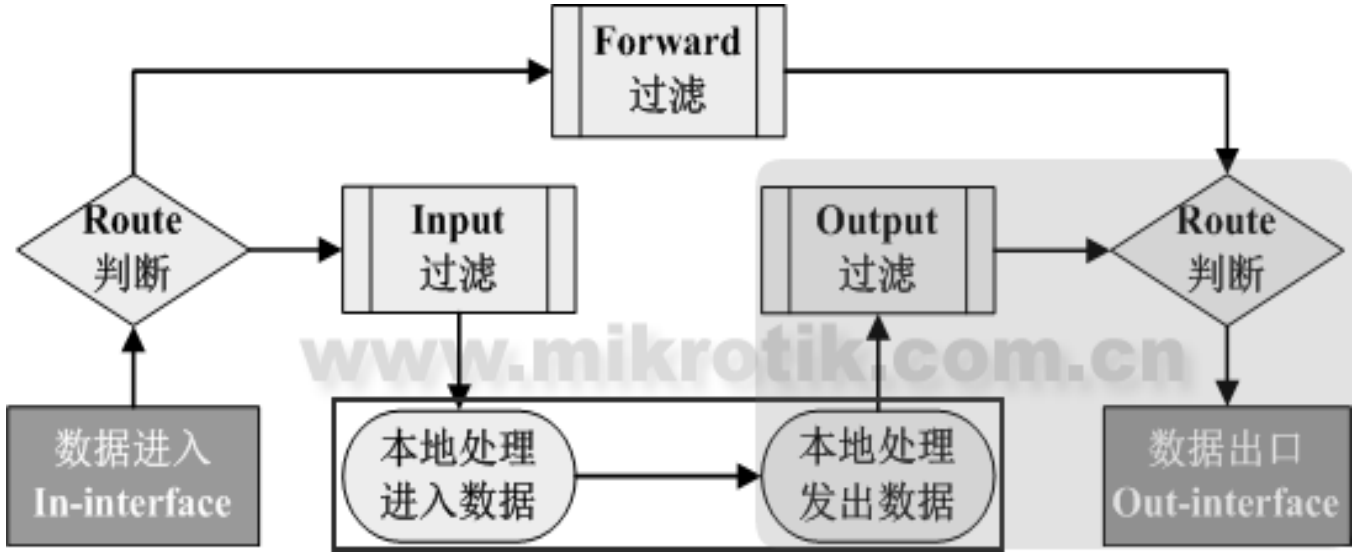
防火墙过滤 - **Input**

- Input的过滤流程



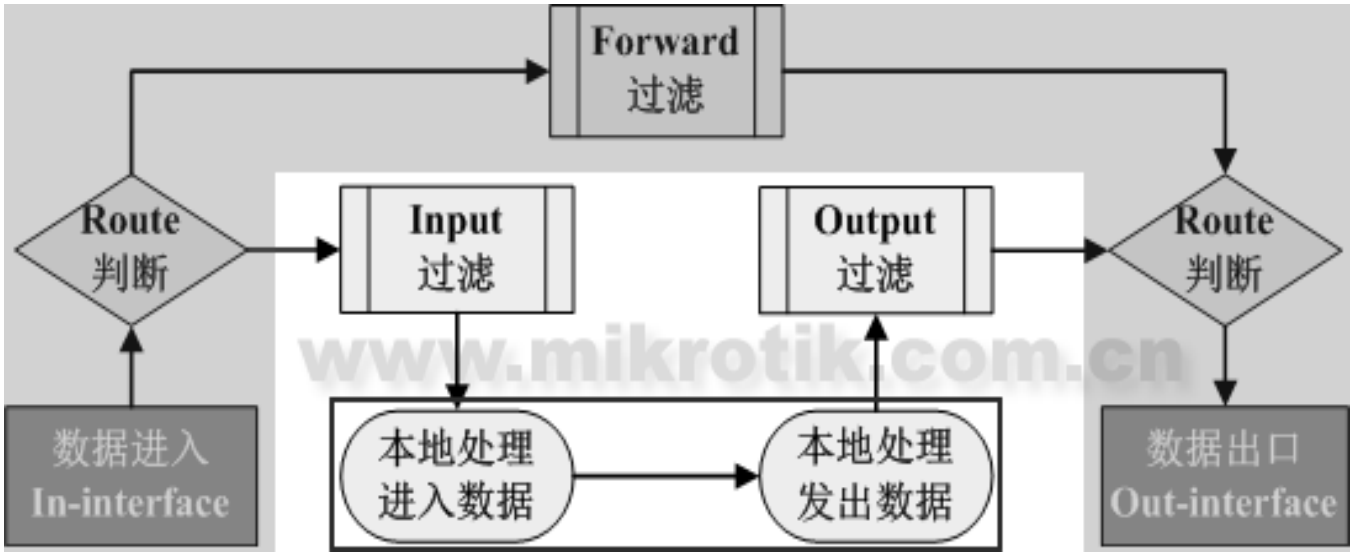
防火墙过滤 - **Output**

- Output的过滤流程



防火墙过滤 - **Forward**

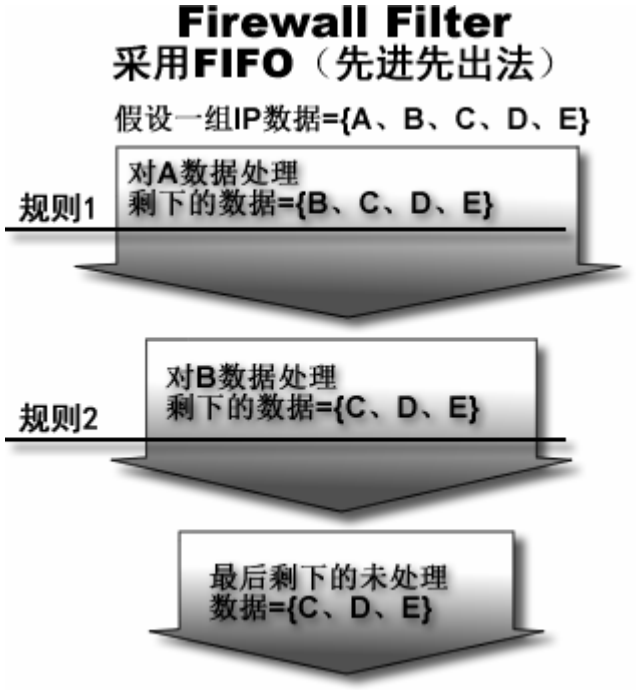
- Forward过滤流程



Filter规则原则

- firewall filter 被用于IP数据包过滤，即三层数据过滤。

- 防火墙规则构成是if – then 的方式（**if** 环境条件 **then** action执行）



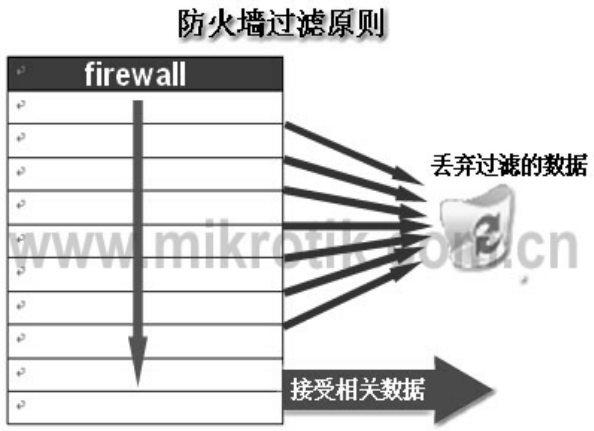
- 防火墙规则执行是从上而下，成为FIFO（First In First Out）方式

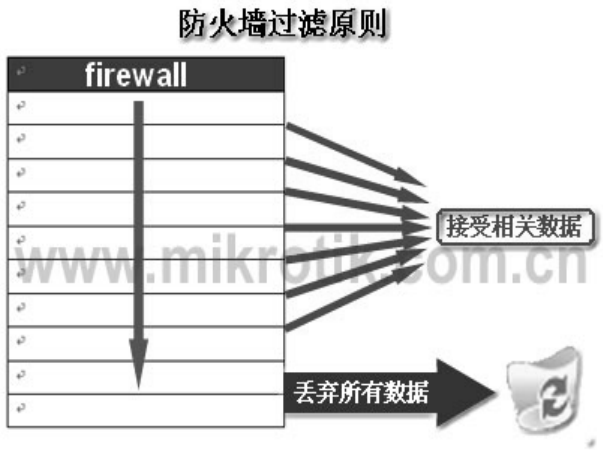
Filter过滤原则

过滤数据时我们可以通过以下的两种原则：

- 先丢弃后接受

- 先接受后丢弃



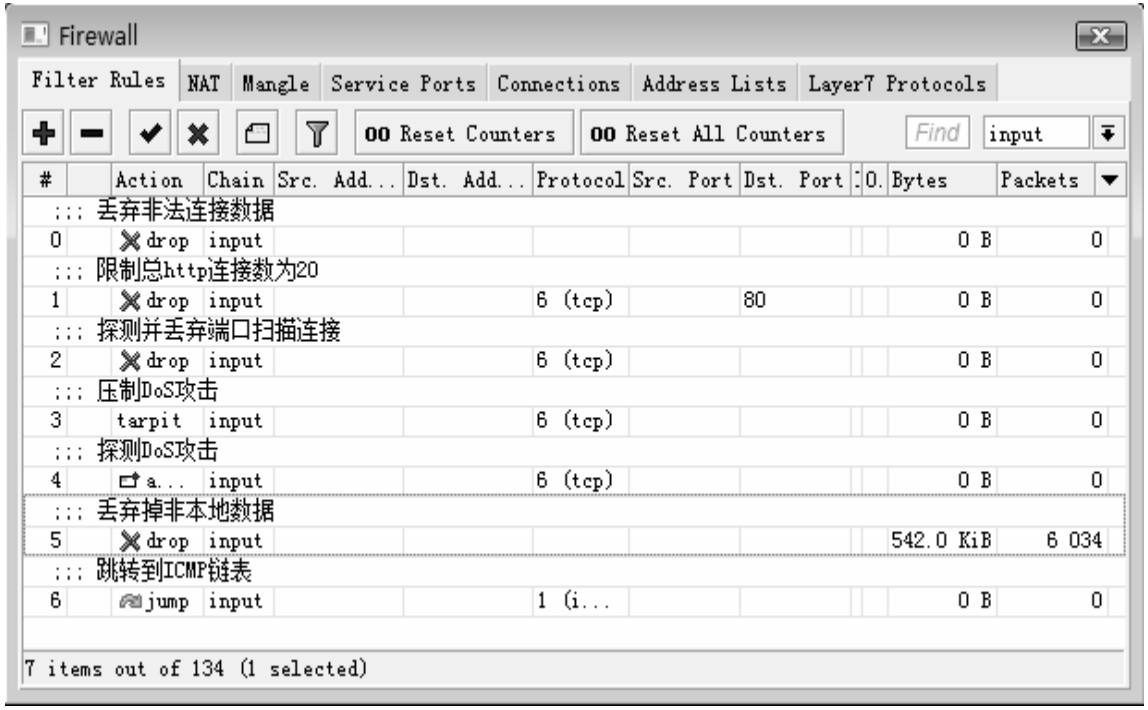


Filter action

- ***Accept –*** 接受数据包，没有任何的操作。例如接受数据包直接通过，不再以后的规则进行处理；
- ***Add-dst-to-address-list –*** 根据规则条件，将IP数据包的目标地址IP添加到指定address-list；
- ***Add-src-to-address-list –*** 根据规则条件， 将IP数据包的源地址IP添加到指定的address-list；
- ***Drop –*** 丢弃掉数据包（不会发送ICMP拒绝信息）；
- ***Jump –*** 跳转到指定的链表；
- ***Log –*** 与之匹配的操作将会被记录到system log中；
- ***Passthrough –*** 忽略该规则，继续向后执行下一条规则；
- ***Reject –*** 拒绝数据包，并发送ICMP拒绝信息；
- ***Return –*** 通过返回操作，返回到上一跳转链表；
- ***Tarpit –*** 捕捉并控制进入的TCP连接。

input 链表

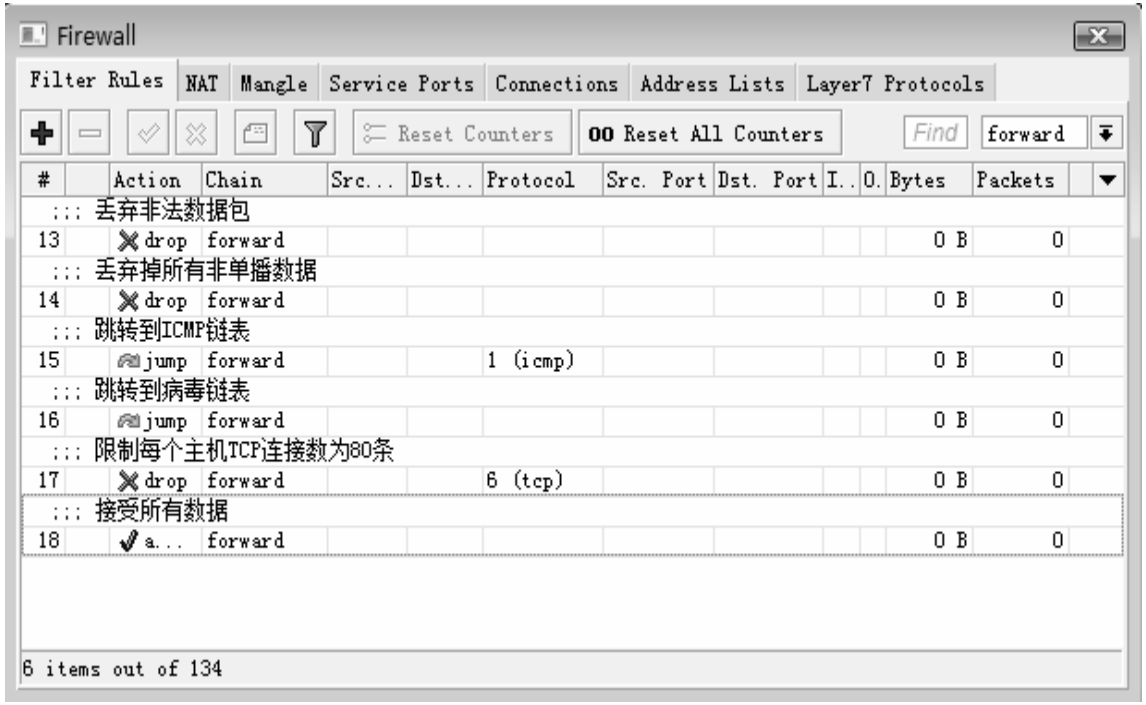
- 现在我来看事例中的防火墙规则，我先从input链表开始，这里是对所有访问路由的数据进行过滤和处理：



从**input**链表中可以看到，我们对进入路由器的数据采用先拒绝非法的数据和连接，并将**ICMP**数据跳转到自定义的**ICMP**的链表中过滤。

forward链表

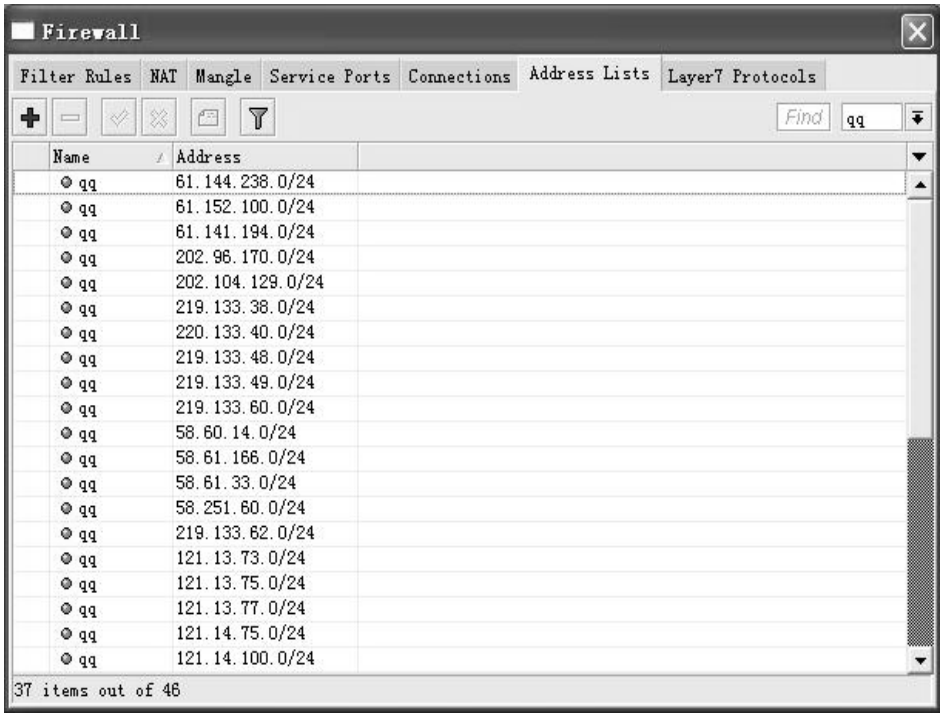
- 下面是forward链表一个应用防火墙事例：



forward链表，对非法数据包、非单播数据、 **ICMP**协议和常见的病毒等进行过滤，控制**TCP**连接数 。

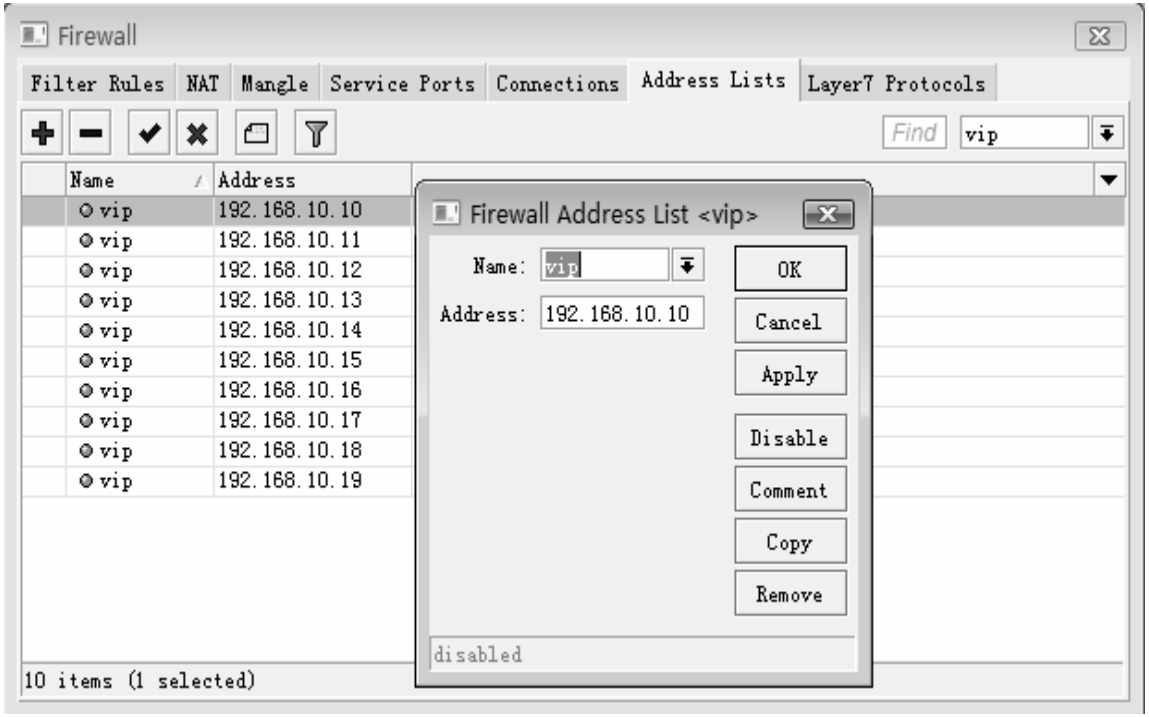
Firewall address-list

- 能通过address-list定义IP地址分组
- address-list能被多种应用调用和执行，例如过滤、流控和路由
- address-list能被mangle或者firewall filter动态定义和调用



定义地址列表 1

- 我们可以定义10个经理的IP地址到address-list中，并取名为VIP，
- 这样可以提供给filter、mangle和NAT使用



动态定义地址列表

- Address-list可以通过ip firewall fiter/nat/mangle建立动态定义地址列表

- 如记录所有访问udp/8000端口的客户IP动态添加到access8000的address-list中，然后等待其他规则调用并处理

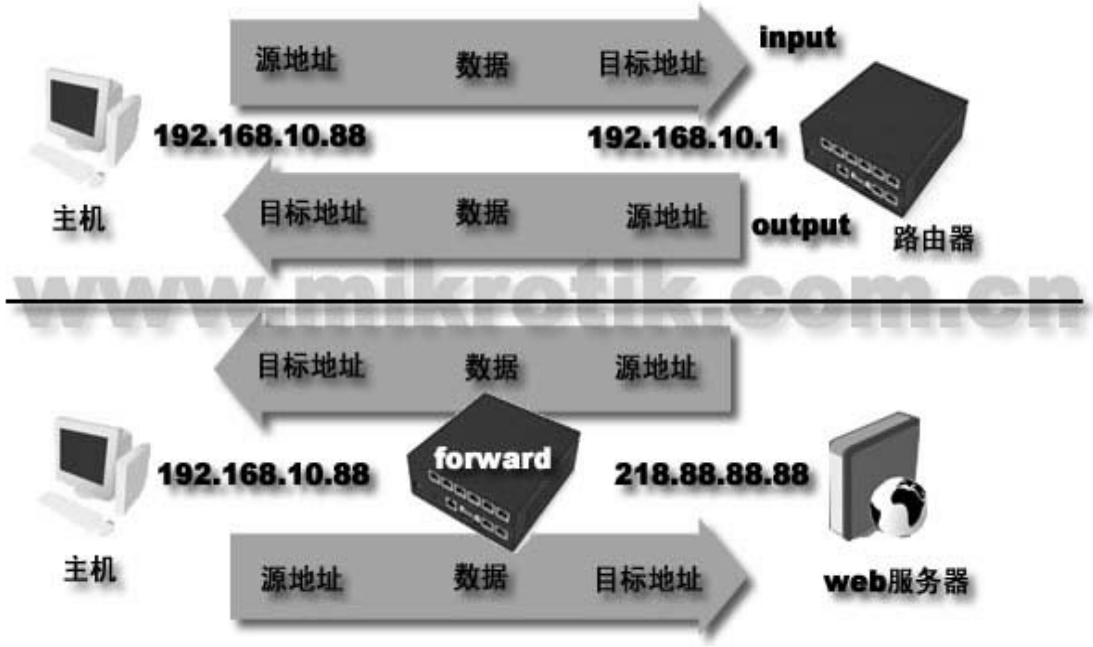
- 也可以定义超过连接数超过100的用户定义到blacklist黑名单

Firewall操作实例

- 允许VIP用户192.168.10.10-192.168.10.19能连接QQ，禁止其他地址访问QQ
- 禁止192.168.10.129-192.168.10.254访问外网
- 禁止访问访问www.163.com的网站，但10个VIP可以正常访问

IP源地址和目标地址概念 1

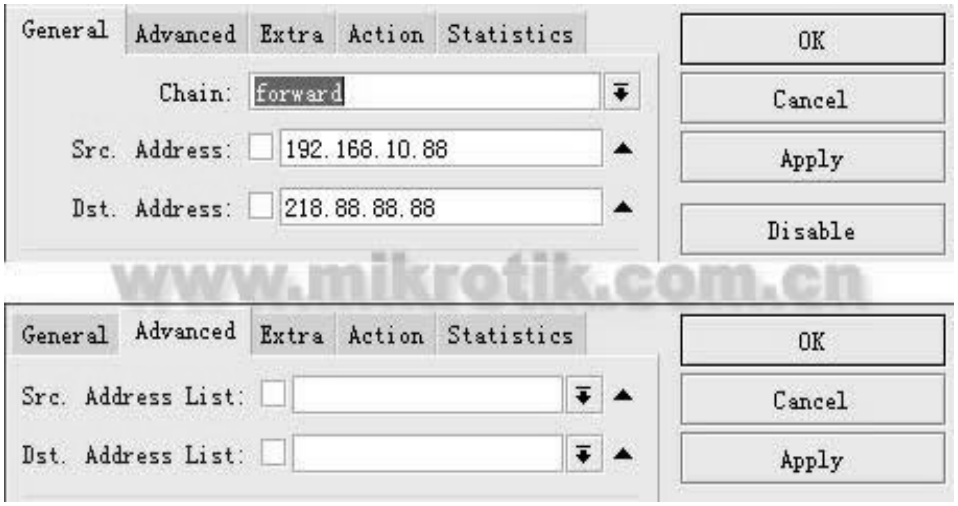
- 如何判断源地址和目标地址，与他们在ip firewall filter的链表，如下面的图：



从该图上可以看到，内网主机**192.168.10.88**与路由器**192.168.10.1**通信，内网主机**192.168.10.88**向路由和外网的**web**服务器通信。
不同情况下源目标**IP**地址的转变和使用的**chain**链表情况。

IP源地址和目标地址概念 2

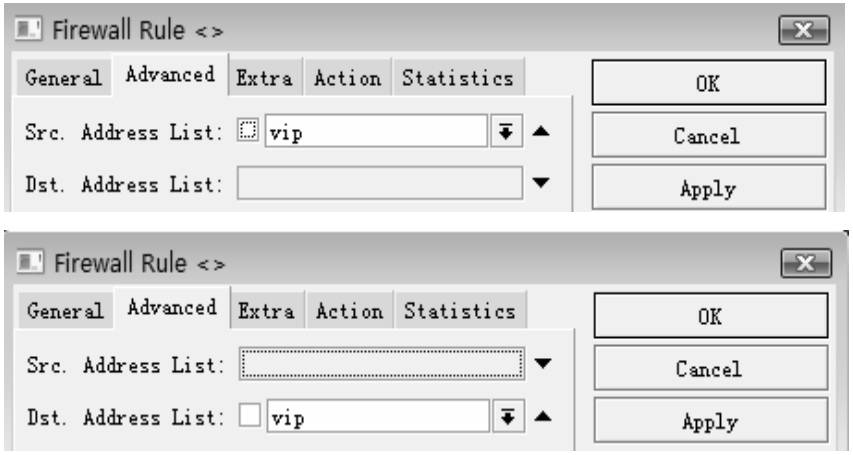
- 在这里要记住任何通信是双向的，而不仅只有源到目标一条链路。
- 在RouterOS中两个选择涉及到源和目标地址，General标签中的src-address、dst-address和Advanced的src-address-list、dst-address-list如下图



网络访问控制 1

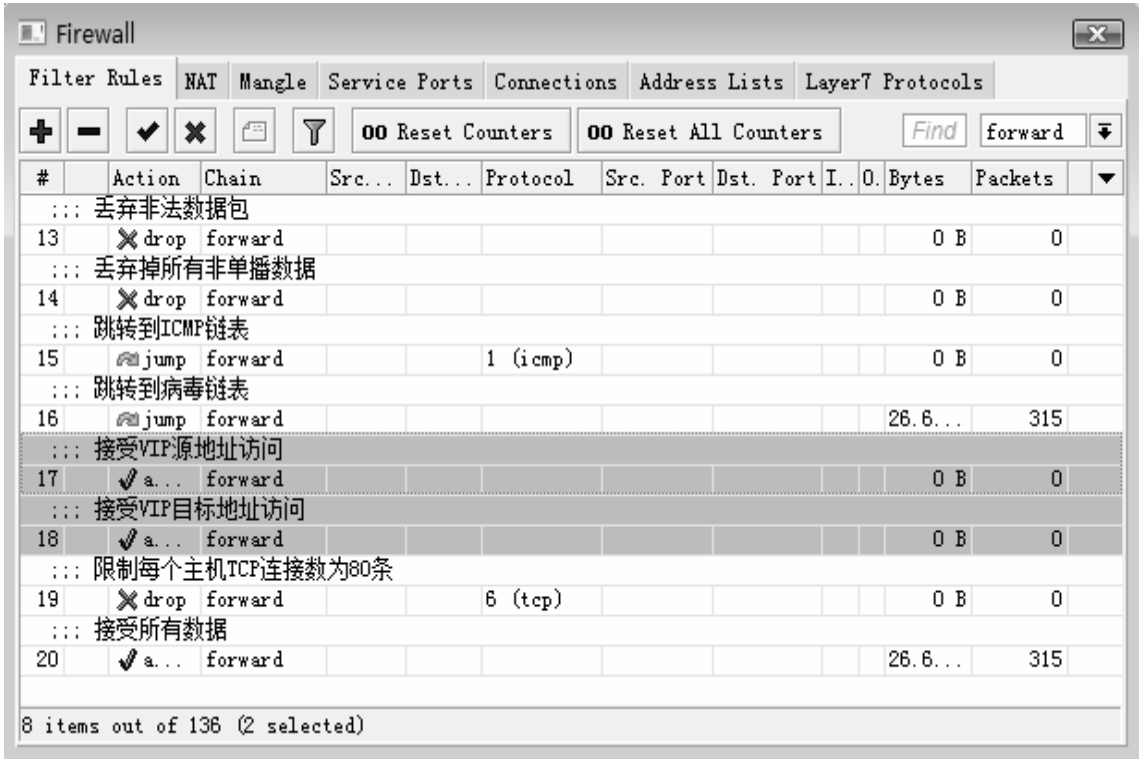
- 通过address-list列表，将10个经理的IP地址设置规则为直接通过，不受限制的访问网络

- 在ip firewall filter的forward中我们需要建立2条规则，一条是接受源地址的访问，一条是接受目标地址的访问



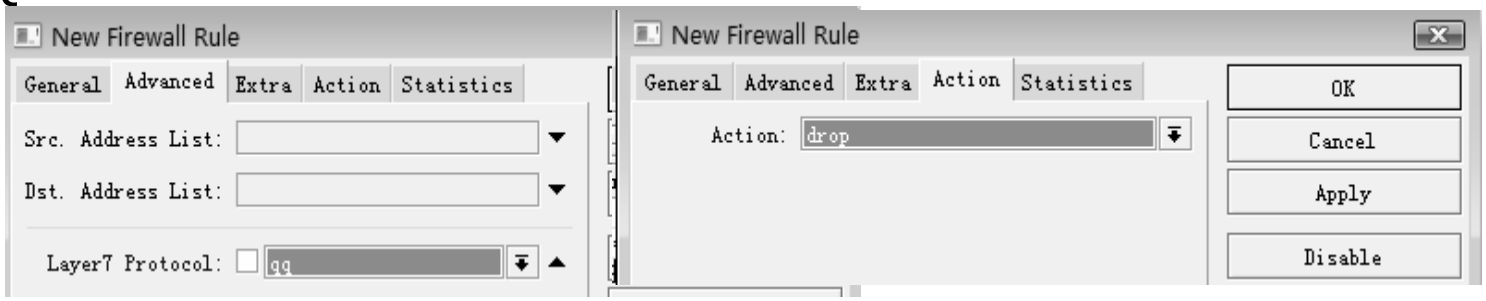
网络访问控制 2

- 添加后完两条规则后，需要将规则移动到在病毒过滤的下方，TCP连接数限制的上方



网络访问控制 3

- 禁止所有人用QQ聊天，在forward链表中调用7层过滤，禁止连接QQ

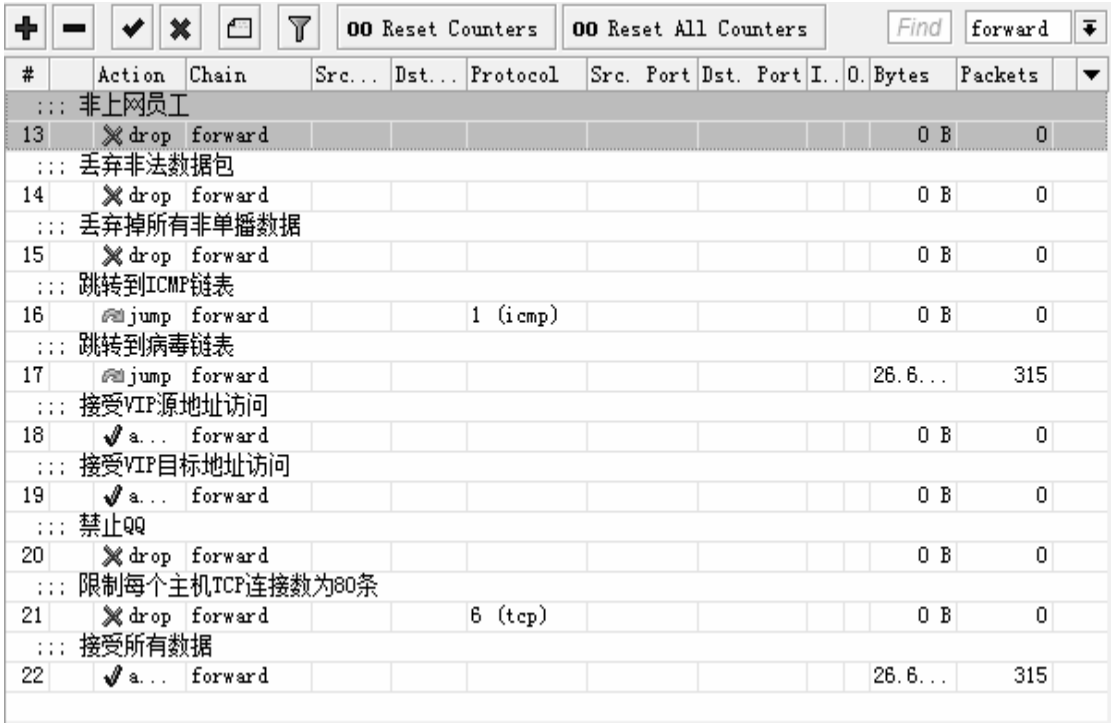


00 Reset Counters 00 Reset All Counters Find forward

#		Action	Chain	Src...	Dst...	Protocol	Src. Port	Dst. Port	I..	O.	Bytes	Packets
;;; 丢弃非法数据包												
13		drop	forward								0 B	0
;;; 丢弃掉所有非单播数据												
14		drop	forward								0 B	0
;;; 跳转到ICMP链表												
15		jump	forward			1 (icmp)					0 B	0
;;; 跳转到病毒链表												
16		jump	forward								26.6...	315
;;; 接受VIP源地址访问												
17		a...	forward								0 B	0
;;; 接受VIP目标地址访问												
18		a...	forward								0 B	0
;;; 禁止QQ												
19		drop	forward								0 B	0
;;; 限制每个主机TCP连接数为80条												
20		drop	forward			6 (tcp)					0 B	0
;;; 接受所有数据												
21		a...	forward								26.6...	315

网络访问控制 4

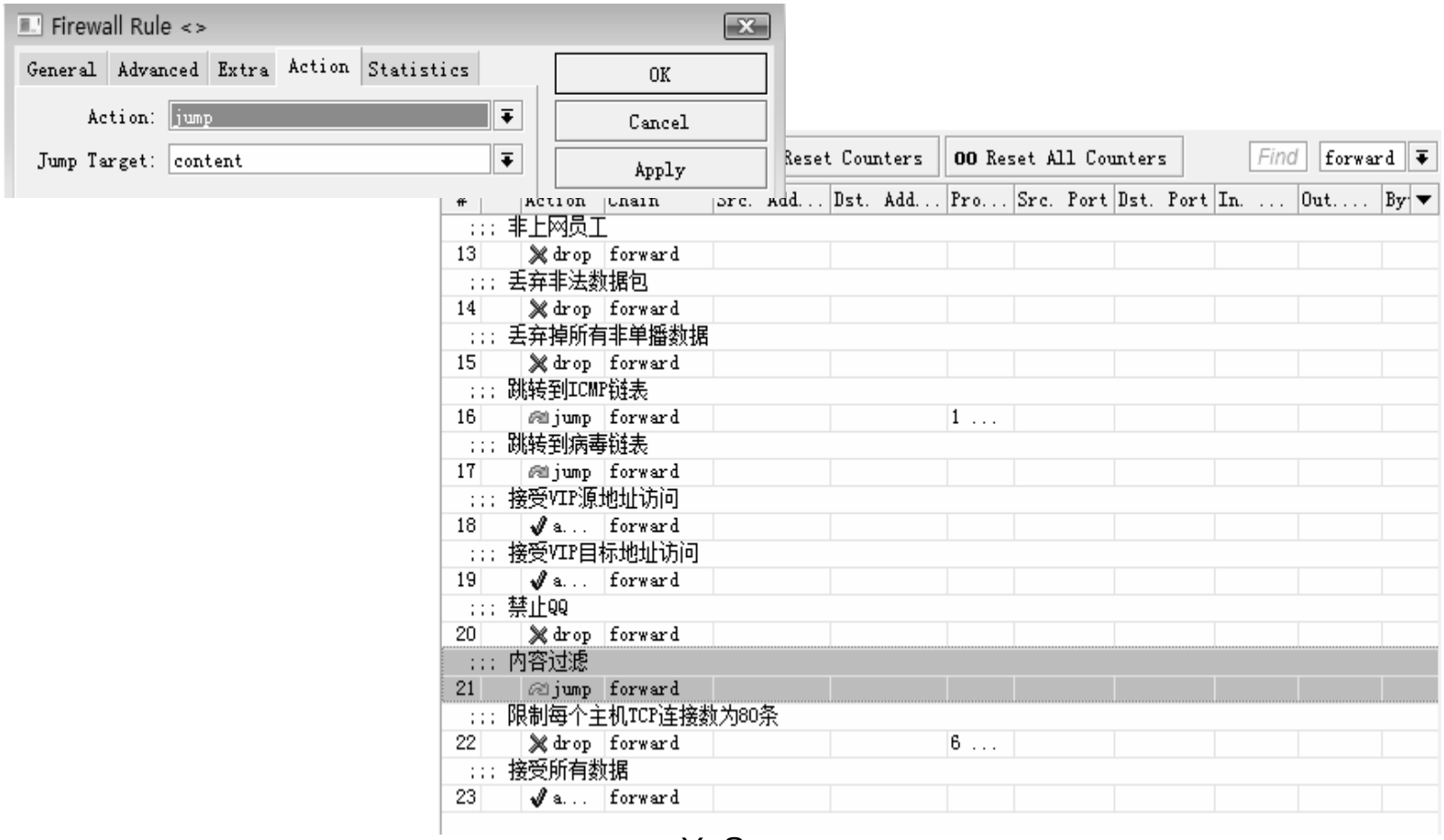
- 禁止非上网员工连接网络，通过intercept列表控制
- 在/ip firewall filter add chain=foreward src-address-list=intercept action=drop
- 并将规则移动到最上



#	Action	Chain	Src...	Dst...	Protocol	Src. Port	Dst. Port	I..	O.	Bytes	Packets
::: 非上网员工											
13	drop	forward								0 B	0
::: 丢弃非法数据包											
14	drop	forward								0 B	0
::: 丢弃掉所有非单播数据											
15	drop	forward								0 B	0
::: 跳转到ICMP链表											
16	jump	forward			1 (icmp)					0 B	0
::: 跳转到病毒链表											
17	jump	forward								26.6...	315
::: 接受VIP源地址访问											
18	a...	forward								0 B	0
::: 接受VIP目标地址访问											
19	a...	forward								0 B	0
::: 禁止QQ											
20	drop	forward								0 B	0
::: 限制每个主机TCP连接数为80条											
21	drop	forward			6 (tcp)					0 B	0
::: 接受所有数据											
22	a...	forward								26.6...	315

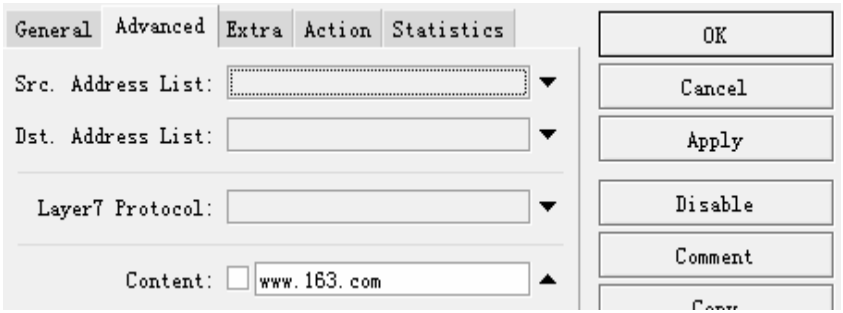
网络访问控制 5

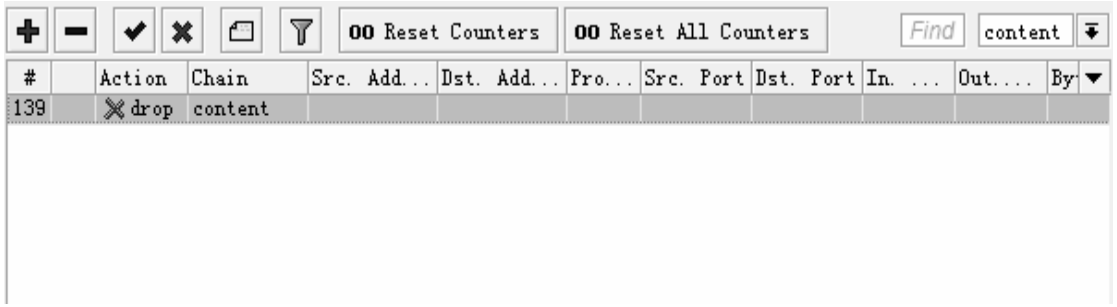
- 新建一个content链表，对内容进行过滤，针对一些网页传输的字符，如是web中的内容和网站名称



网络访问控制 6

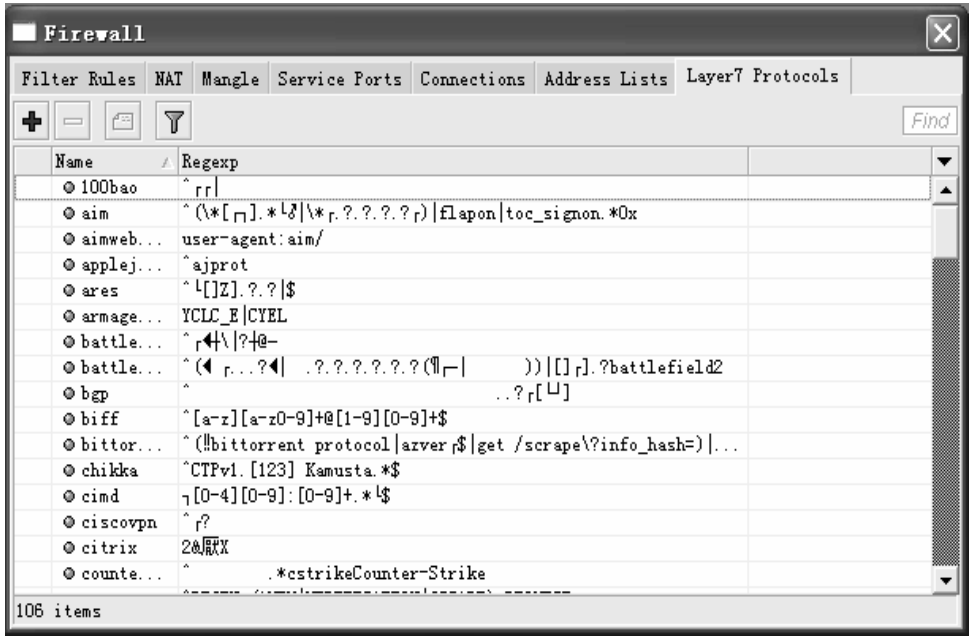
- 在content链表中，新建一个action=drop 指定的www.163.com的内容过滤
- 可以在content添加更多的内容过滤

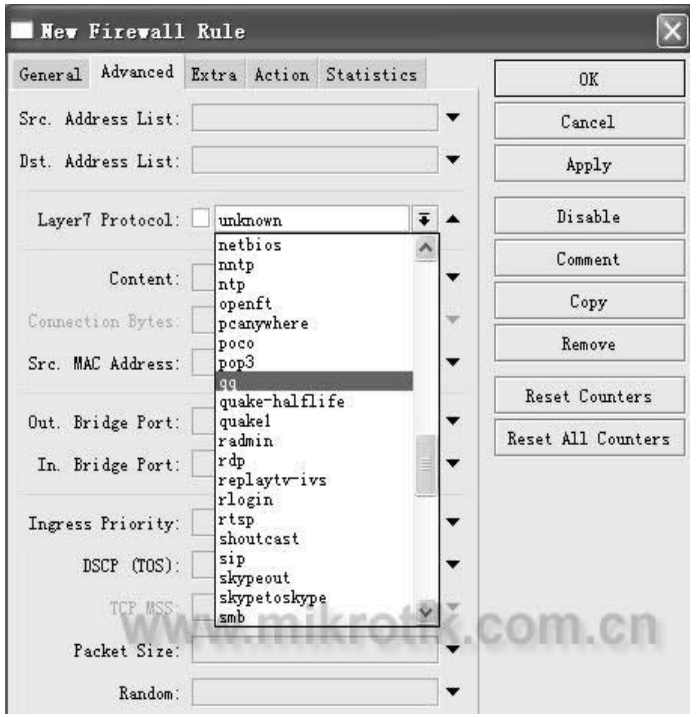




Layer 7协议

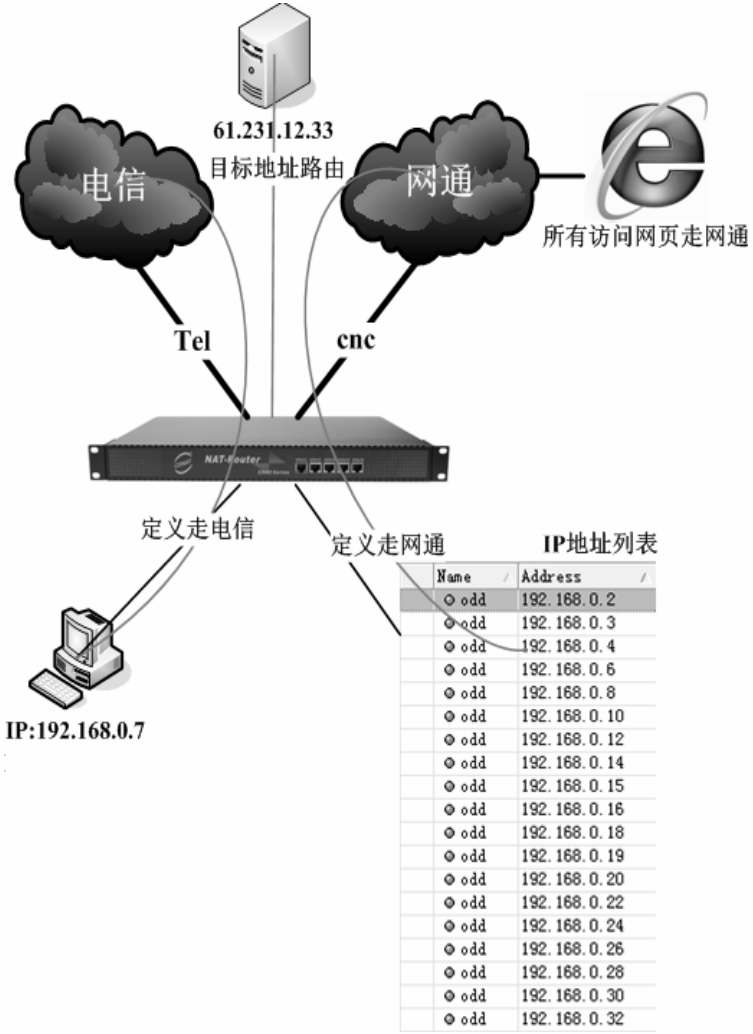
- 在RouterOS3.0中增加了Layer7协议过滤功能，即对应用程序的代码进行过滤，这些代码我们通过Regexp的脚本进行编辑，也可以通过我们预先编辑好的RouterOS脚本导入Layer7协议应用列表





RouterOS 路由策略

RouterOS的策略路由



- 支持源IP地址的策略路由
- 支持目标IP地址策略路由
- 支持网页等端口的策略路由
- 支持IP地址列表的策略路由
- 负载均衡Nth与PCC
- 各种策略都可以组合使用

路由表的关系 1

RouterOS能维护多个独立的路由表，能灵活的分配策略路由规则；

通过下面的操作命令可以标记路由与定义路由策略表

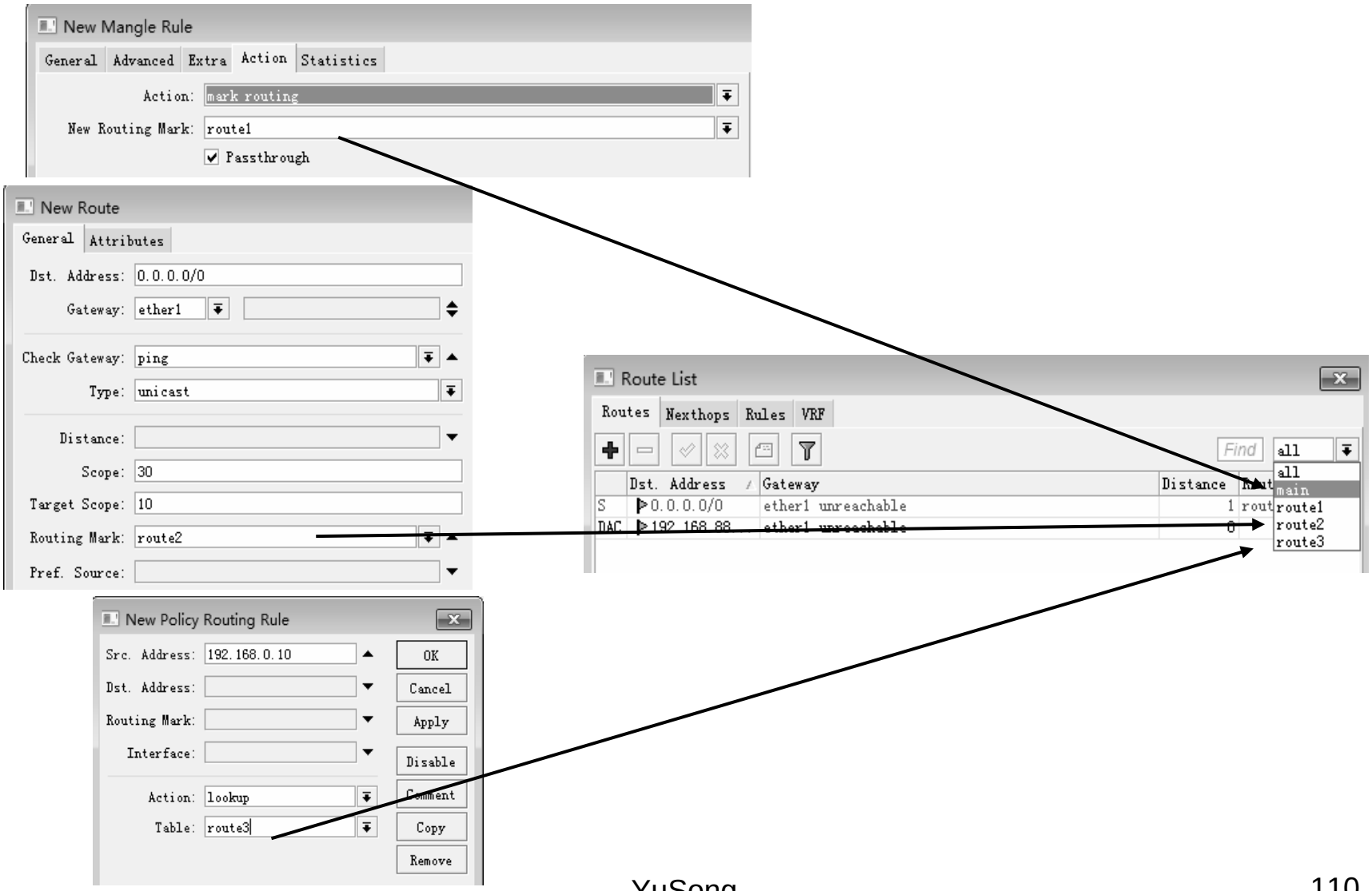
- /ip firewall mangle mark-routing （支持源目标和端口路由）
- /ip route routing-mark （支持源目标路由）
- /ip route rule table （支持源目标路由）

他们之间关系是平等的：

mark-routing = routing-mark = table

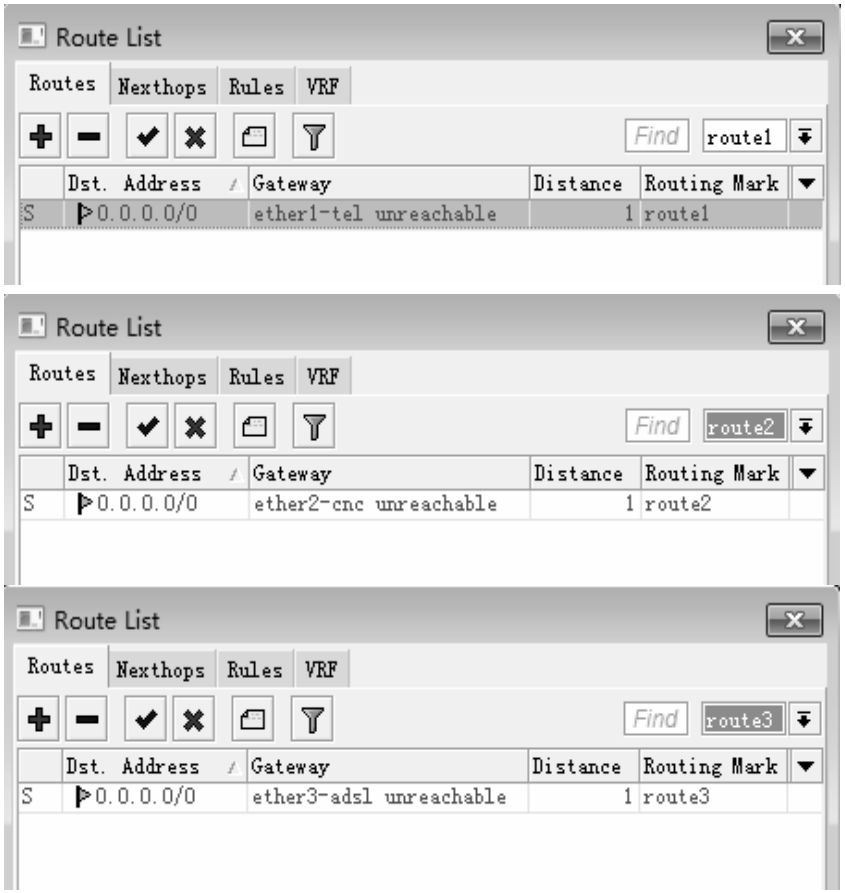
当他们被定义后，都会在**ip route**中新建路由表，

路由表的关系 2


New Mangle Rule
General
Advanced
Extra
Action
Statistics
Action: mark routing
New Routing Mark: route1
Passthrough
New Route
General
Attributes
Dst. Address: 0.0.0.0/0
Gateway: ether1
Check Gateway: ping
Type: unicast
Distance:
Scope: 30
Target Scope: 10
Routing Mark: route2
Pref. Source:
Route List
Routes
Nexthops
Rules
VRF
Find
all
Dst. Address
Gateway
Distance
S 0.0.0.0/0 ether1 unreachable 1
all
main
route1
route2
route3
New Policy Routing Rule
Src. Address: 192.168.0.10
Dst. Address:
Routing Mark:
Interface:
Action: lookup
Table: route3
OK
Cancel
Apply
Disable
Comment
Copy
Remove

路由表的关系 3

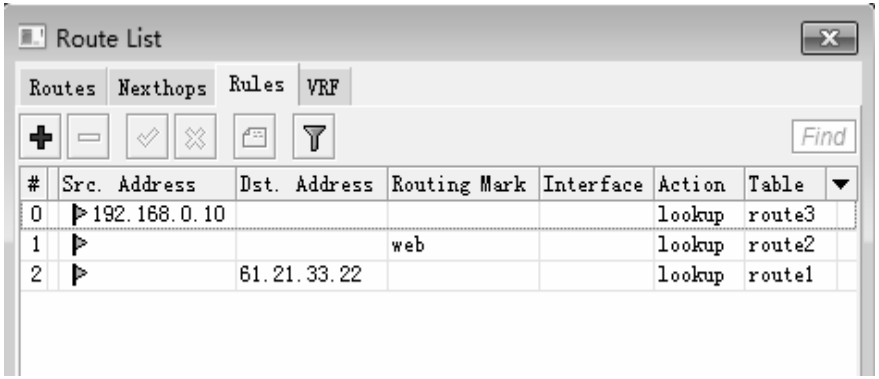
- 在ip route中定义各个列表的网关出口



路由表的关系

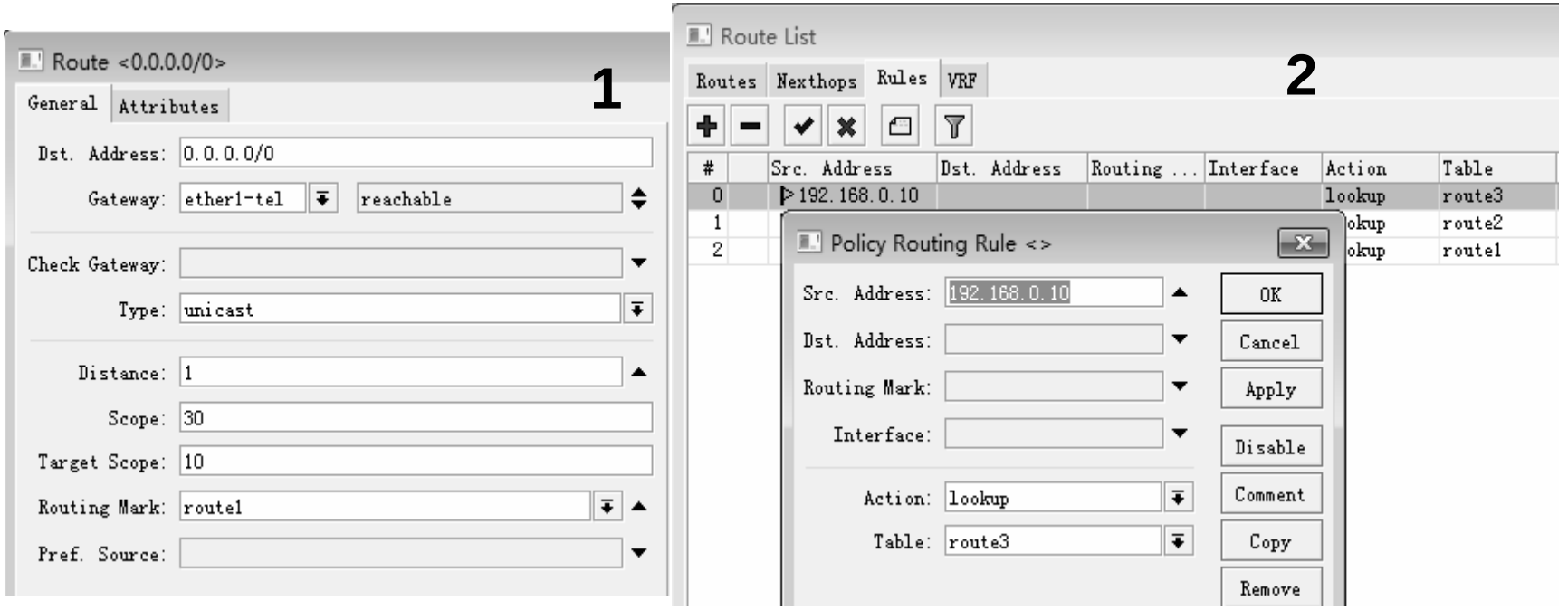
- ip route rules是对源和目标IP地址，以及端口设置到指定的路由表

- 即rules规则类似一个分配器，将不同类型的路由标记和规则指定到相应的路由表中



源IP地址的策略路由

- 我们内网一台主机IP为192.168.0.10需要指定到电信线路
- 首先在ip route 定义route1的路由表网关
- 然后在ip route rule定义src-address的源IP地址，并定义到route1的路由表

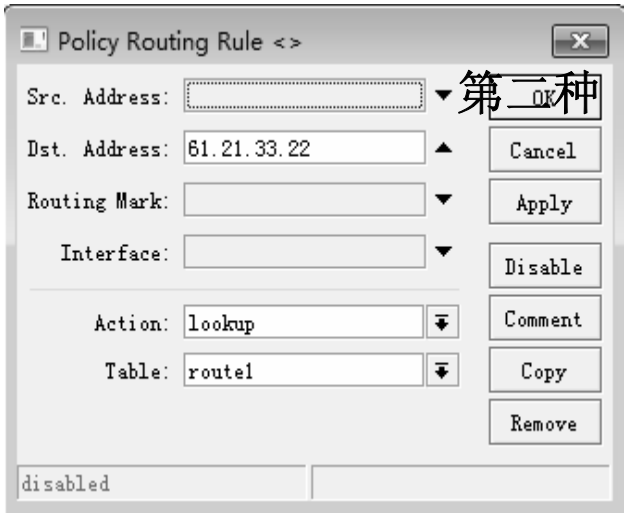


目标IP地址路由

- 目标IP路由有三种方式，但这里我们主要讲其中两种，电信网通的双线路由表就属于目标IP地址路由

- 第一种：在ip route定义dst-address，添加后直接生效
- 第二种：在ip route rule定义dst-address，添加后需要知道路由表网关

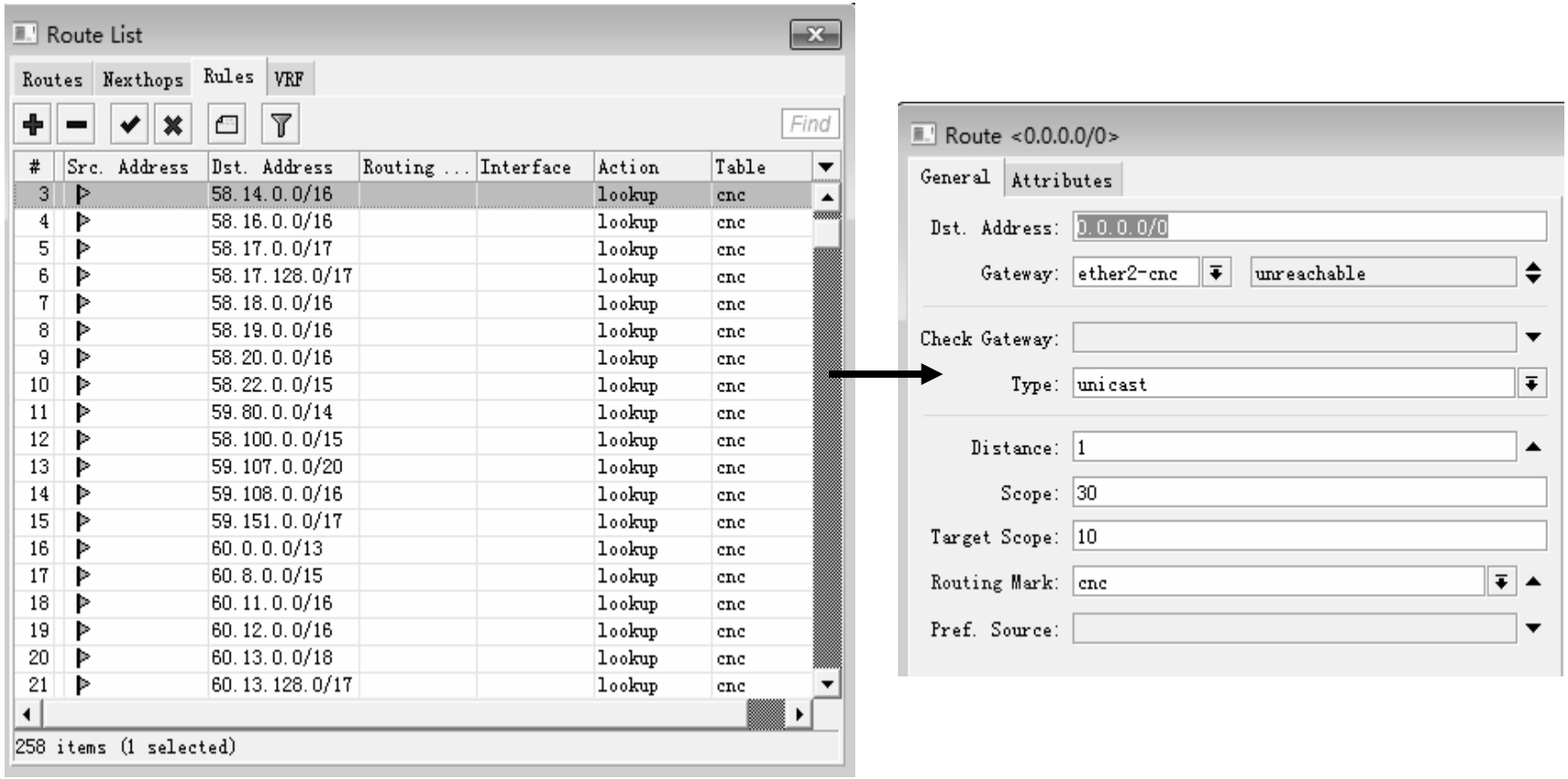
第一种



第二种

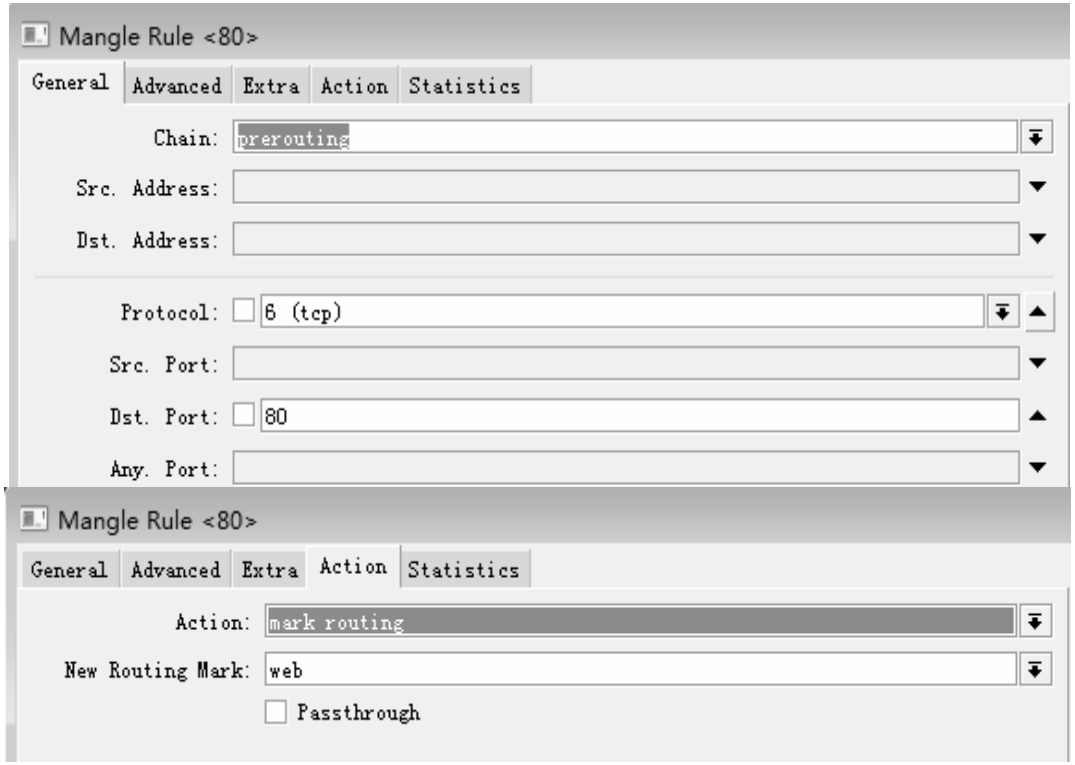
目标IP地址路由

- 通过导入cnc的网通路由表到ip route rule中后，在ip route添加给路由表的网关和路由标记



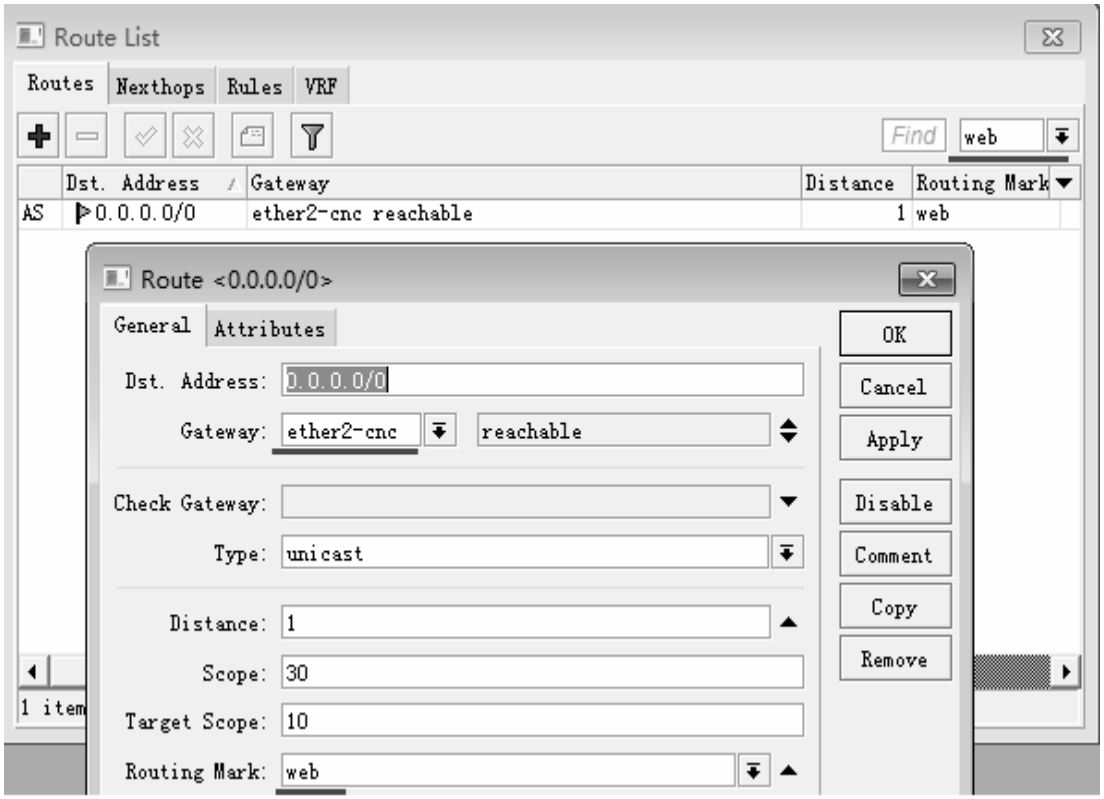
端口策略路由

- 端口策略路由需要使用mangle的标记路由。
- 这里我们定义网页的访问路由，协议为tcp，目标端口为80，执行mark routing，并标记新的路由连接取名web。



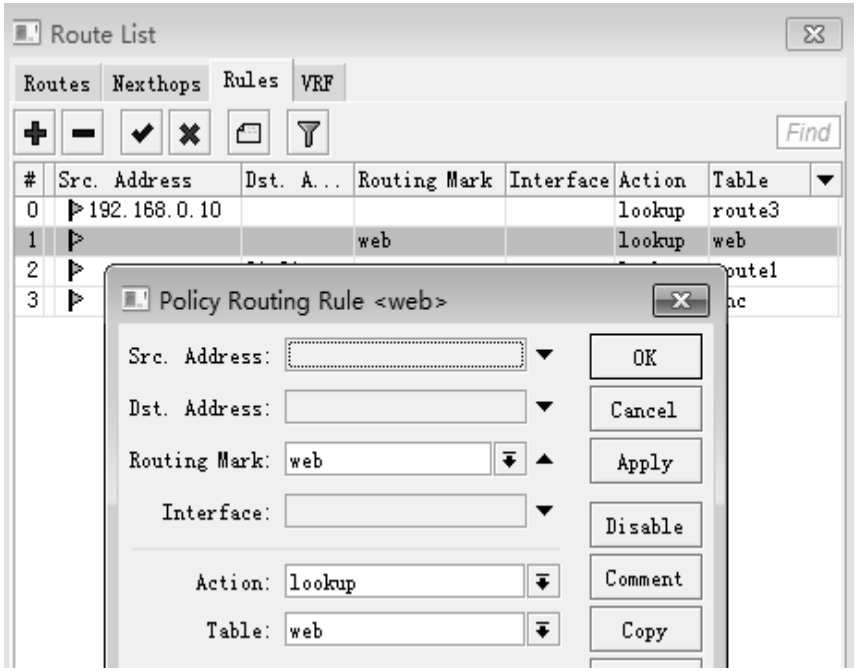
端口策略路由 2

- 在mangle中标记好端口后，在ip route设置web标记的网关，如果是普通双线只做网页80端口的策略规则，配置后可执行生效。



端口策略路由 2

- 如果当多线路，在ip route rule配置了各种规则，我们就需要在ip route rule里再次定义web的路由表。



地址列表的路由 1

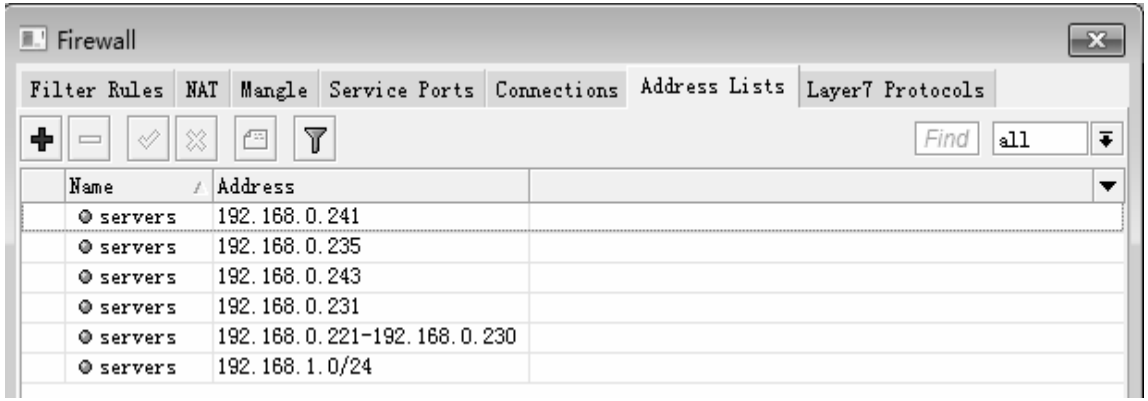
- 有一些特定的主机需要走指定的线路出去，但ip route rule每条规则只能指定IP地址段，无法同时设定非连续的地址，如服务器地址可能存在非连续或者局域网的VIP用户等。

- 或者我们需要对奇数和偶数主机地址的路由策略，在双线策略中我们可以将内网主机分类奇数和偶数组，分别走不通的线路，这样分流更佳合理。

- 通过ip firewall address-list定义地址列表，并在mangle中标记路由数据。

地址列表的路由 2

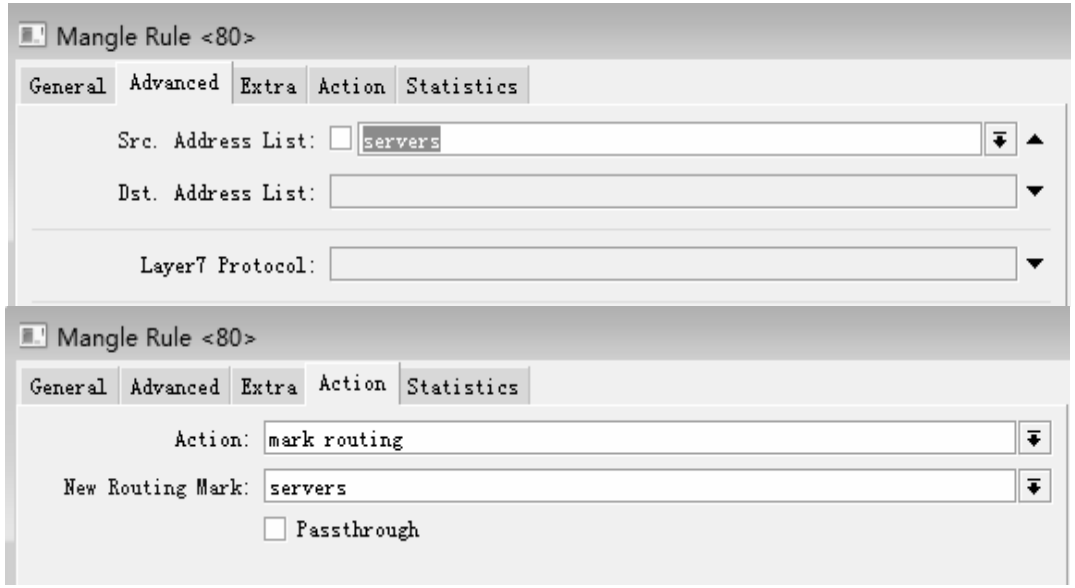
- 我们定义服务器的IP地址列表走指定的路由
- 在ip firewall address-list 添加名称为“servers”
- 定义地址列表我可以填写固定IP，IP地址范围和IP地址段，如下：



地址列表的路由 3

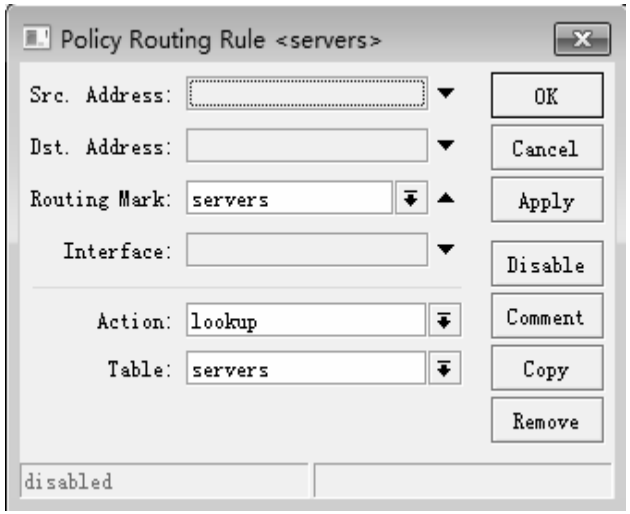
- 在ip firewall mangle标记servers的地址列表路由

- 因为是内网地址我们选择src-address-list，并设置servers的地址列表，并定义new routing mark 为servers



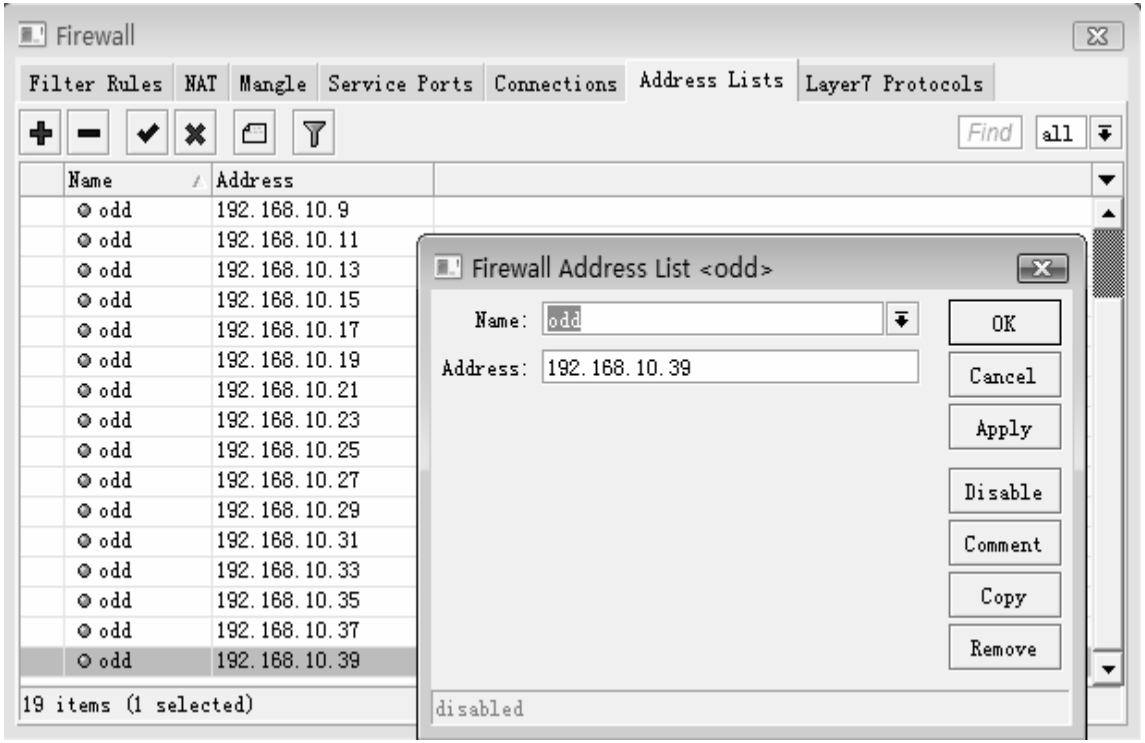
地址列表的路由 4

- 进入ip route为servers路由标记建立路由表，指定网关为ether1-tel
- 为确保规则运行正常，在ip route rule中添加规则



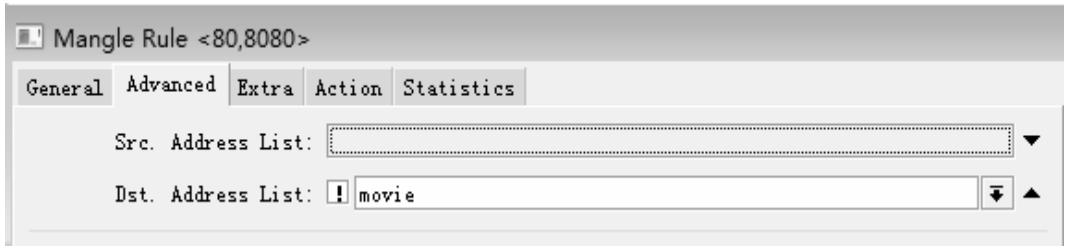
地址列表的路由 4

- 奇偶的路由策略，我们需要在ip address-list定义奇数或者偶数IP地址，如下面的列表为odd（奇数IP地址）
- 之后的操作和servers服务器的标记相同，这样的地址列表路由也属于源地址策略路由，相反也可以应用到目标地址策略路由



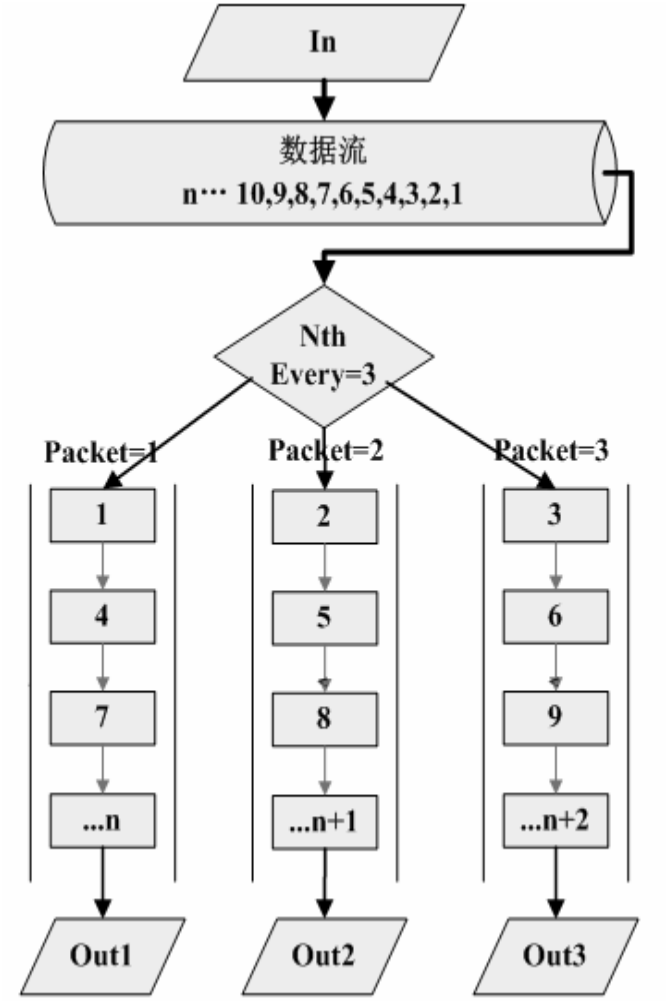
策略组合应用

- 通过Mangle的标记我们可以将地址列表和端口的策略路由组合使用。
- 如网页的80端口，我们可以添加奇数源地址地址列表，仅让奇数IP主机走指定的网关出口。
- 或者排除到某些目标网站不走指定的网关出口。
- 通过在规则参数前点“！”，即非的意思，就可以排除该地址列表范围。



RouterOS的负载均衡

RouterOS Nth

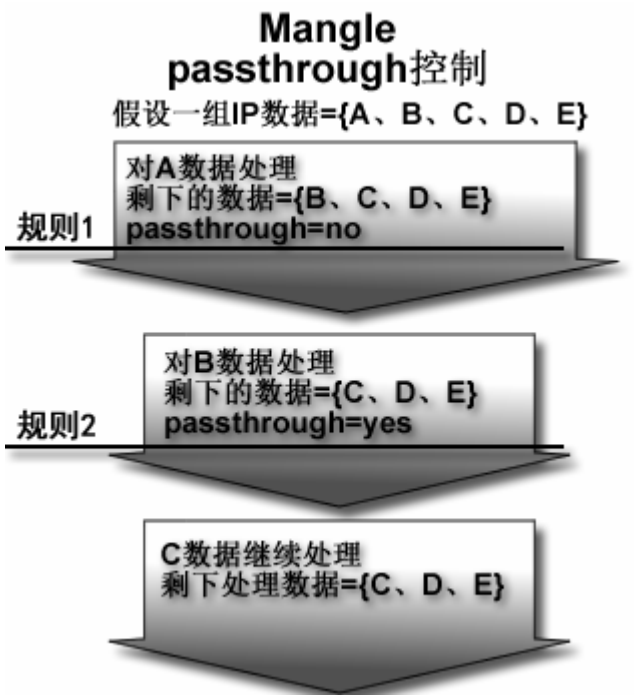


- 匹配特定的第N次收到的数据包的规则。对数据包重新标记和排列。
- Nth的一个计数器最多可以计数16个数据包
- **Every** – 匹配每every数据包，同时指定**Counter**（计数器值）
- **Packet** – 匹配给定的数据数，例如，Nth=3,1，匹配3个数据包的第1个

Passthrough对Nth的控制

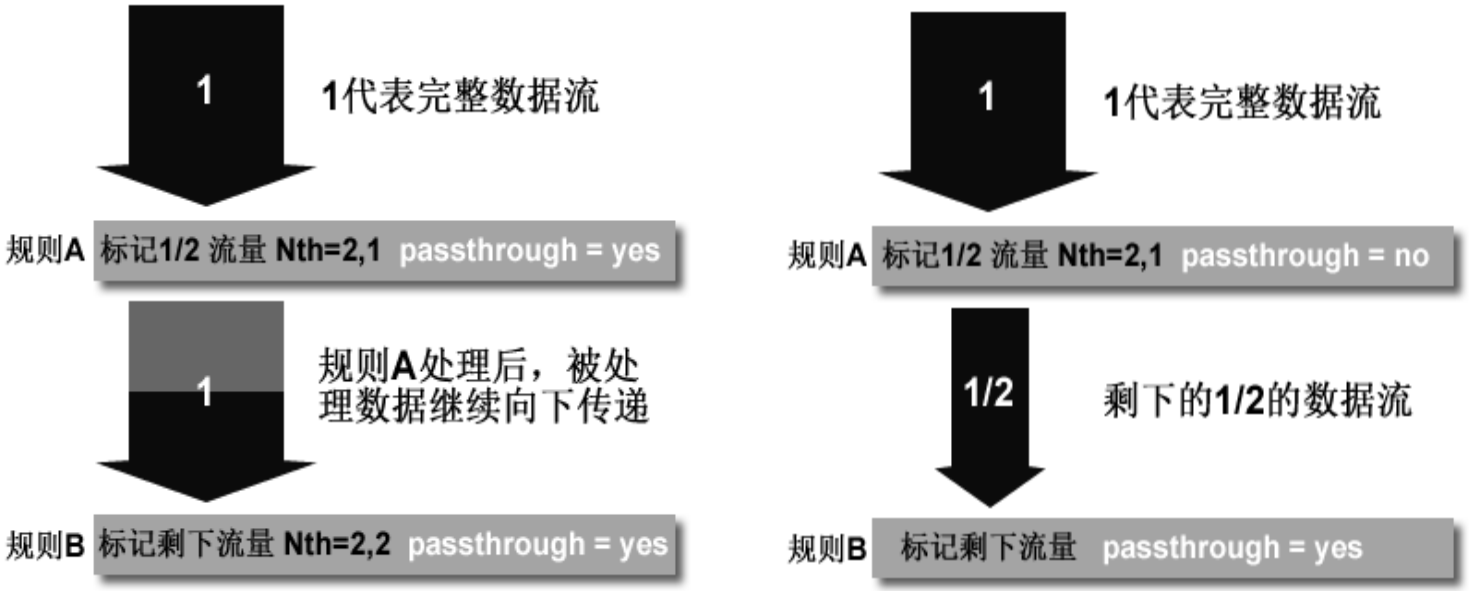
- 在RouterOS中实现相同的Nth结果时，有两种方法配置规则，改变Passthrough参数（Passthrough为是否将该规则数据继续向下传递，**no**为停止向下传递，**yes**则相反继续向下传递）。即两种不同的规则配置方法。

- 首先要知道Mangle标记捕获数据是采用FIFO先进先出算法，即从上往下执行，我们在配置Mangle的Nth规则，需要注意前后顺序



Passthrough对Nth的控制

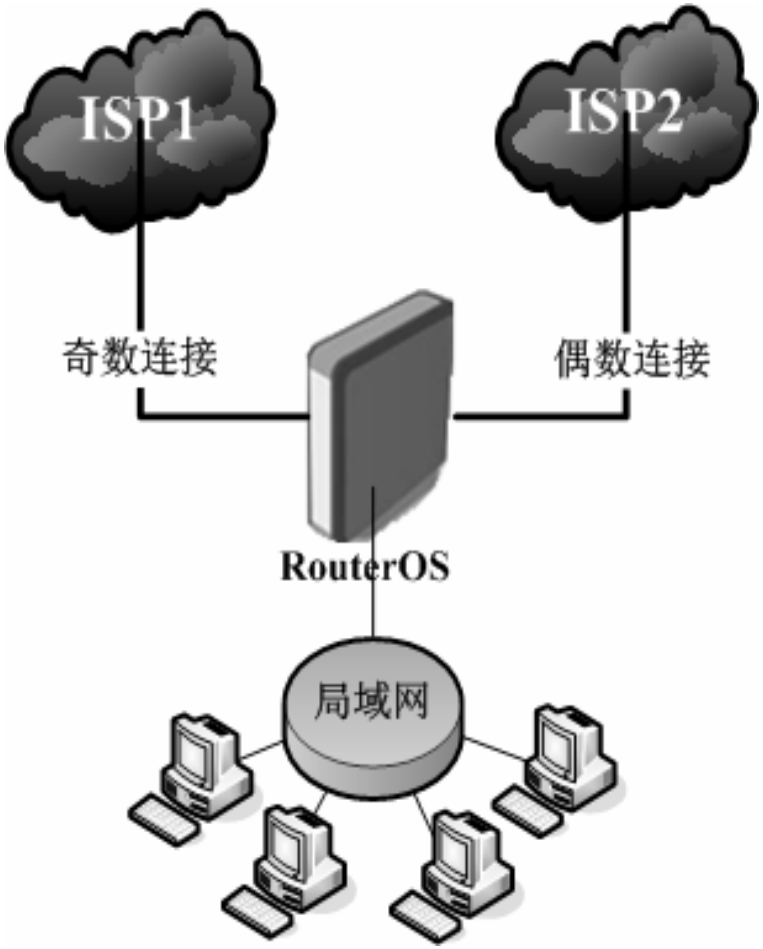
- 把数据流标记为两个组，即一条为1/2，另一条也为1/2，把一个数据流看成“1”，而我们把可以通过两种方法配置：



Passthrough对Nth的控制

- 当我们需要将数据流标记3组时，即每条规则为1/3。配置方法同样有两种，如下图

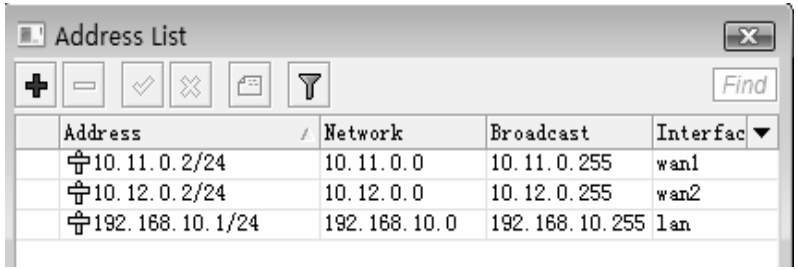
Nth事例1/4



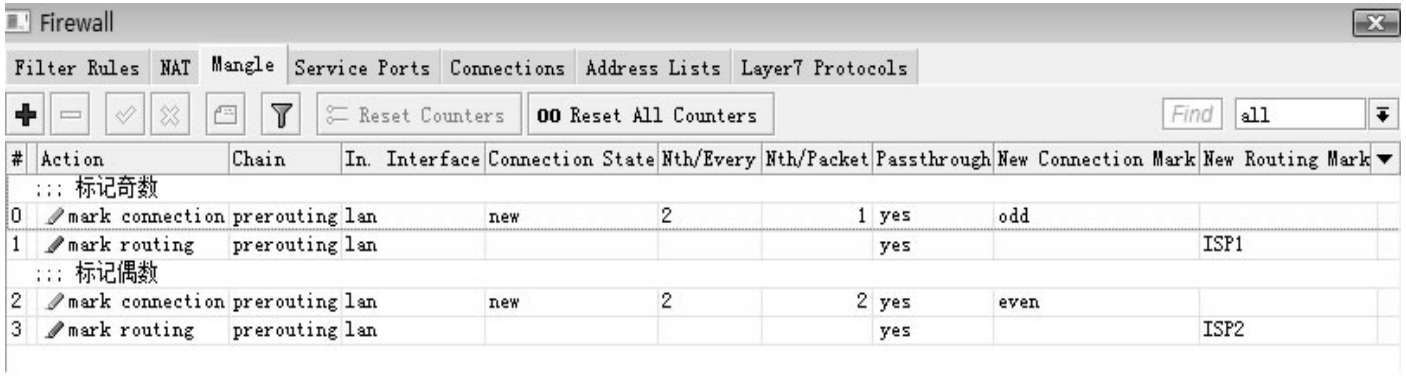
- 根据Nth的原理我们可以将来至内网的联接分为两组，即一组为奇数连接、一组为偶数连接，即奇数走一条线路，偶数走另外一条线路。
- 因为我们定义的是连接状态为new，即新建立的连接，对正常的访问没有任何影响，每次新建立所产生的后续数据都会按照原来的线路连接。
- 我们从所有的连接中，提取每次新建立的连接connection=new，并对他们做Nth的标记，将这些连接中相关的奇数（odd）包和偶数（even）包分离开，并走两个不同的网关（ISP1与ISP2）出去。这样就能保持每次连接的持续性。

Nth事例 2/4

IP地址配置

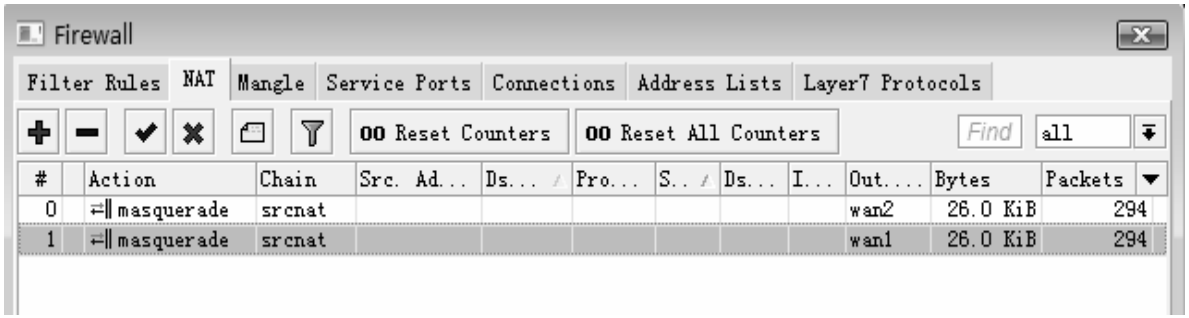


在ip firewall mangle中标记奇数和偶数的Nth，并配置路由标记，奇数Nth连接标记取名为odd，偶数连接标记取名为even，将奇数的路由标记取名为ISP1，将偶数的路由标记取名为ISP2

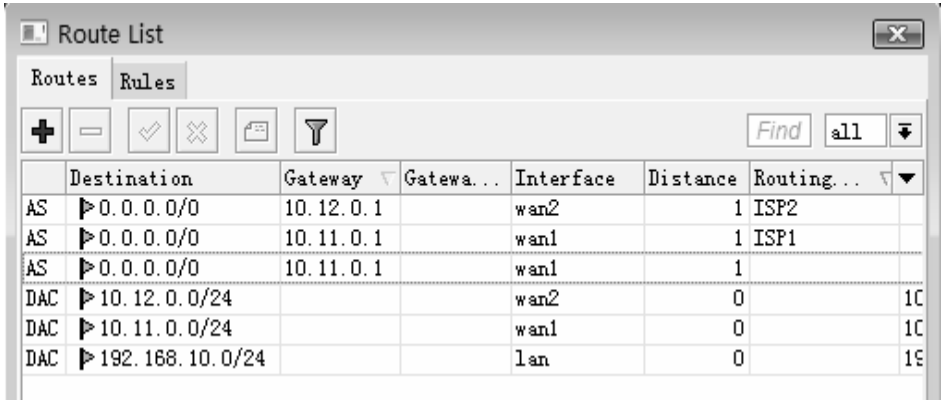


Nth事例 3/4

NAT配置



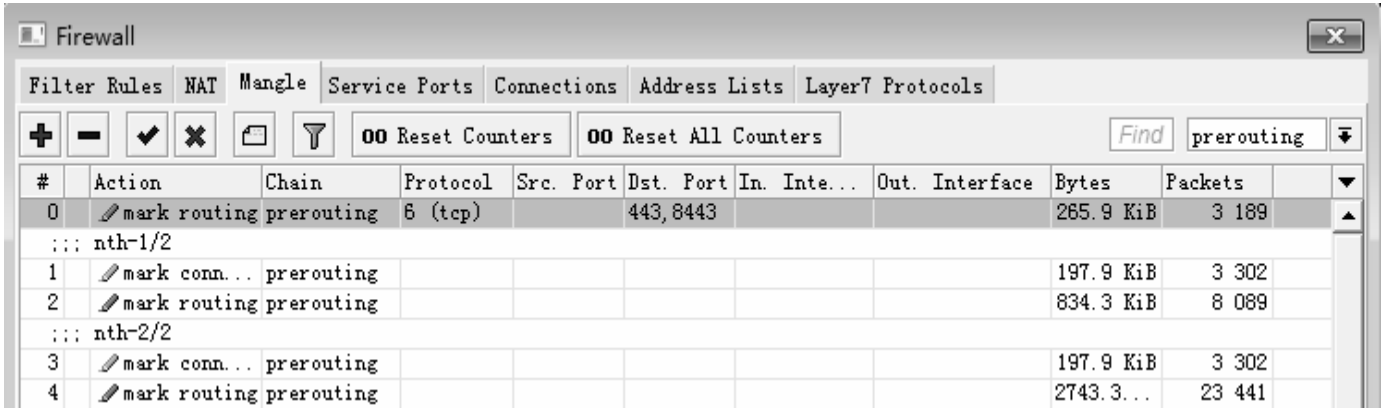
进入ip route中配置路由规则，配置10.12.0.1对应ISP2的路由标记，10.11.0.1对应ISP1的路由标记，我们用10.11.0.1作为路由器本身的默认网关。



Nth事例 4/4

- Nth会出现一个问题，即一些网银和要求IP验证的网站无法正常打开，因为每次连接都会通过不同的IP地址出去。

- 解决这个问题，只需要指定TCP协议的443和8443端口到一条固定的线路上，而且必须执行在Nth规则之前，并设置passthrough=no。



RouterOS PCC

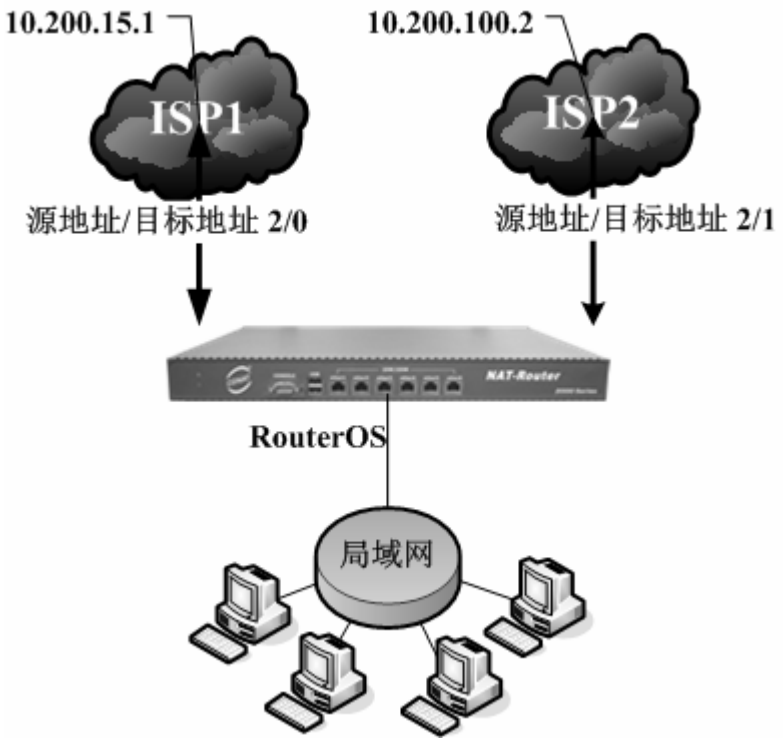
- PCC（per-connection-classifier） 每次连接分类器
- 通过哈希散列算法，将每次连接的地址或者端口进行分类
- 稳定性优于Nth

PCC基本配置

- /ip firewall mangle add chain=prerouting action=mark-connection new-connection-mark=1st_conn per-connection-classifier=both-addresses:3/0
- /ip firewall mangle add chain=prerouting action=mark-connection new-connection-mark=2nd_conn per-connection-classifier=both-addresses:3/1
- /ip firewall mangle add chain=prerouting action=mark-connection new-connection-mark=3rd_conn per-connection-classifier=both-addresses:3/2
- per-connection-classifier=both-addresses:3/0，这条规则的含义为我们对双向地址进行分类，3/0为一共有3条出口，定义第一条，3/1则是第二条，以此类推。

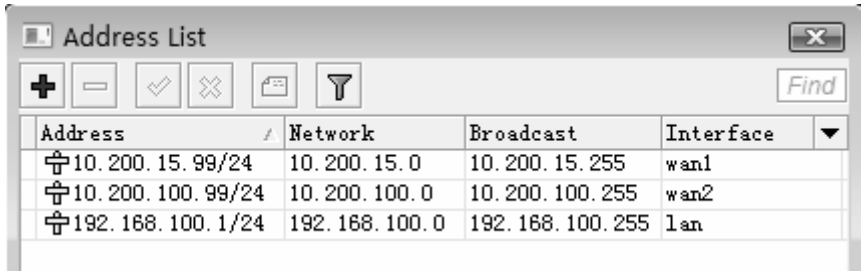
双线PCC事例

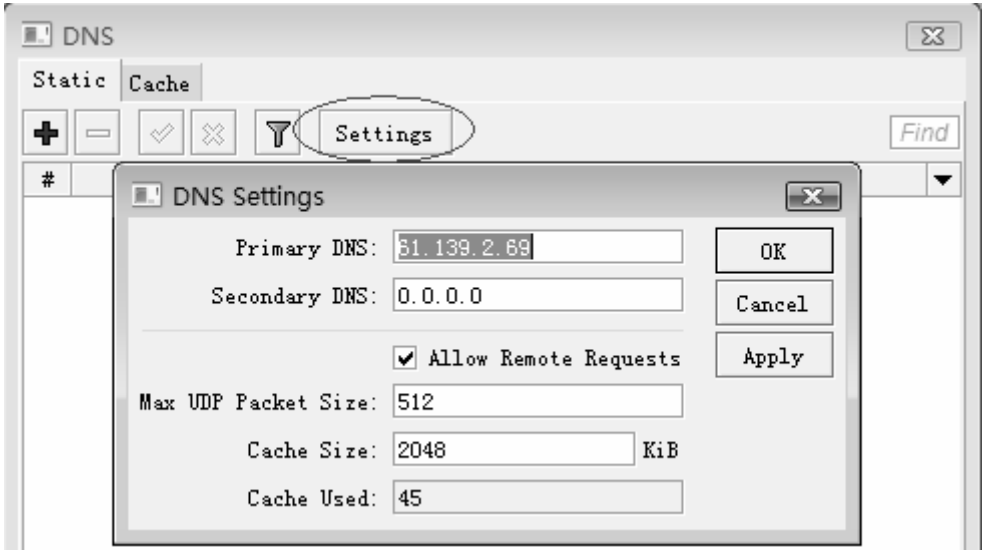
- 通分组双向地址实现负载平衡，这里我们建立2个WAN出口分别是wan1和wan2，网络环境如下：
- ISP1地址10.200.15.99/24，网关：10.200.15.1；
- ISP2地址10.200.100.99/24，网关：10.200.100.2；
- 内网IP地址192.168.100.1/24；
- 启用DNS缓存功能，用192.168.100.1作内网DNS解析；



PCC配置 1

添加IP地址和DNS





PCC 配置 2

- 进入ip firewall mangle里在prerouting里添加第一组规则，设置per-connection-classifier=both-address:2/0，并从连接里提取路由标记

```
/ip firewall mangle

add action=mark-connection chain=prerouting dst-address-type=!local \
    new-connection-mark=1st_conn passthrough=yes per-connection-classifier=\
    both-addresses:2/0 src-address=192.168.100.0/24

add action=mark-routing chain=prerouting connection-mark=1st_conn \
    new-routing-mark=1st_route passthrough=yes src-address=192.168.100.0/24
```

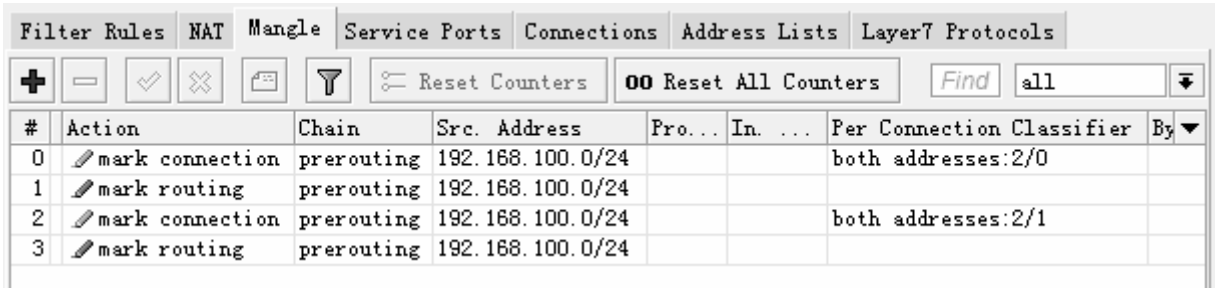
PCC 配置 3

- 进入ip firewall mangle里在prerouting里添加第一组规则，设置per-connection-classifier=both-address:2/1，并从连接里提取路由标记

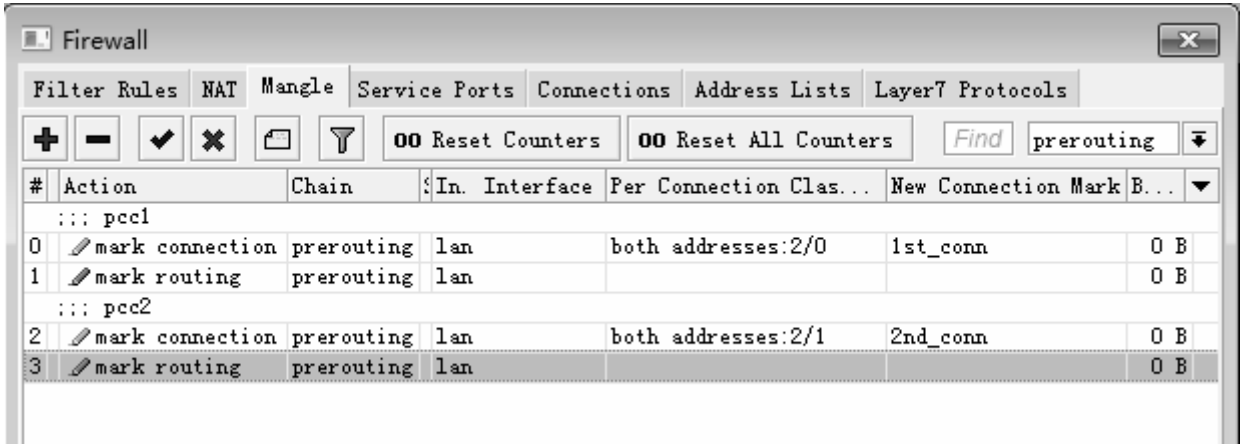
```
/ip firewall mangle

add action=mark-connection chain=prerouting dst-address-type=!local \
    new-connection-mark=2nd_conn passthrough=yes per-connection-classifier=\
    both-addresses:2/1 src-address=192.168.100.0/24

add action=mark-routing chain=prerouting connection-mark=2nd_conn \
    new-routing-mark=2nd_route passthrough=yes src-address=192.168.100.0/24
```

PCC 配置 4

使用src-address=192.168.100.0/24的地址范围标记PCC，通常是PPPoE拨号用户使用，这样的方式，如小区PPPoE认证负载均衡的RouterOS，如下图

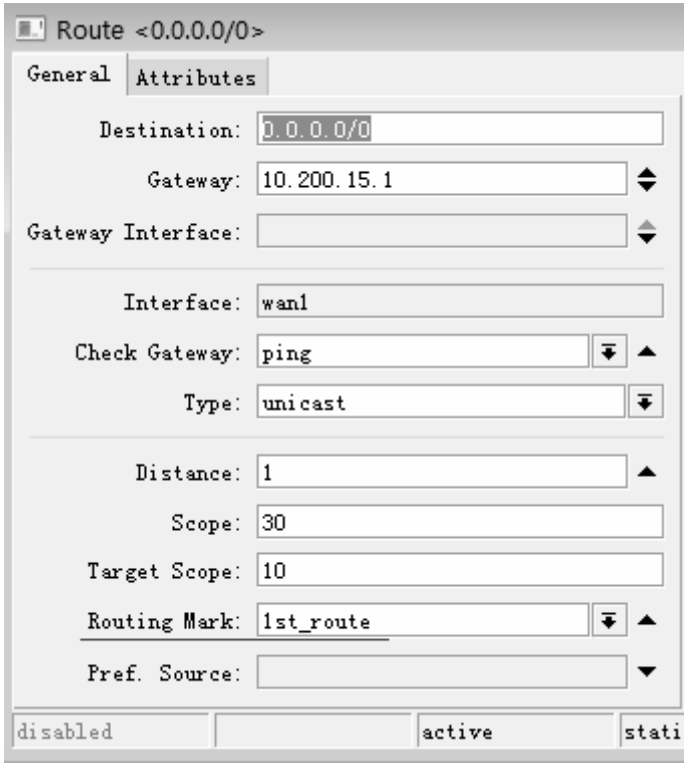


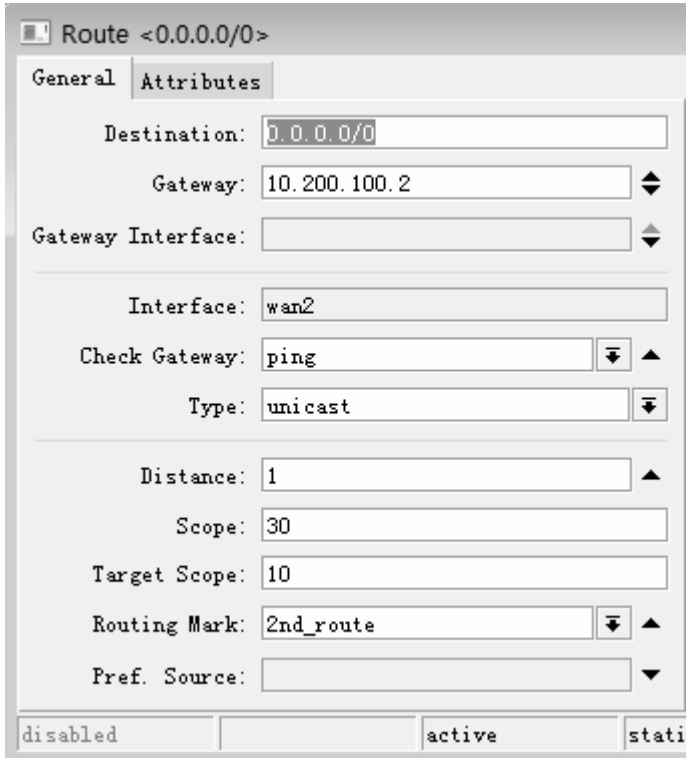
我们也采用指定in-interface的方式代替src-address的地址范围标记PCC，通常是使用固定IP上网的办公室和网吧，可以选择这样的方式，如下图



PCC 路由配置 1

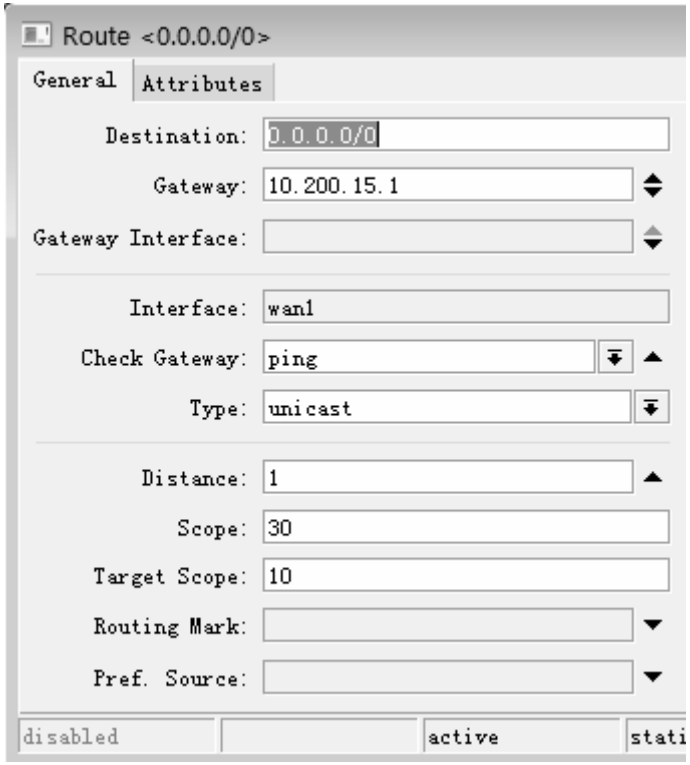
- 配置完标记后路由后，我们进入ip route配置路由，设置负载均衡的标记路由，分别添加两天规则，设置对应的网关和对应的routing-mark路由标记，如下图

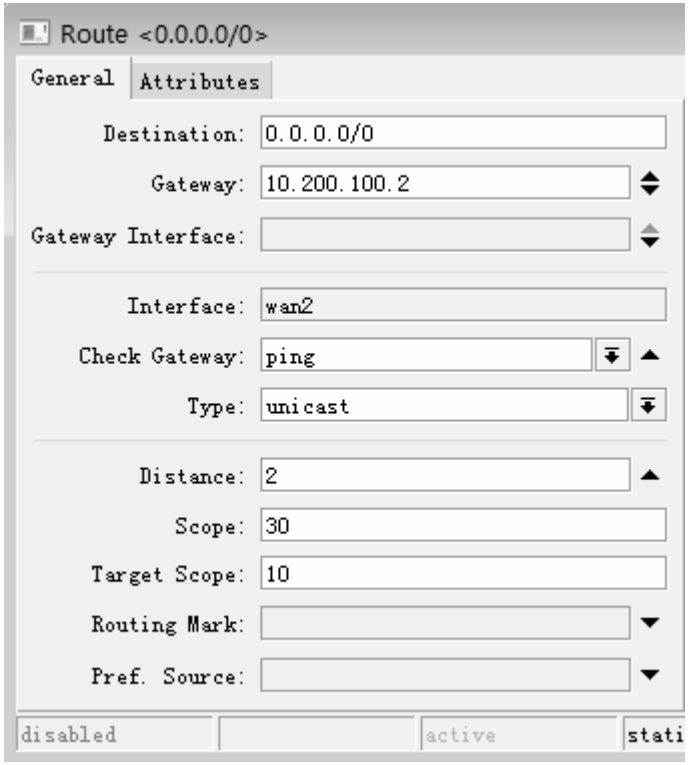




PCC 路由配置 2

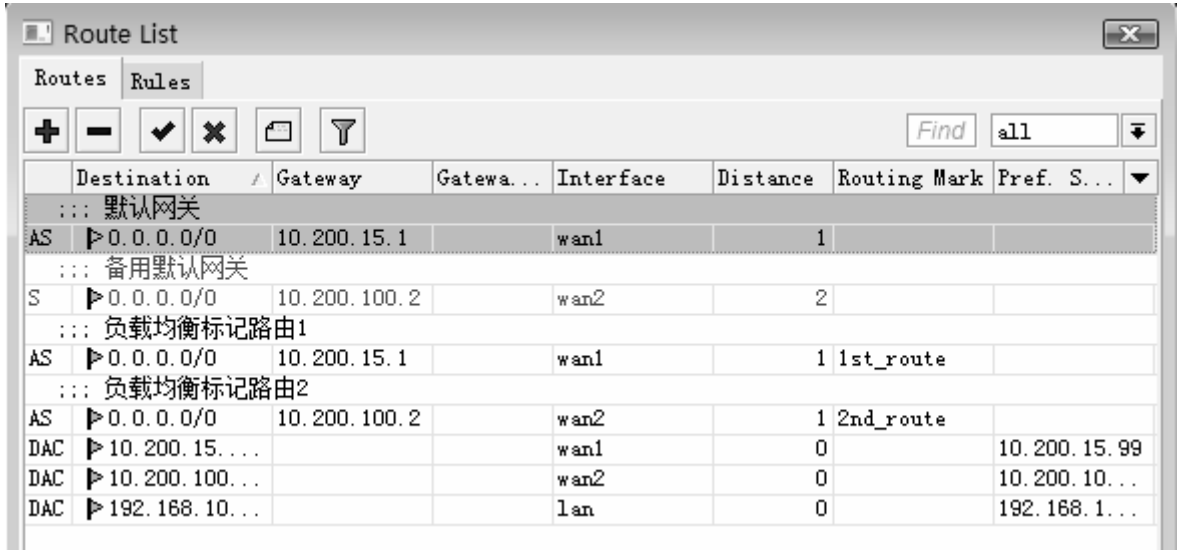
- 添加完网关后，我们在设置路由的默认网关和备份线路规则，默认网关distance1，备份网关distance设置为2





PCC 路由配置 2

- 配置完成后如下图：



PCC 路由配置 回程路由

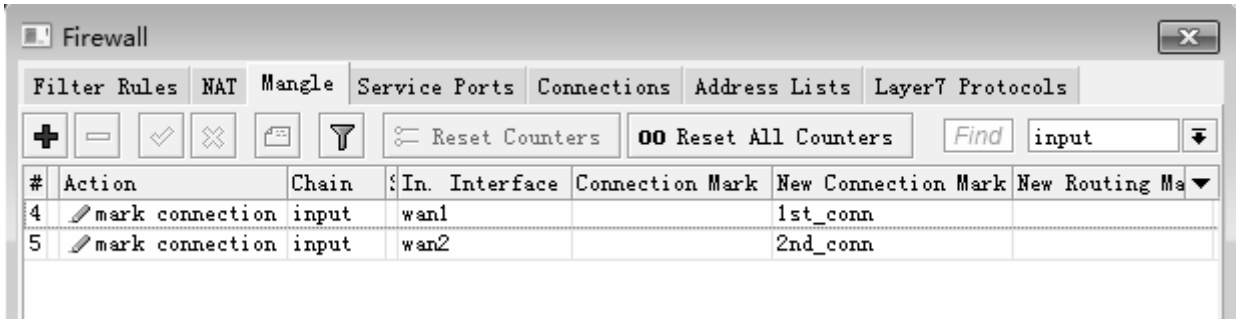
回程路由设置

- 我们需要将从某一个wan口进入的数据这个口返回给访问端，即能保证管理员在外网能从任何一个wan口进入路由器访问

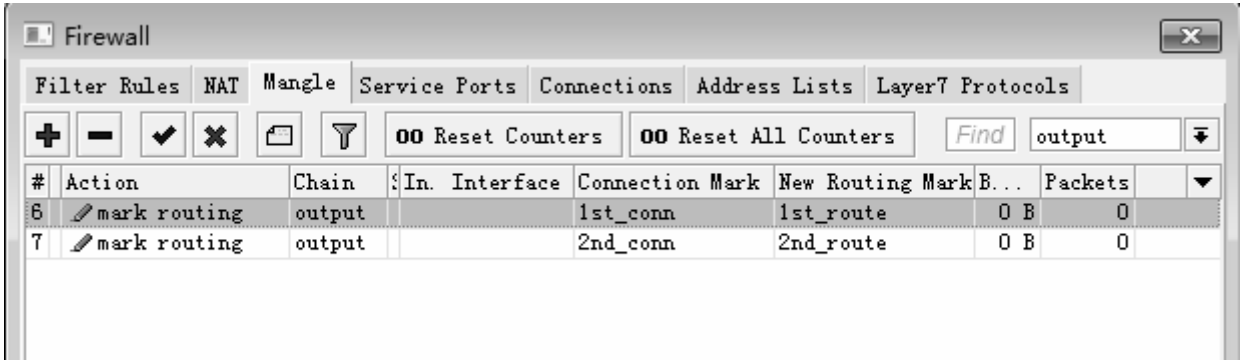
- 我们只需要在mangle的input链表里标记从指定wan口进入的数据，并将他们通过output链表返回到相同的路由即可

PCC 路由配置 回程路由

进入ip firewall mangle里在input链表添加2条in-interface=wan1和in-interface=wan2的连接标记，并分别归入1st_conn和2nd_conn连接里

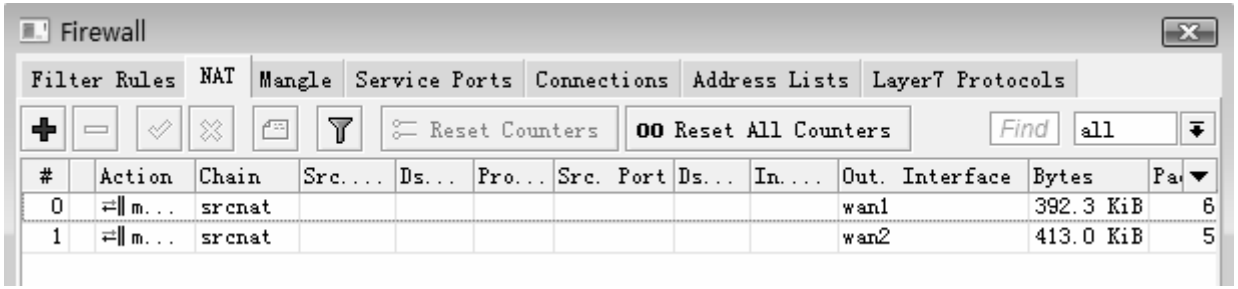


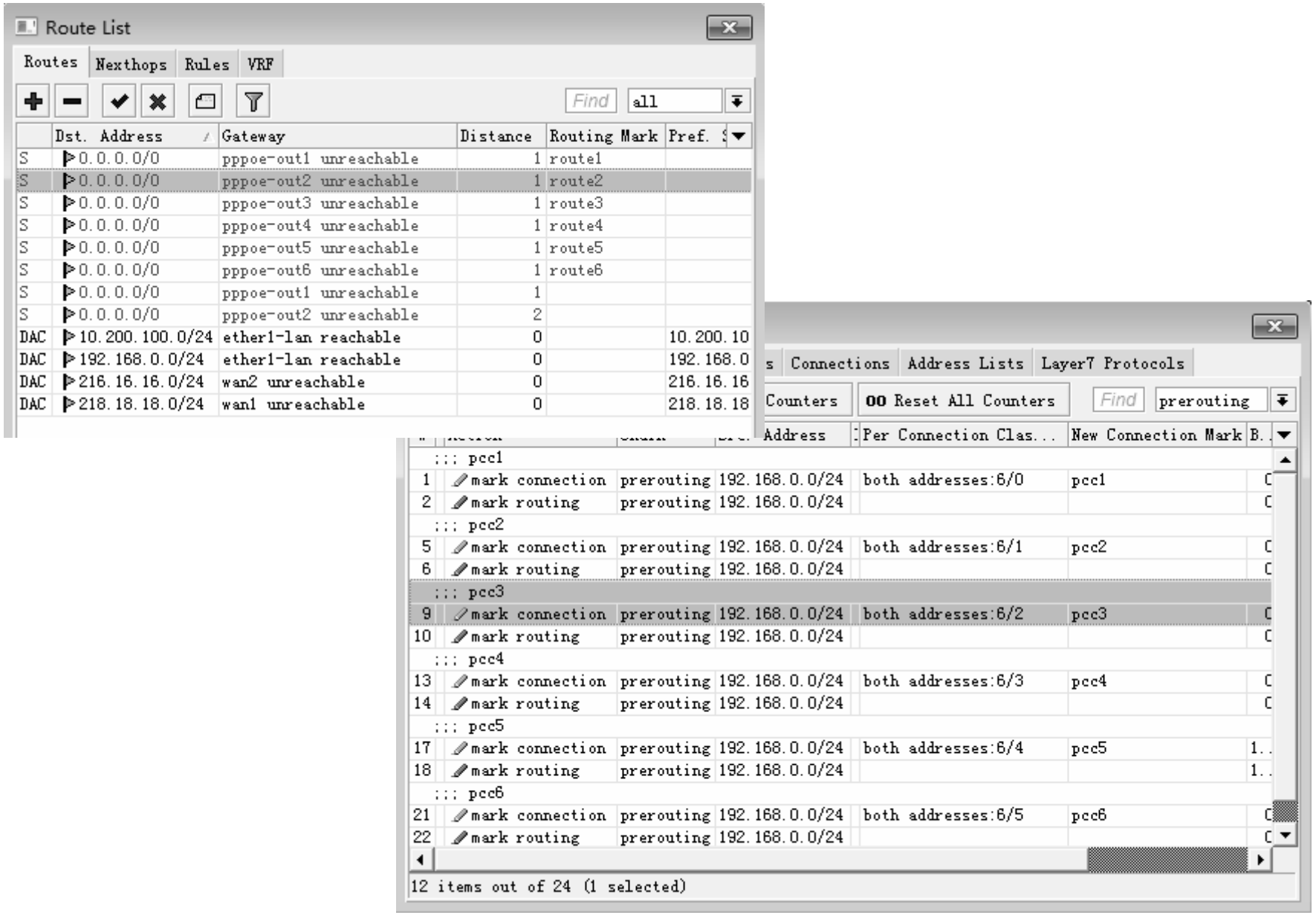
然后在ip firewall mangle的output链表中，将1st_conn和2nd_conn也归入1st_route和2nd_route里



PCC NAT转换

- 最后，我们完成NAT规则的配置，将wan1和wan2分配添加两条masquerade的伪装规则




Route List
Routes
Nexthops
Rules
VRF
Find
all
Dst. Address
Gateway
Distance
Routing Mark
Pref.
S 0.0.0.0/0 pppoe-out1 unreachable 1 route1
S 0.0.0.0/0 pppoe-out2 unreachable 1 route2
S 0.0.0.0/0 pppoe-out3 unreachable 1 route3
S 0.0.0.0/0 pppoe-out4 unreachable 1 route4
S 0.0.0.0/0 pppoe-out5 unreachable 1 route5
S 0.0.0.0/0 pppoe-out6 unreachable 1 route6
S 0.0.0.0/0 pppoe-out1 unreachable 1
S 0.0.0.0/0 pppoe-out2 unreachable 2
DAC 10.200.100.0/24 ether1-lan reachable 0 10.200.10
DAC 192.168.0.0/24 ether1-lan reachable 0 192.168.0
DAC 216.16.16.0/24 wan2 unreachable 0 216.16.16
DAC 218.18.18.0/24 wan1 unreachable 0 218.18.18
Connections
Address Lists
Layer7 Protocols
Counters
Reset All Counters
Find
prerouting
Address
Per Connection Clas...
New Connection Mark
;;; pcc1
1 mark connection prerouting 192.168.0.0/24 both addresses:6/0 pcc1
2 mark routing prerouting 192.168.0.0/24
;;; pcc2
5 mark connection prerouting 192.168.0.0/24 both addresses:6/1 pcc2
6 mark routing prerouting 192.168.0.0/24
;;; pcc3
9 mark connection prerouting 192.168.0.0/24 both addresses:6/2 pcc3
10 mark routing prerouting 192.168.0.0/24
;;; pcc4
13 mark connection prerouting 192.168.0.0/24 both addresses:6/3 pcc4
14 mark routing prerouting 192.168.0.0/24
;;; pcc5
17 mark connection prerouting 192.168.0.0/24 both addresses:6/4 pcc5
18 mark routing prerouting 192.168.0.0/24
;;; pcc6
21 mark connection prerouting 192.168.0.0/24 both addresses:6/5 pcc6
22 mark routing prerouting 192.168.0.0/24
12 items out of 24 (1 selected)

PCC配置脚本

```
:global interface "ether1-lan"
:global pppoe "pppoe-out"
:global address "192.168.0.0/24"
:global n 6
:for i from=1 to=$n do={
        :log info "a1"
        /ip firewall mangle add chain=input in-interface=($pppoe . $i) action=mark-
connection new-connection-mark=("pcc" . $i)
        :log info "a2"
        /ip firewall mangle add chain=prerouting src-address=$address per-
connection-classifier=("both-addresses:" . $n . "/" . $i-1) dst-address-type=!local
action=mark-connection new-connection-mark=("pcc" . $i)
        /ip firewall mangle add chain=prerouting src-address=$address
connection-mark=("pcc" . $i) action=mark-routing new-routing-mark=("route" . $i)
        /ip firewall mangle add chain=output connection-mark=("pcc" . $i)
action=mark-routing new-routing-mark=("route" . $i)
}
```

RouterOS QoS

RouterOS Queue类型

常见的流控规则有**2**种

FIFO – 先进先出法，规定队列长度（包括bytes或Packets）

- Simple queue中默然使用的是PFIFO

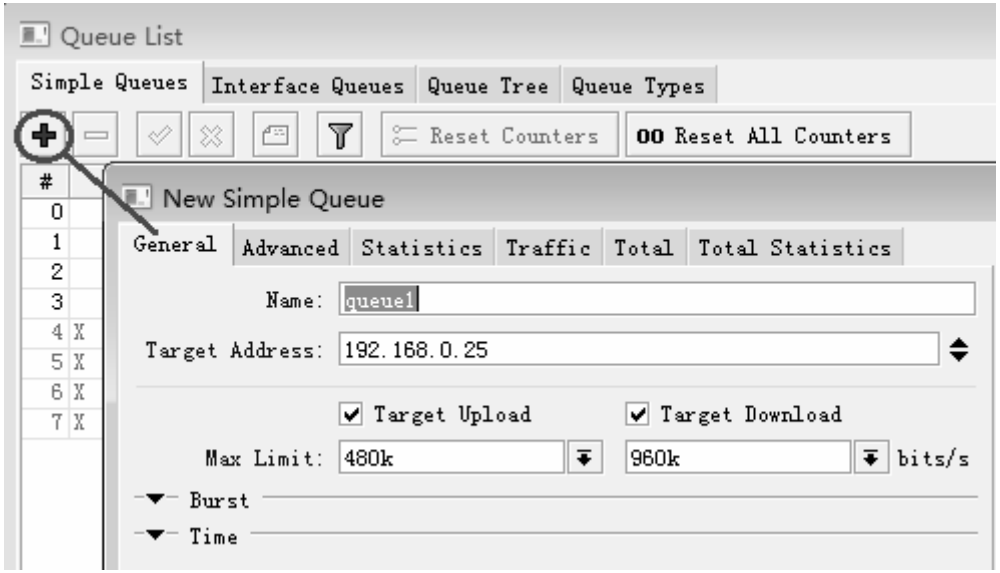
PCQ – 每次连接队列，一种高效的流量控制算法

优先级控制策略

HTB – 等级令牌桶，基于Queue tree的一种等级优先的流量控制策略

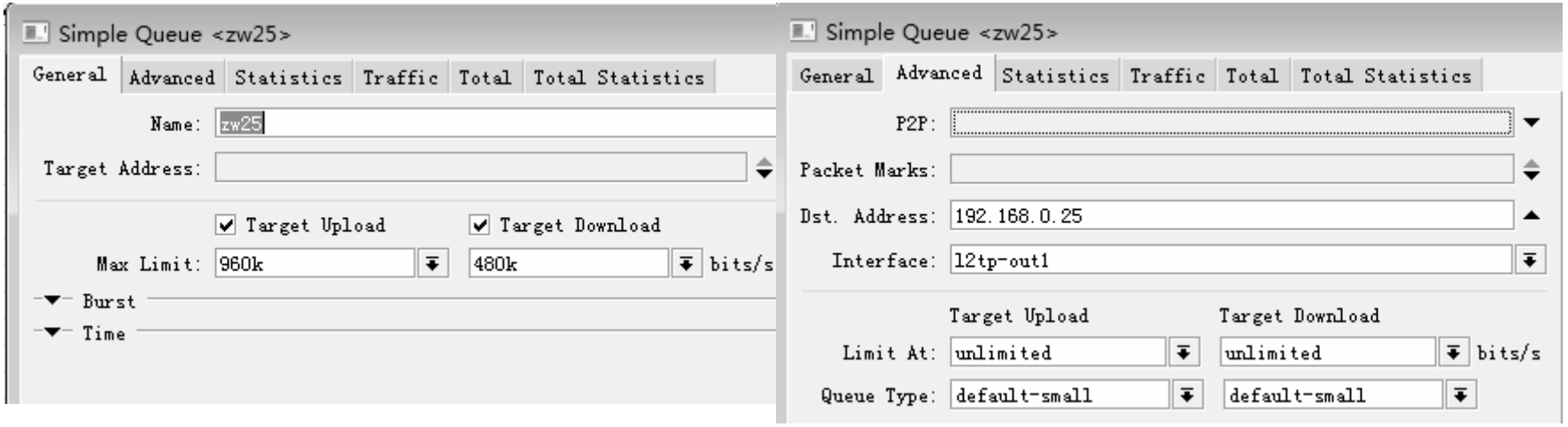
Simple Queue 1

- 简单队列的配置在queue simple中添加规则，下面是192.168.0.25的主机做流量控制。
- target-address设置主机的IP地址，Max-limit的upload是上行带宽480kbps，download是下行带宽为960Kbps

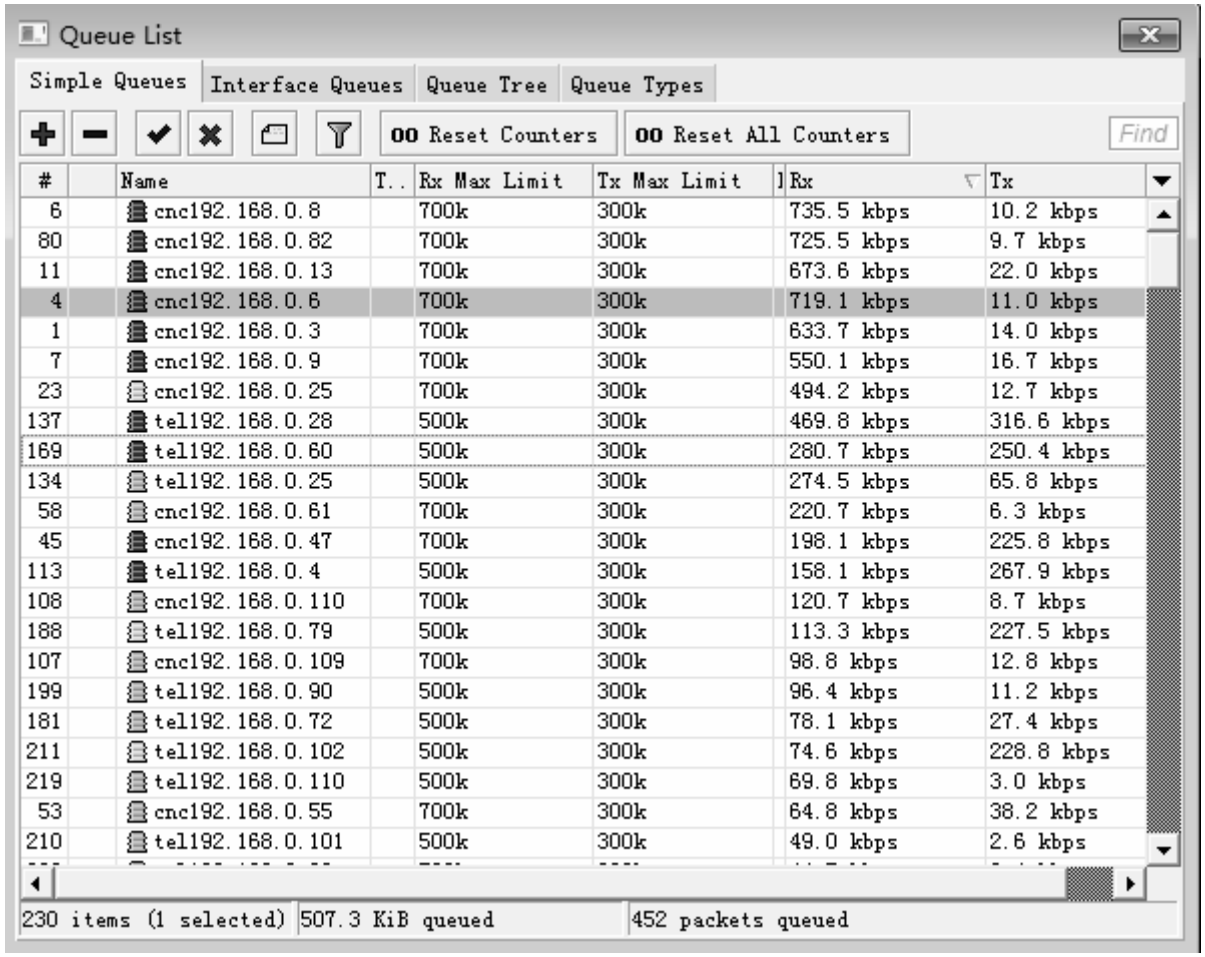


Simple Queue 2

- 对接口的带宽控制，如果是多条线路的情况下，我们可以通过simple queue定义每个主机流量离开这个接口的带宽
- 我们不在定义target-address参数，而是选择dst-address，并选择interface，这里的接口选择的是l2tp-out1，即电信的VPN专网。
- 但是，这里的带宽控制需要相反填写，即upload填写下行，download填写上行数据



Simple队列的单机流量查看



Simple queue

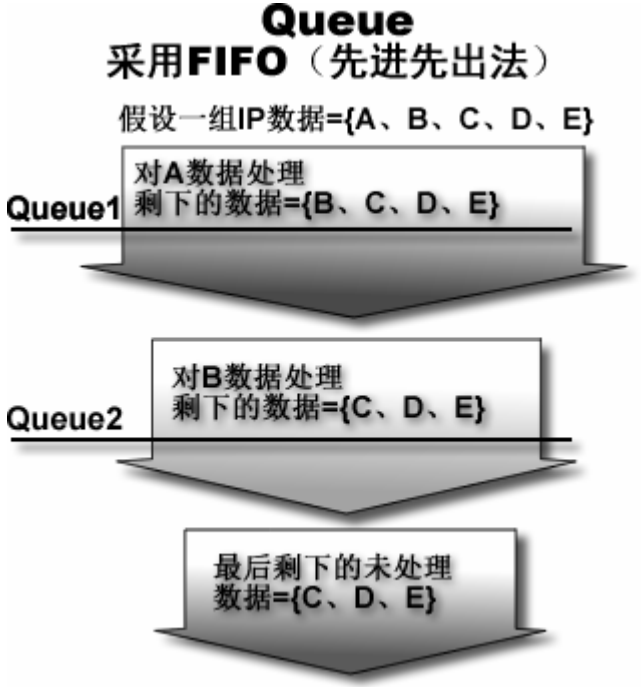
- 在Queue中我们常用的是Simple queue

- Simple queue与firewall规则类似

规则越多，处理的数据越多,CPU消耗越大

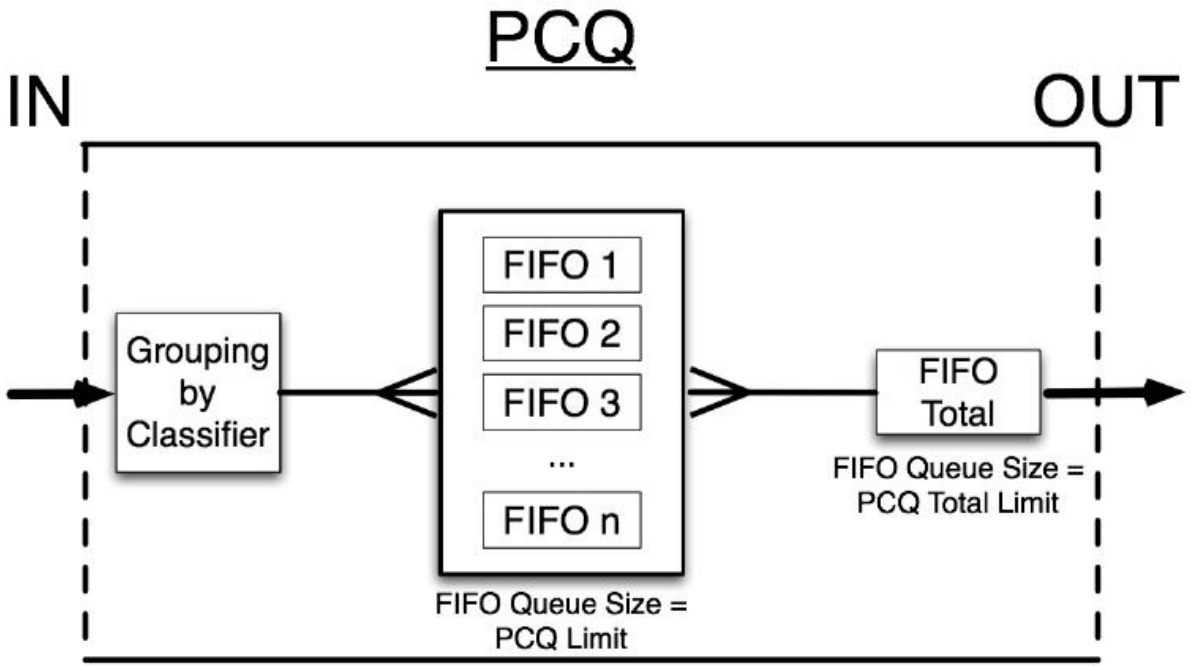
规则越多，后面的规则获取带宽的几率越小

如果有1000条Simple queue规则，那必须判断查询999条规则（必要时减少queue数量）

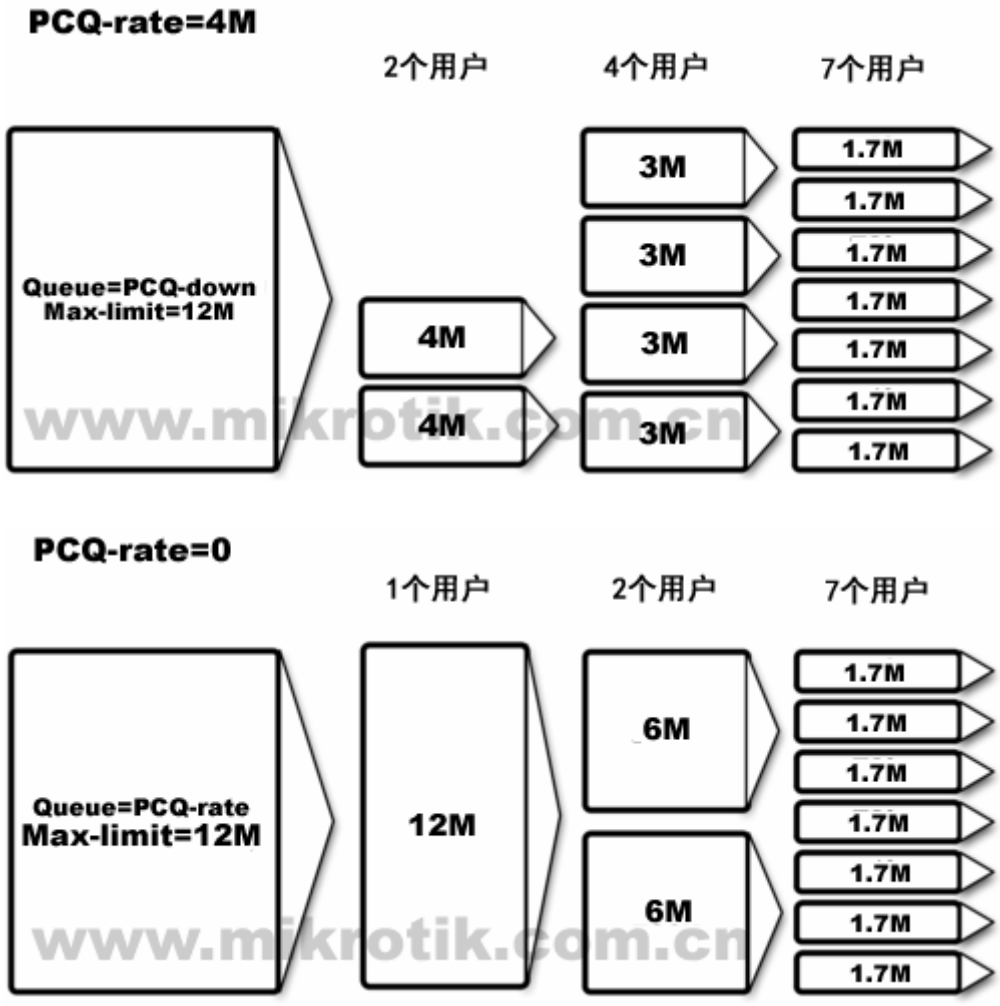


PCQ原理1

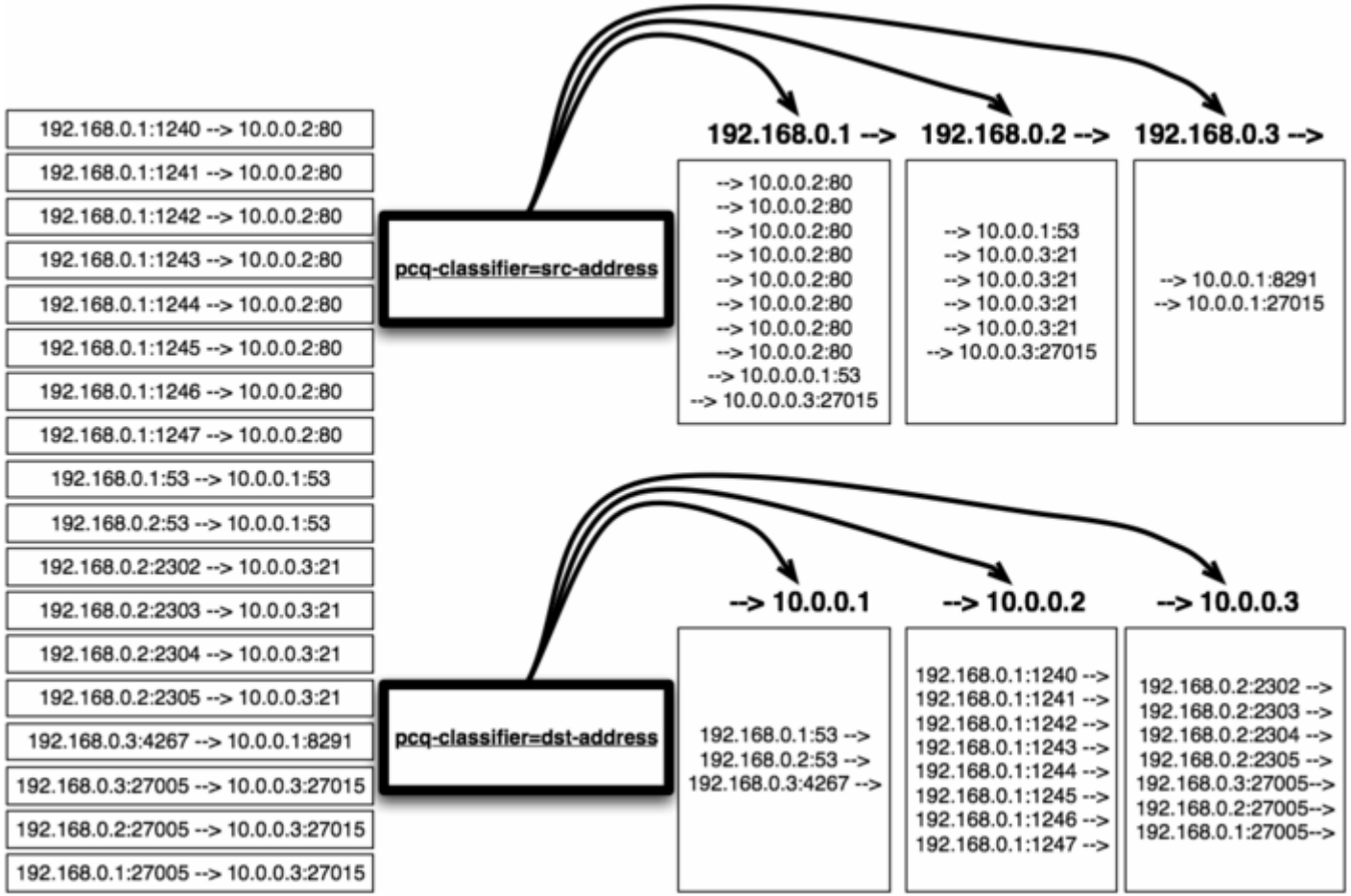
- 每次连接分类器，采用动态监测当前连接的主机和端口，并进行分类，区分IP和端口信息，对相同IP地址或端口进行分类流控



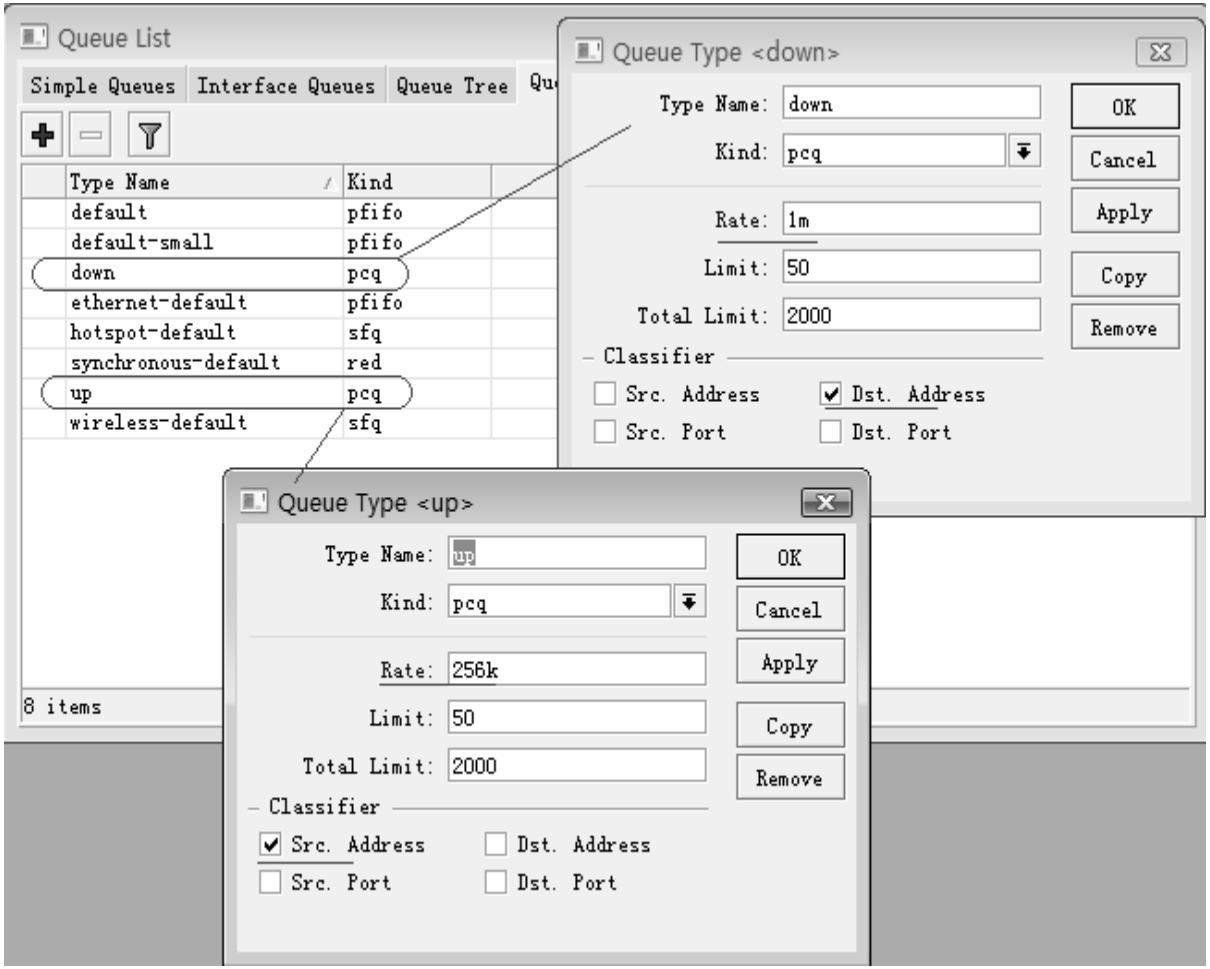
PCQ原理 2


PCQ-rate=4M
2个用户
4个用户
7个用户
Queue=PCQ-down
Max-limit=12M
4M
4M
3M
3M
3M
3M
1.7M
1.7M
1.7M
1.7M
1.7M
1.7M
1.7M
www.mikrotik.com.cn
PCQ-rate=0
1个用户
2个用户
7个用户
Queue=PCQ-rate
Max-limit=12M
12M
6M
6M
1.7M
1.7M
1.7M
1.7M
1.7M
1.7M
1.7M
www.mikrotik.com.cn

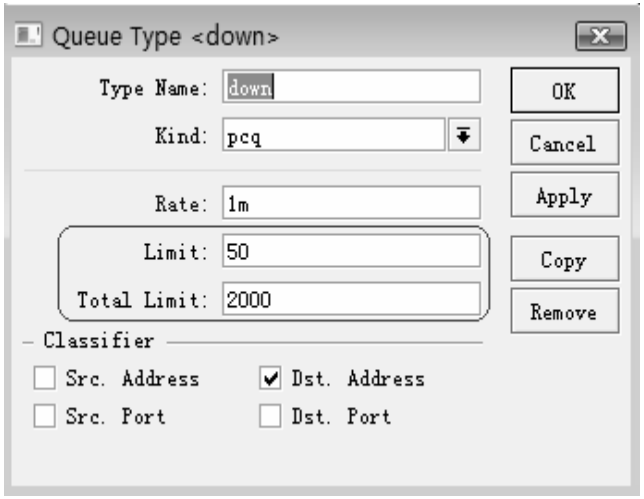
PCQ原理 3


192.168.0.1:1240 --> 10.0.0.2:80
192.168.0.1:1241 --> 10.0.0.2:80
192.168.0.1:1242 --> 10.0.0.2:80
192.168.0.1:1243 --> 10.0.0.2:80
192.168.0.1:1244 --> 10.0.0.2:80
192.168.0.1:1245 --> 10.0.0.2:80
192.168.0.1:1246 --> 10.0.0.2:80
192.168.0.1:1247 --> 10.0.0.2:80
192.168.0.1:53 --> 10.0.0.1:53
192.168.0.2:53 --> 10.0.0.1:53
192.168.0.2:2302 --> 10.0.0.3:21
192.168.0.2:2303 --> 10.0.0.3:21
192.168.0.2:2304 --> 10.0.0.3:21
192.168.0.2:2305 --> 10.0.0.3:21
192.168.0.3:4267 --> 10.0.0.1:8291
192.168.0.3:27005 --> 10.0.0.3:27015
192.168.0.2:27005 --> 10.0.0.3:27015
192.168.0.1:27005 --> 10.0.0.3:27015
pcq-classifier=src-address
192.168.0.1 -->
192.168.0.2 -->
192.168.0.3 -->
--> 10.0.0.2:80
--> 10.0.0.2:80
--> 10.0.0.2:80
--> 10.0.0.2:80
--> 10.0.0.2:80
--> 10.0.0.2:80
--> 10.0.0.2:80
--> 10.0.0.2:80
--> 10.0.0.1:53
--> 10.0.0.3:27015
--> 10.0.0.1:53
--> 10.0.0.3:21
--> 10.0.0.3:21
--> 10.0.0.3:21
--> 10.0.0.3:21
--> 10.0.0.3:27015
--> 10.0.0.1:8291
--> 10.0.0.1:27015
pcq-classifier=dst-address
--> 10.0.0.1
--> 10.0.0.2
--> 10.0.0.3
192.168.0.1:53 -->
192.168.0.2:53 -->
192.168.0.3:4267 -->
192.168.0.1:1240 -->
192.168.0.1:1241 -->
192.168.0.1:1242 -->
192.168.0.1:1243 -->
192.168.0.1:1244 -->
192.168.0.1:1245 -->
192.168.0.1:1246 -->
192.168.0.1:1247 -->
192.168.0.2:2302 -->
192.168.0.2:2303 -->
192.168.0.2:2304 -->
192.168.0.2:2305 -->
192.168.0.3:27005-->
192.168.0.2:27005-->
192.168.0.1:27005-->

PCQ winbox配置



PCQ参数



- 如果**total-limit=x**，那内存占用为**X***（**2000byte+200byte**）

1个包缓冲占用**2000byte**
1个包协议占用**200byte**

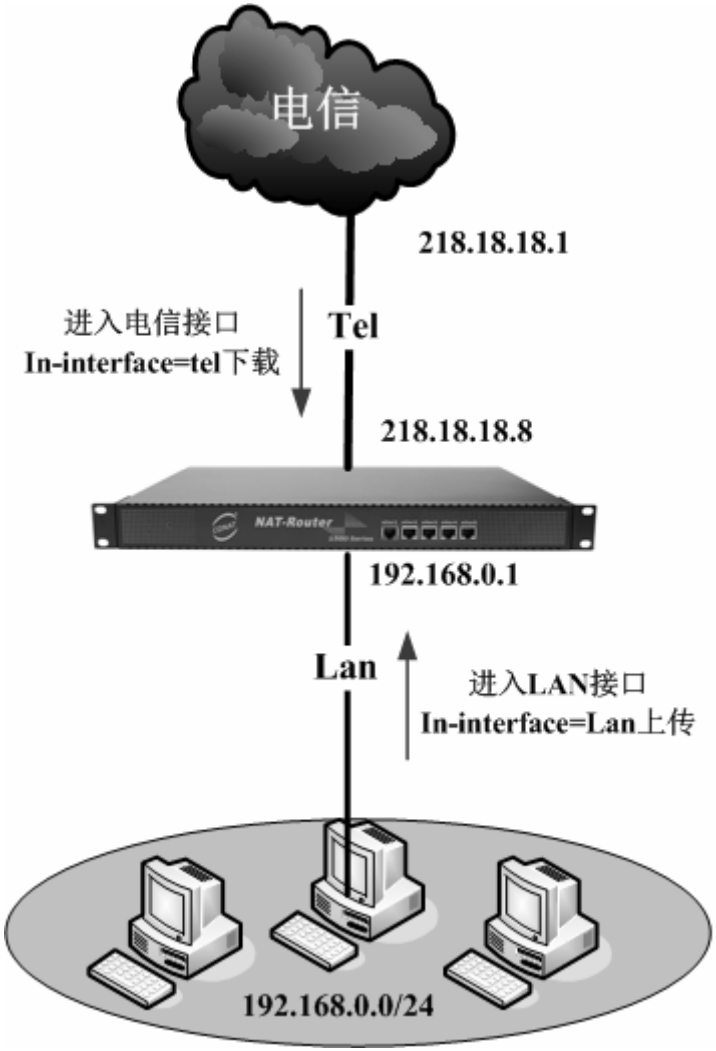
- **total-limit=2000=<4.2MB RAM**
- **total-limit=5000=<10.5MB RAM**

- 该规则仅能容纳**40**个用户（**total-limit/limit=2000/50=40**）
- 解决方法必须增加**total-limit**或者减少**limit**
- 但必须保证每个用户队列**(limit)**获取**10-20**个数据包

网吧单线接入

- 单线接入配置非常简单，与基本操作相同；
- 需要注意的是nat转换需要指定内网源地址和外网网卡，这样效率更高；
- nat中指定外网网卡在多线路下特别重要，这样保证线路的最近转换路径；
- 流量控制如果采用simple，需要使用脚本添加单机的流量控制规则；
- 流量控制如果采用动态，只需要标记上下行数据，设置PCQ。

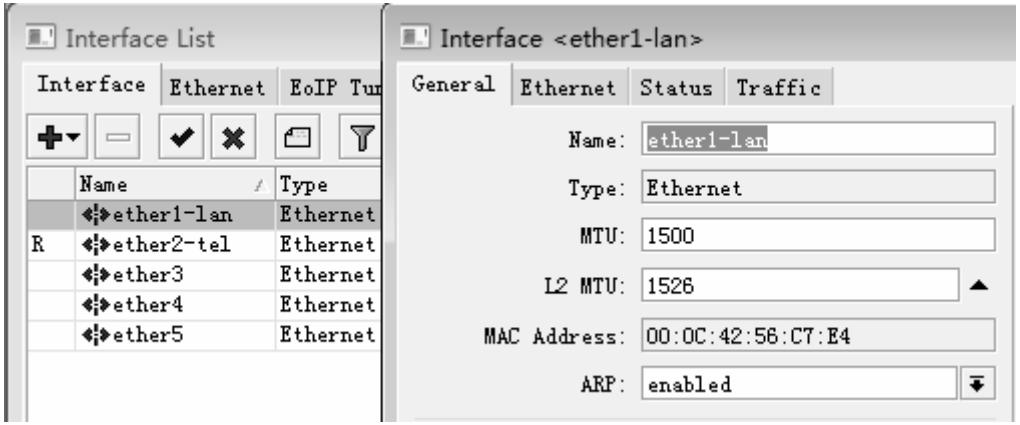
案例分析



- 电信单线接入，电信IP地址218.18.18.8，电信网关218.18.18.1
- 内网口IP地址192.168.0.1/24
- 单线接入只需要指定默认网关，和配置nat伪装规则
- 流量控制采用Simple固定限速或者PCQ动态流量控制

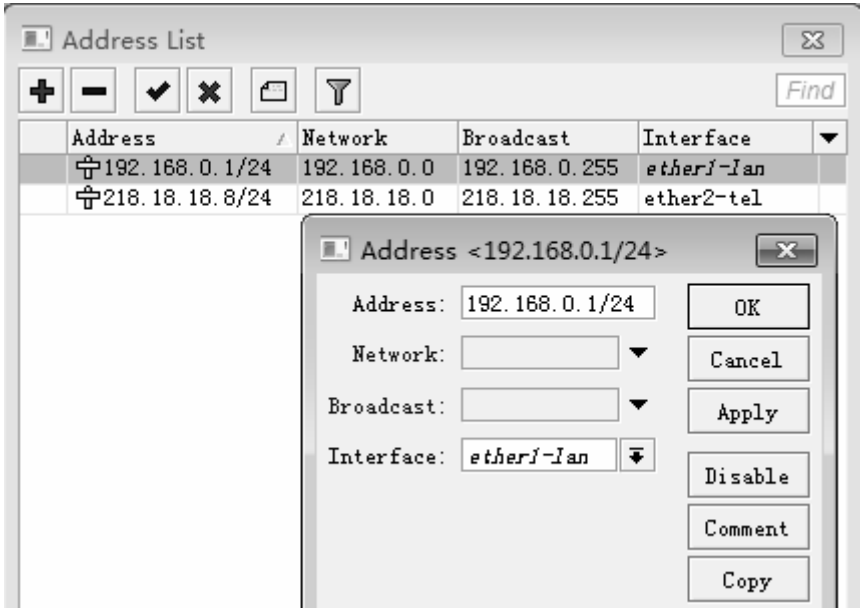
案例配置 – IP地址 （1）

- 首先定义网卡名称，以便识别对应的网卡
- 打开interface菜单，设置对应接口的网卡名称
- ether1设置为lan内网口，ether2设置为tel电信口



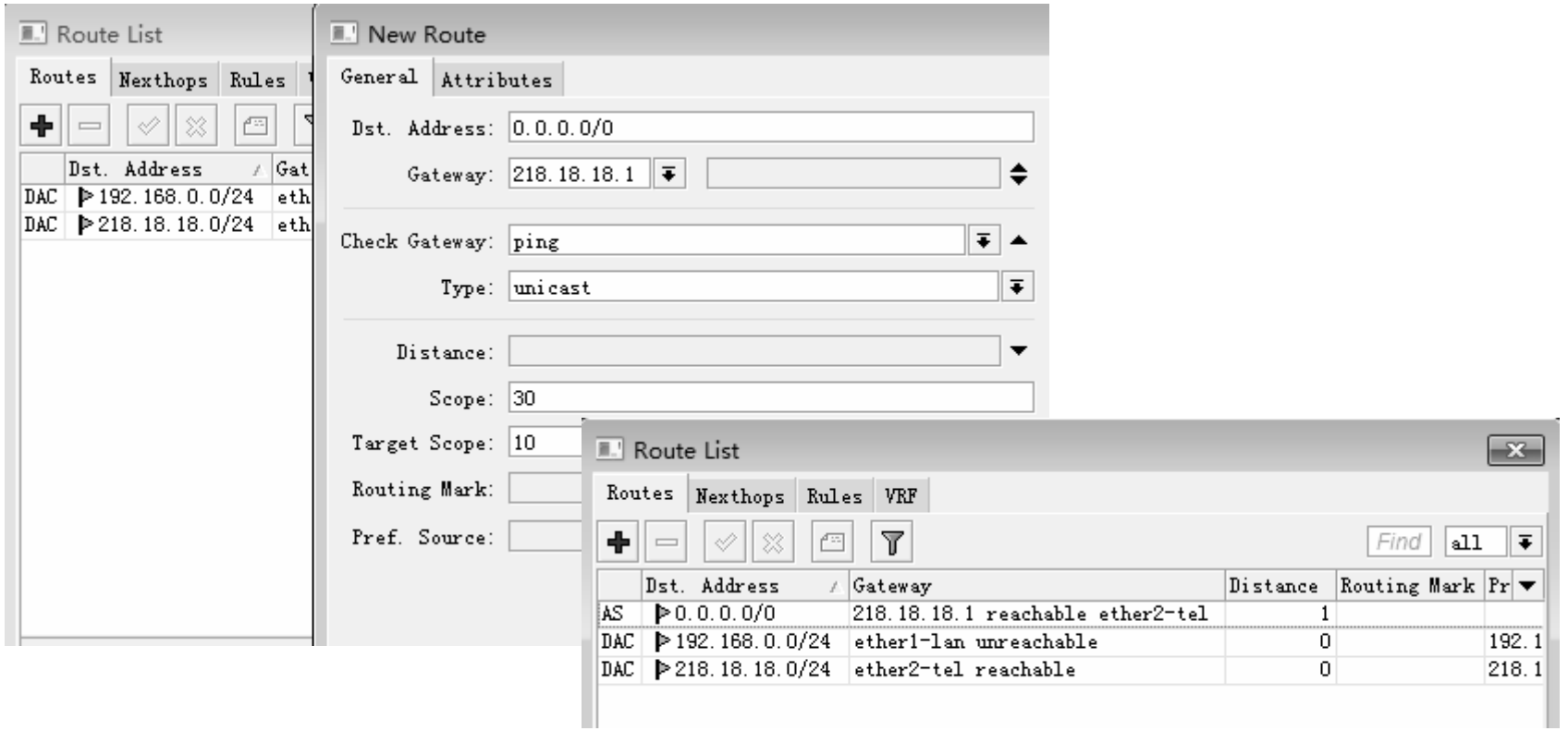
案例配置 – IP地址 （2）

- 进入ip address菜单配置对应网口的IP地址
- 将192.168.0.1的IP地址设置到ether1-lan,注意这里的子网掩码是255.255.255.0，而RouterOS支持按位显示即换算为24，在IP地址后用“/”间隔ip和子网掩码
- 将218.18.18.8/24的IP地址设置到ether2-tel



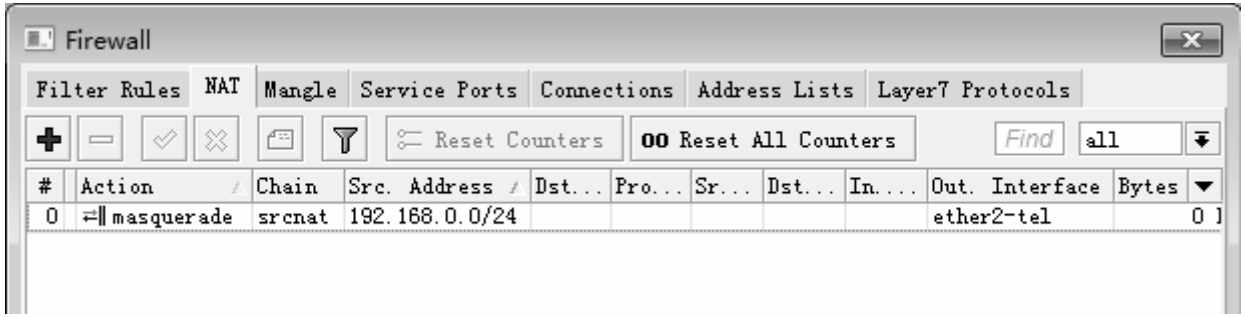
案例配置 – 路由

- 进入ip route添加默认网关，设置gateway=218.18.18.1，打开网关监测功能chek-gateway=ping



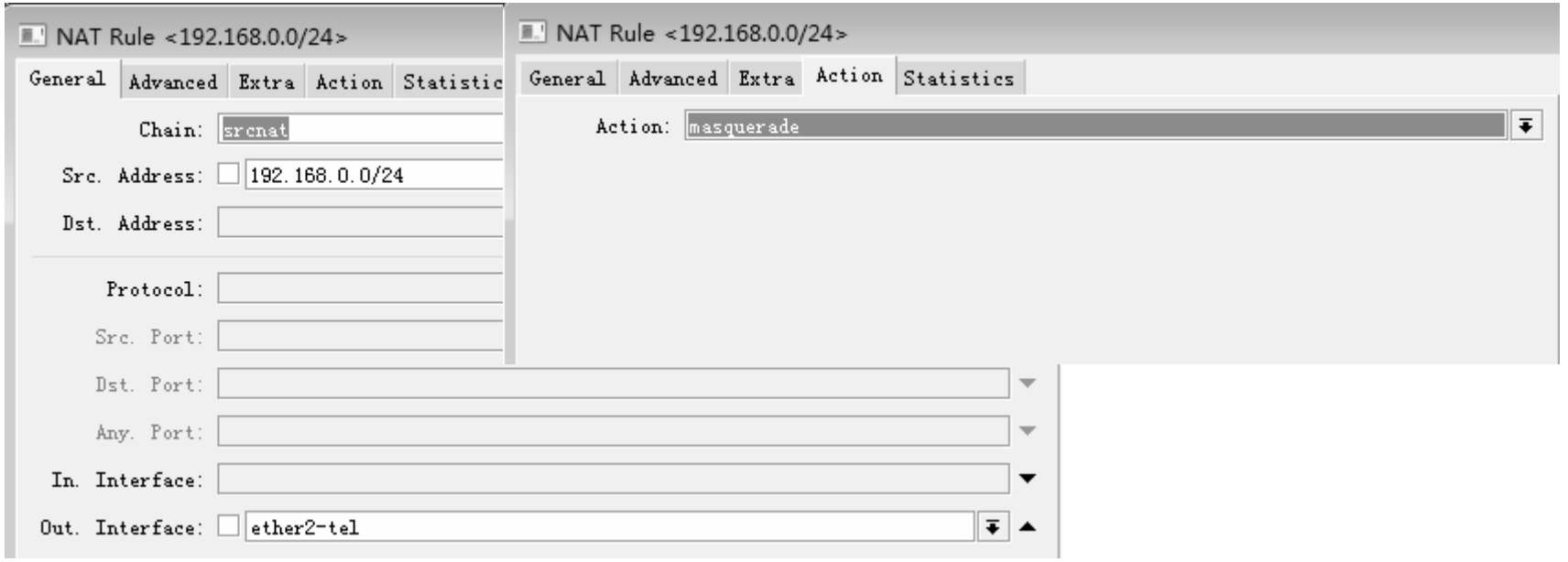
案例配置 – NAT （1）

- 进入ip firewall nat选择“srcnat链表”配置网络地址转换，
- NAT功能是将192.168.0.0/24私有地址转换为公网的218.18.18.8；



案例配置 – NAT （2）

- 通常的方法是直接设置srcnat的action=masquerade，但这样效率不高，特别在多线路接入时；
- 为了让nat地址转换效率更好，需要详细定义转换参数src-address=192.168.0.0/24和out-interface=ether2-tel，这样可以减少nat转换范围，使其转换更精确。



案例配置 – 流量控制

- 流量控制是网吧里最重要的部分，配置的好坏直接影响到网络速度的稳定

- 网吧早期的流量控制大多采用simple queue的简单队列流量控制，这样的方式是对单机进行固定限速，无法实现动态分配带宽

- 如果采用PCQ流量控制，能实现动态流量控制，配合queue tree还能实现游戏端口的优化

案例配置 – simple流控 （1）

- Simple队列一般只需要定义IP地址和上下行带宽，也可以设置单机突发带宽
- 添加方式，可以通脚本进行for循环添加所有的流量控制规则
- Simple流量控制缺点：

  不能动态流量限速和游戏优化；

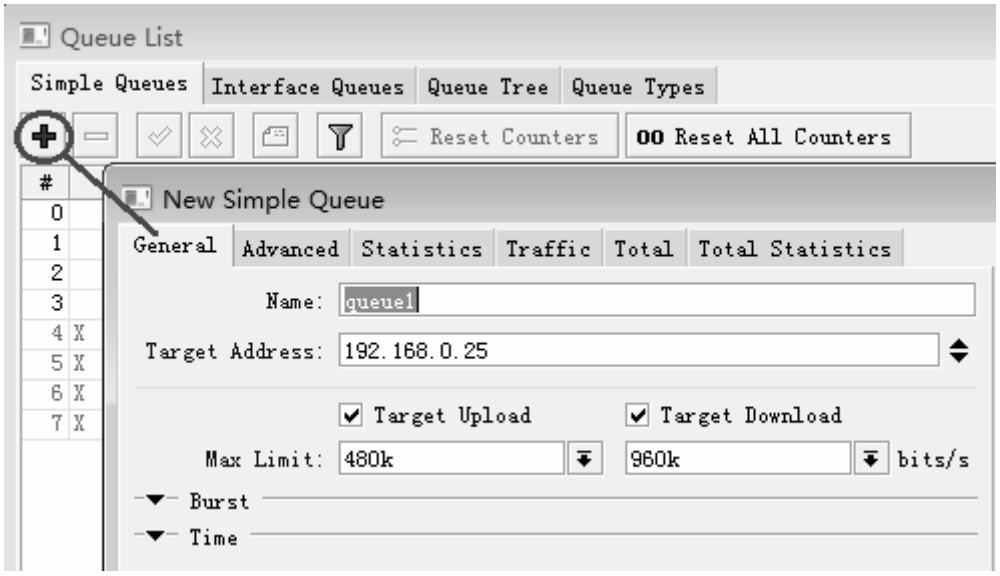
  规则越多，处理的数据越多，CPU消耗越大；

  规则越多，后面的规则获取带宽的几率越小；

  如果有1000条Simple queue规则，那必须判断 查询999条规则。

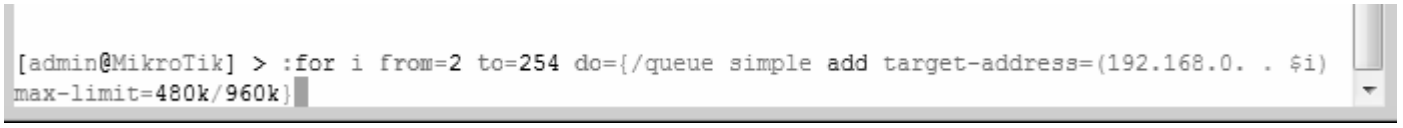
案例配置 – simple流控 （2）

- 简单队列的配置在queue simple中添加规则，下面是192.168.0.25的主机做流量控制。
- target-address设置主机的IP地址，Max-limit的upload是上行带宽480kbps，download是下行带宽为960Kbps



案例配置 – simple流控 （3）

- 通过脚本添加，使用脚本:for循环语句添加
- :for i from=2 to=254 do={/queue simple add target-address=(“192.168.0.” . $i) max-limit=480k/960k}
- :for i(定义变量i) from=2(从2开始循环) to=254(到254结束)
- do(执行以下指令)={/queue simple add(指定路径使用add添加命令)
- target-address=(“192.168.0.” . $i)(定义主机IP地址，用“.”连接“192.168.0.”和变量i)
- max-limit=480k/960k(定义主机带宽，带宽格式“上行/下行”)

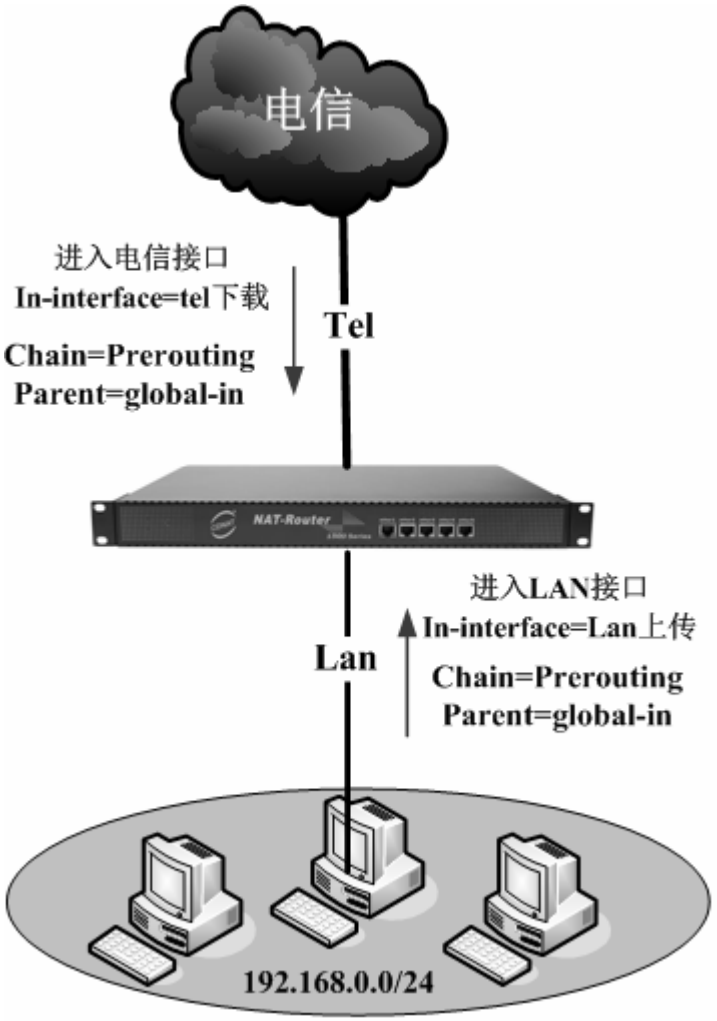


动态流量控制

- RouterOS可以实现对带宽的动态流量分配，如总带宽为12M的网络，可以将每台主机带宽放到2-4Mbps，通过PCQ算法对流量进行动态控制。

- 使用ip firewall mangle标记相应数据流，通过Queue tree对标记的数据流进行动态流量控制。

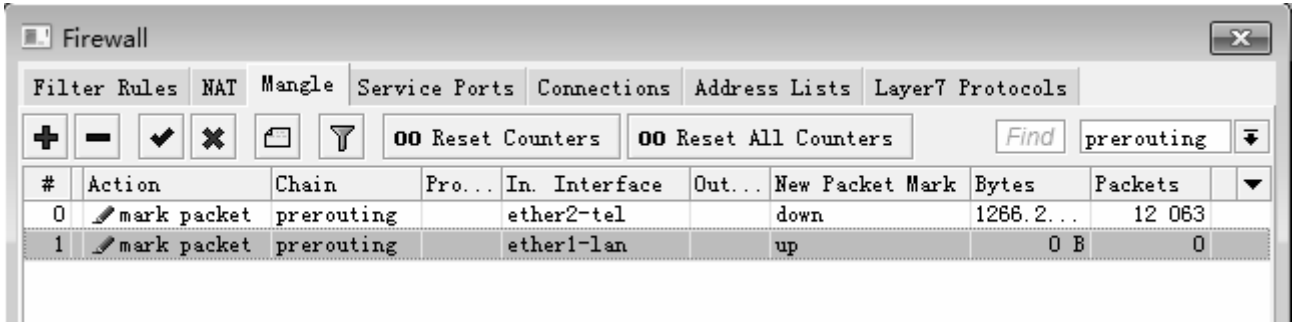
案例配置 – 动态流量控制



- PCQ是每次连接队列，即每次连接的流量都会被分类，根据当前连接数量的地址进行动态流量控制。
- 我们需要对Mangle规则了解，即进入路由器的数据流方向，这幅图的数据属于perouting链表
- 从tel口进入的是整个网络的下载
- 从lan口进入的是整个网络的上传
- 所有在prerouting链表中，我们标记的数据流都属于global-in

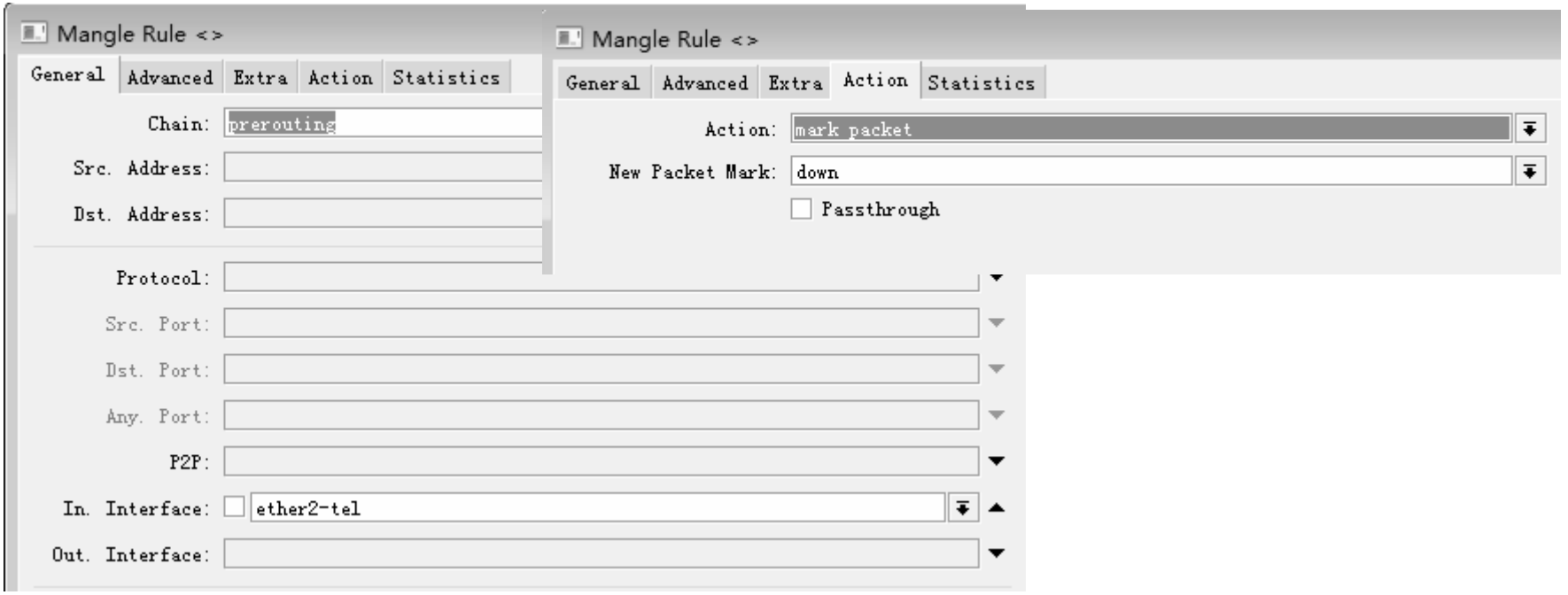
案例配置 – Mangle （1）

- 首先我们需要标记数据流，进入ip firewall mangle
- 选择chain=prerouting链表标记，分别标记从电信口进入的数据和从内网口进入的数据
- 下面是通过Mangle的prerouting链表抓取tel口和lan口数据：



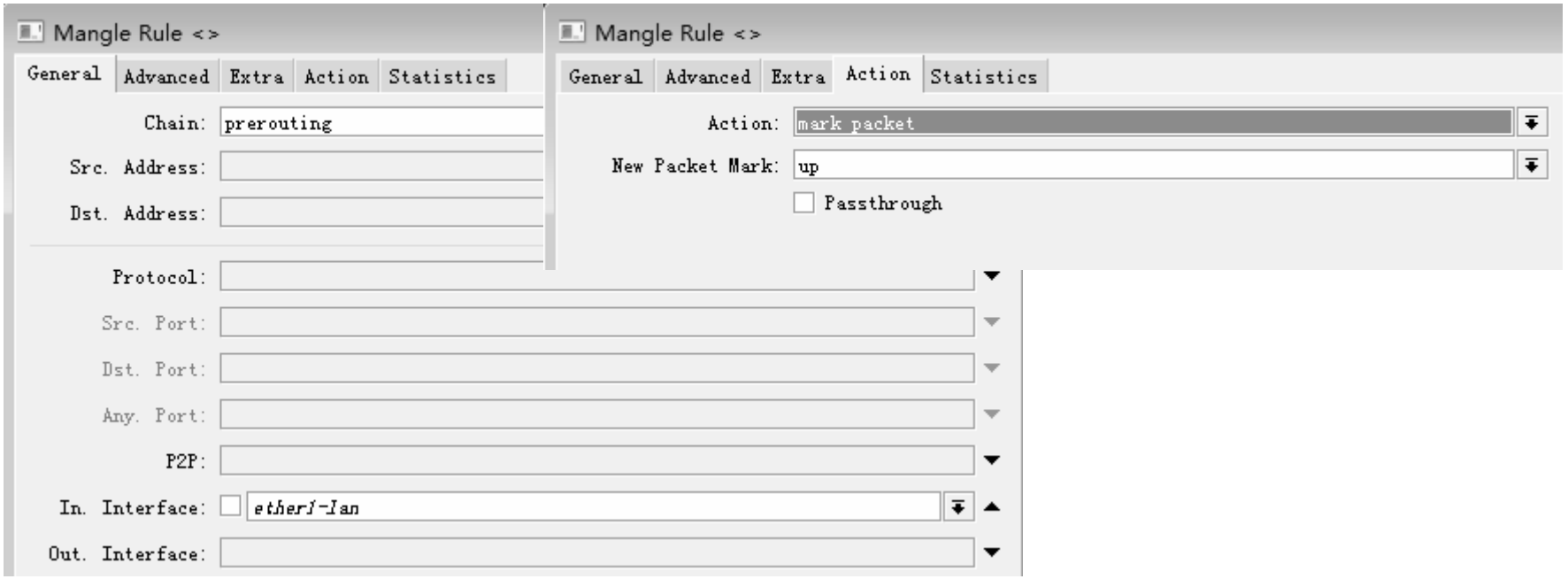
案例配置 – Mangle （2）

- 在winbox中标记电信接口，即网络的下载，标记in-interface=ether2-tel，passthrough=no



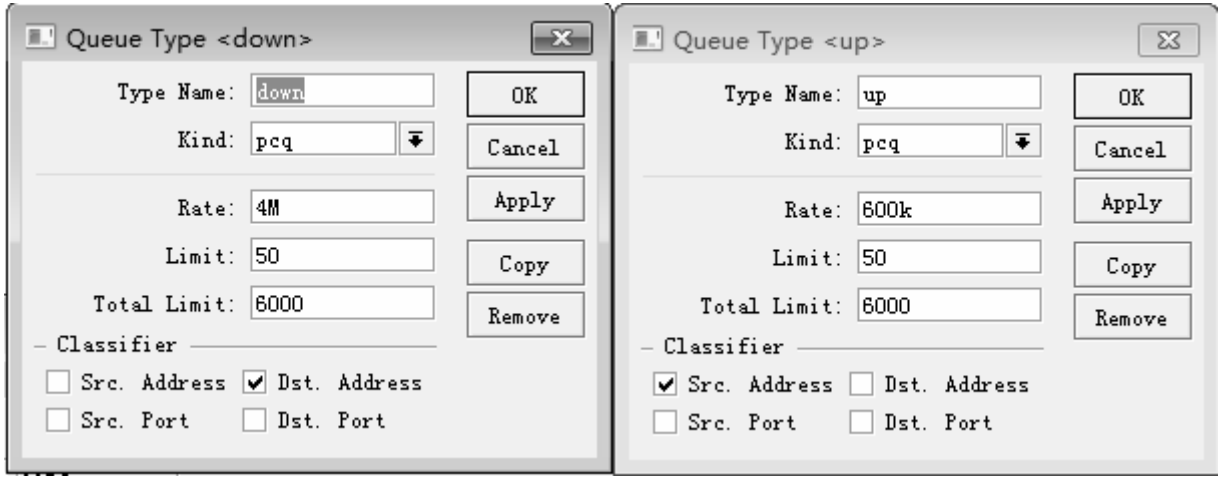
案例配置 – Mangle （3）

- 在winbox中标记电信接口，即网络的上传，标记in-interface=ether1-lan，passthrough=no



案例配置 – PCQ

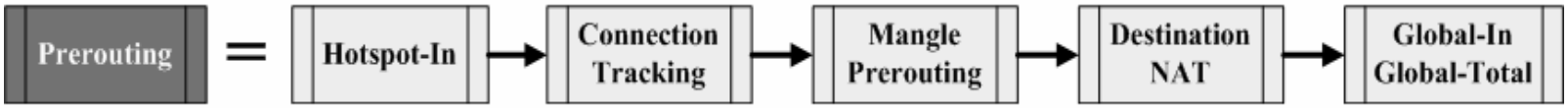
- 该案例我们的电信总带宽为12Mbps，主机在120台；
- 我们为每台主机下载分配总带宽的1/3，即down规则的pcq-rate=4M带宽；
- 通过计算total-limit=limit*120=6000；
- 下载通过dst-address分类，上传通过src-address分类。



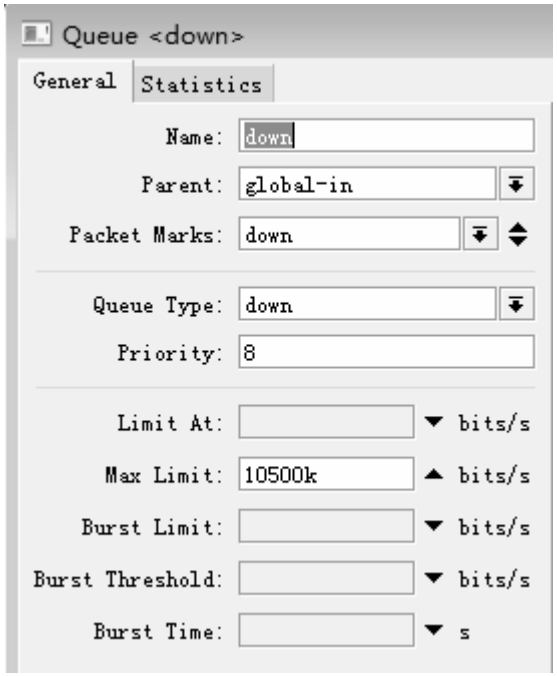
案例配置 – Queue tree（1）

- 之前配置了Mangle的流量标记和PCQ的规则，现在调用先前的配置，并通过queue tree实施流量控制。

- 由于我们标记的数据流都在prerouting链表里，所以在queue tree我们对上行和下载控制都通过global-in，因为在prerouting包含处理如下图：

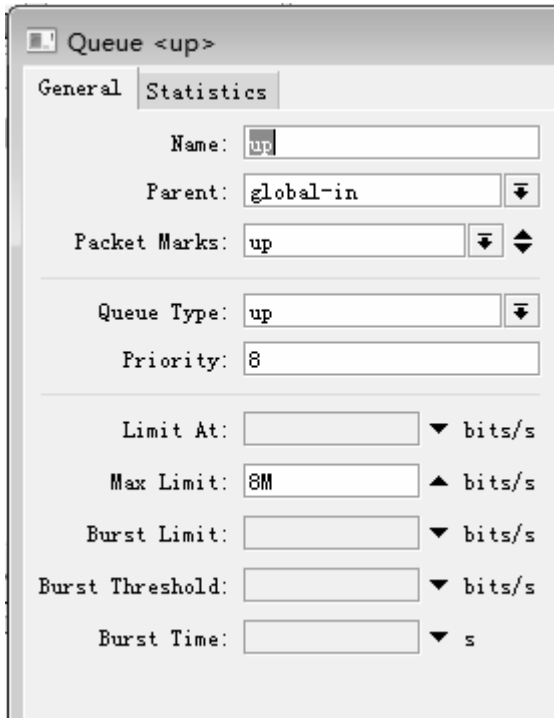


案例配置 – Queue tree（2）

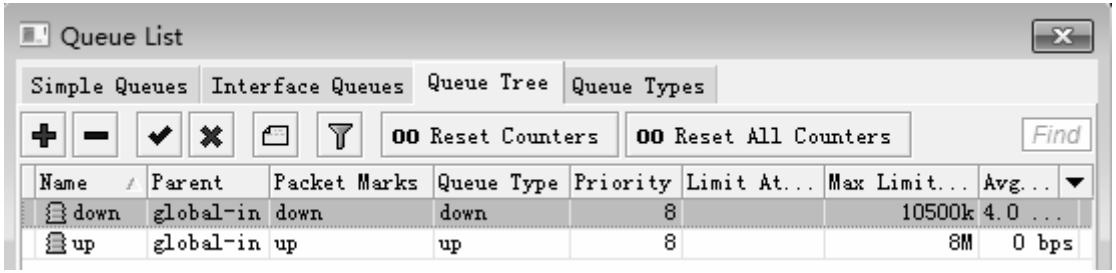


- 在queue tree添加规则，配置下行流量控制，取名为down
- 选择parent=global-in，packet-mark=down（之前mangle标记的下行数据），queu-type=down（在pcq定义的规则）
- 配置下行数据时注意总带宽为12M，需要保留部分缓冲带宽大概在1-2M左右，这里保留1.5M，在RouterOS不支持小数，需要用整数表示，即在max-limit=10500k

案例配置 – Queue tree（3）



- 在queue tree添加规则，配置上行流量控制，取名为up
- 选择parent=global-in，packet-mark=up（之前mangle标记的下行数据），queu-type=up（在pcq定义的规则）
- 配置上行数据一般小于实际带宽，通常情况下上行带宽较小，所以设置为8M，在max-limit=8M



案例配置 – 修改MAC地址

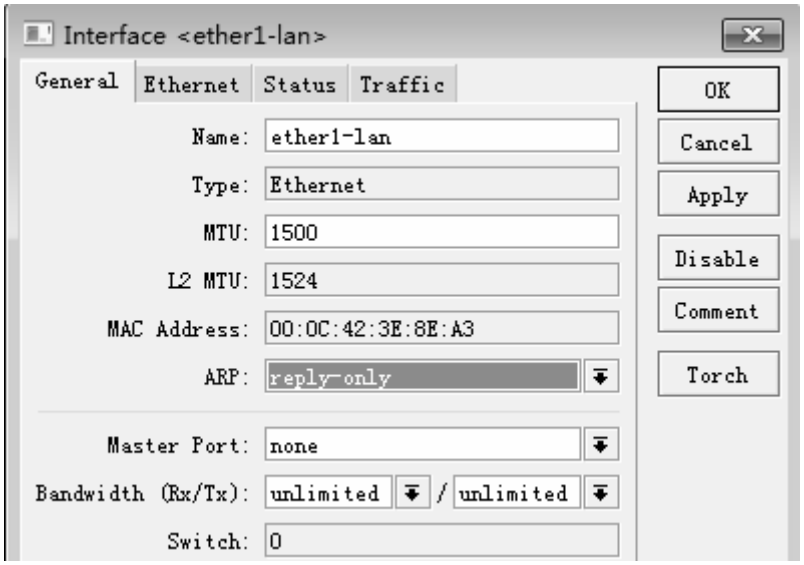
- 有时我们需要替换掉当前的其他型号路由器，但内网口MAC地址已被绑定，为了不让网络受绑定影响，我们直接修改内网口的MAC地址；

- 打开winbox的new terminal进入interface ethernet目录下，修改网卡MAC地址，操作如下：

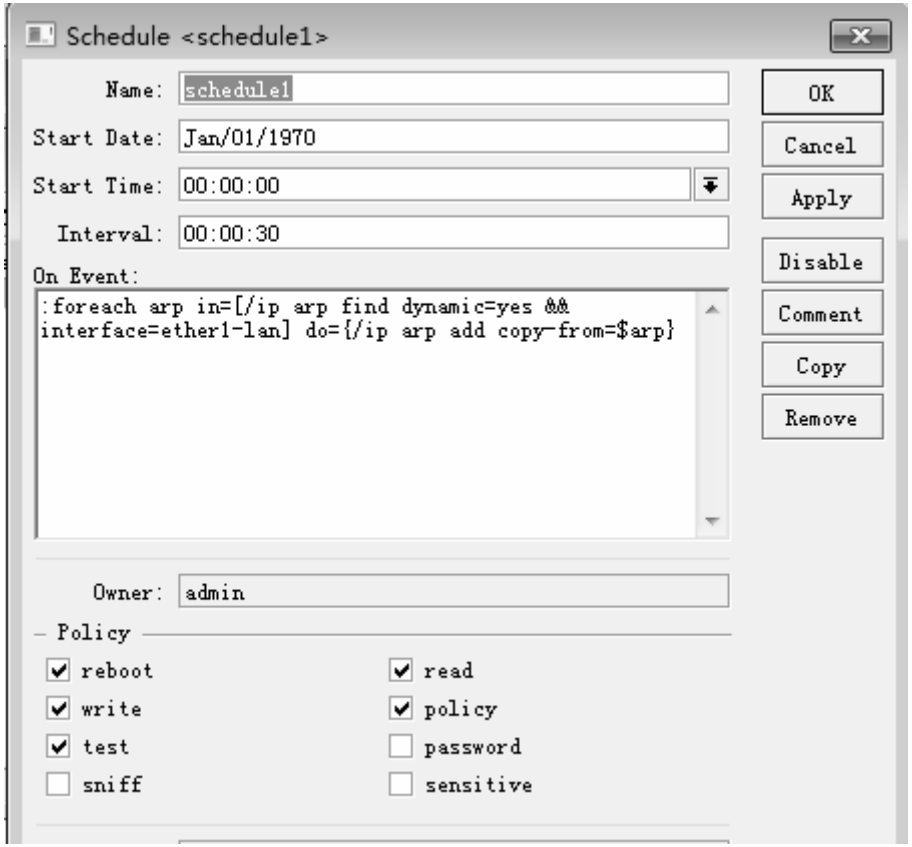
```
[admin@MikroTik] > interface ethernet
[admin@MikroTik] /interface ethernet> pri
Flags: X - disabled, R - running, S - slave
 #    NAME           MTU   MAC-ADDRESS       ARP        MASTER-PORT          SWITCH
 0    ether1-lan     1500  00:0C:42:56:C7:E4 enabled
 1 R  ether2-tel     1500  00:0C:42:56:C7:E5 enabled    none                 switch1
 2    ether3-cnc     1500  00:0C:42:56:C7:E6 enabled    none                 switch1
 3    ether4         1500  00:0C:42:56:C7:E7 enabled    none                 switch1
 4    ether5         1500  00:0C:42:56:C7:E8 enabled    none                 switch1
[admin@MikroTik] /interface ethernet> set 0 mac-address=00:01:bc:12:ad:b1
[admin@MikroTik] /interface ethernet>
```

案例配置 - MAC地址绑定（2）

- 进入interface下，选择ether1-lan的接口，将ARP属性选择为reply-only，仅回应ARP列表中静态的MAC地址，
- 注意在没有完全绑定所有机子的MAC前，请不要设置为reply-only，否则未绑定主机将无法获取MAC。



案例配置 - MAC地址绑定（3）



- 通常情况下在网吧安装路由器，网吧内的电脑不能全部开启，所以下面电脑的MAC地址不一定能全部获取，所以我们通过让RouterOS执行计划任务完成自动绑定网卡的MAC地址

- 进入system scheduler（计划任务）添加规则，设置脚本，并定义interval=30s（每间隔30秒执行）

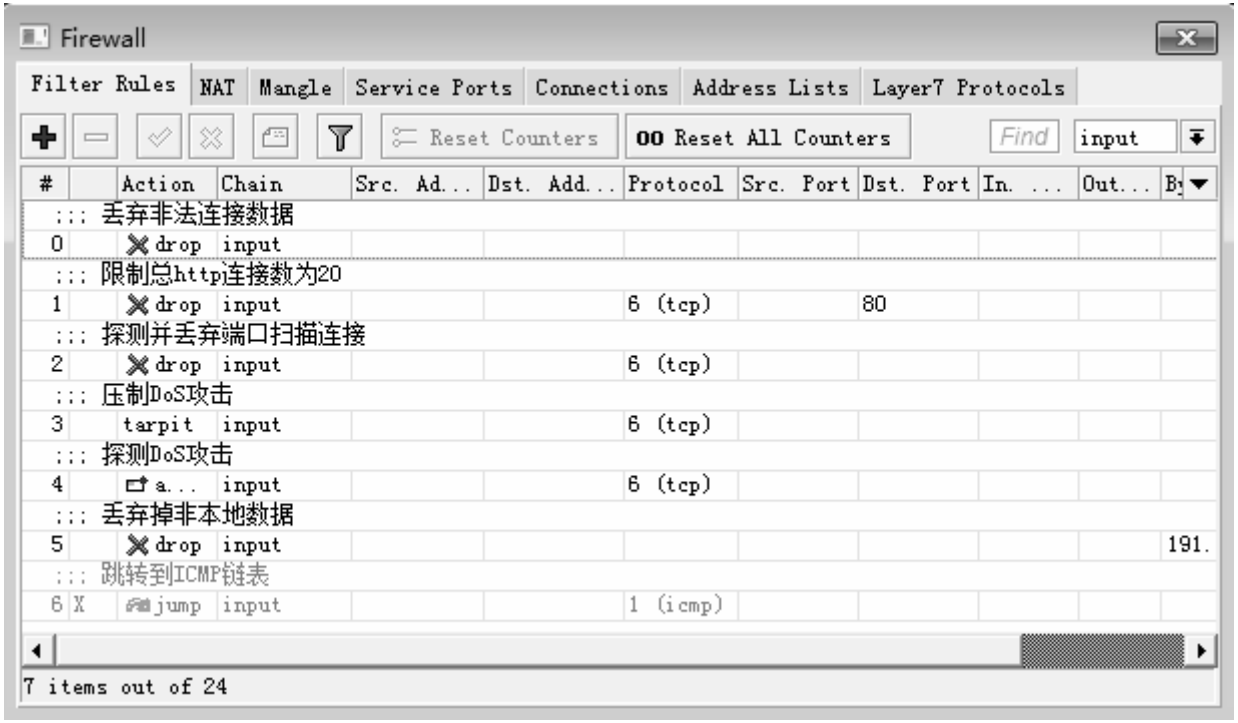
MAC绑定脚本

```
:foreach arp in=[/ip arp find dynamic=yes && interface=ether1-lan] do={/ip arp add copy-from=$arp}
```

案例配置 – 防火墙

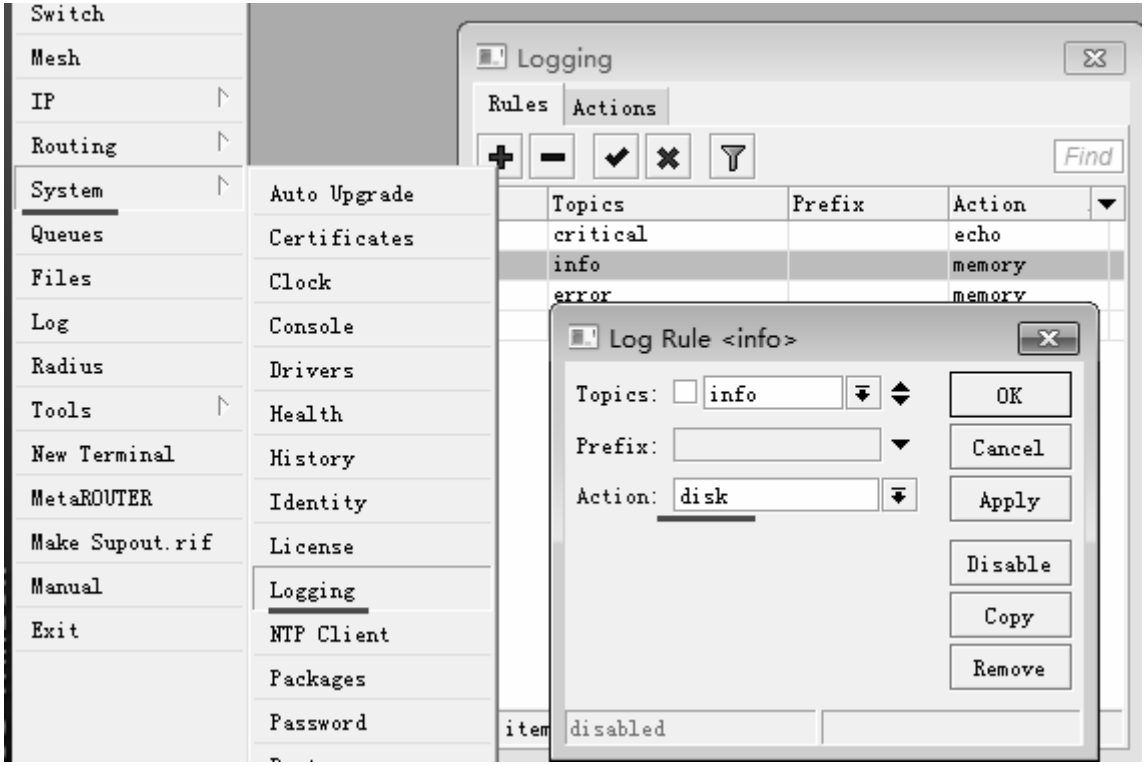
- 通过import导入防火墙脚本

```
[admin@MikroTik] > import 10firewall.rsc
Opening script file 10firewall.rsc

Script file loaded and executed successfully
[admin@MikroTik] >
```



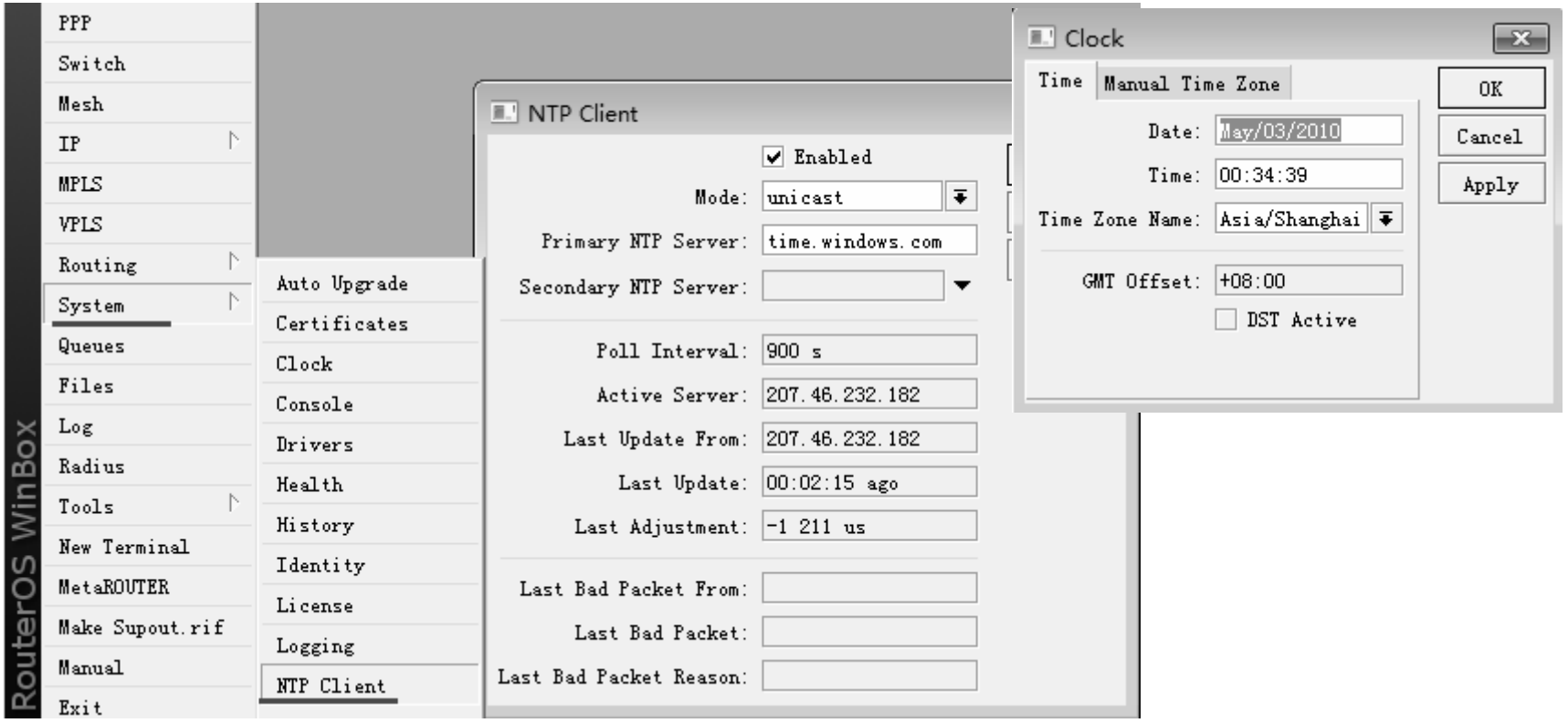
案例配置 – 系统日志记录

- 进入system logging将系统信息日志保持到路由闪存中，便于查询记录：



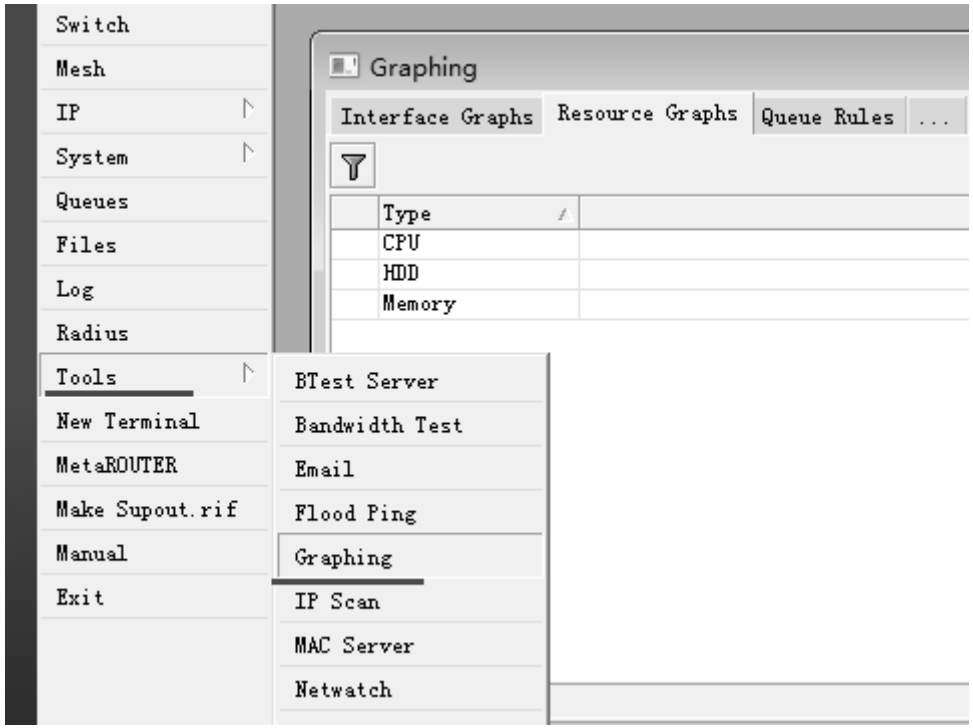
案例配置 – NTP网络对时

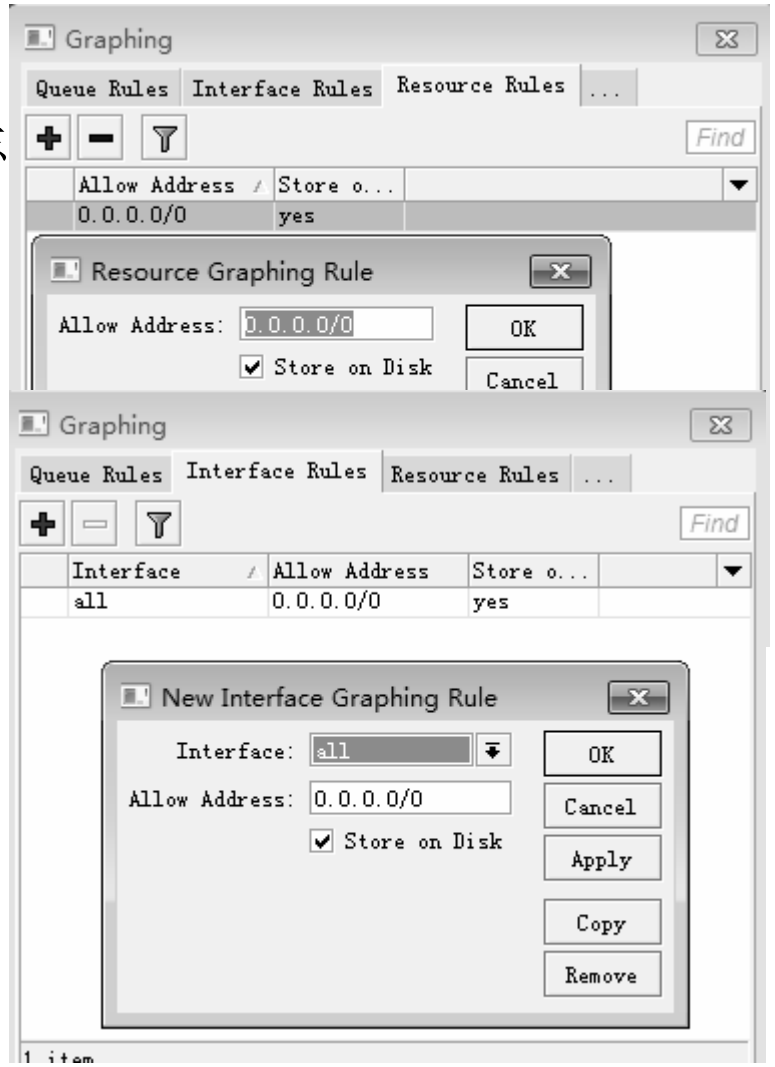
- 通在system NTP-client配置time.windows.com的NTP服务器，然后设置clock的时区为shanghai，通过网络对时保证系统时间正确，方便各种日志查询，配置如下图:



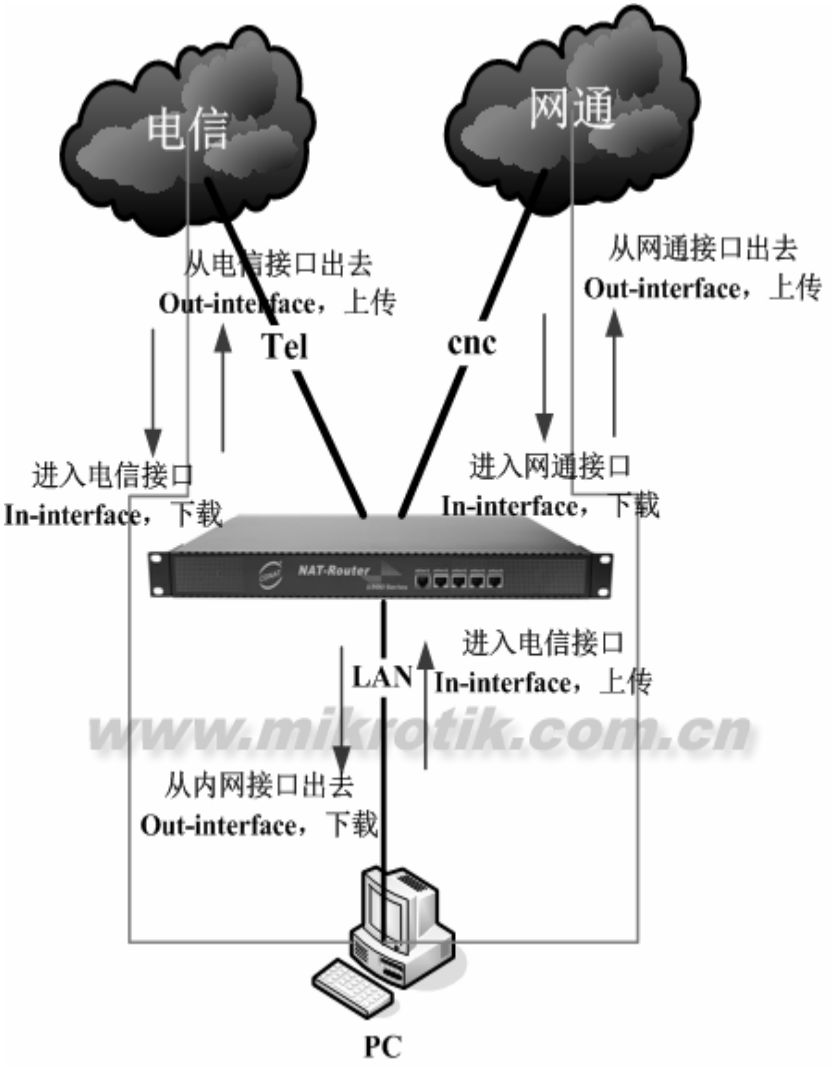
案例配置 – 流量图记录

在tools Graphing添加记录规则，记录流日志





双网卡流控标记特点



- 首先需要明白out-interface和in-interface对带宽控制的区别
- 对于外网接口的电信和网通(tel和cnc)，in-interface相对于路由器，标记的是下载数据，out-interface则是上传数据
- 但内网接口的lan相对于路由器则相反，in-interface是上传，out-interface是下载

PCQ动态流量实例

- 这里我们有一个实际环境，我们需要实现对带宽的动态分配；
- 电信带宽为6M，网通带宽为12M；
- 通过网通路由表实现双线路由，并将所有网页访问走网通线路。

- 配置步骤：

1、在ip firewall mangle标记上下行数据流

2、进入queue type定义单机带宽

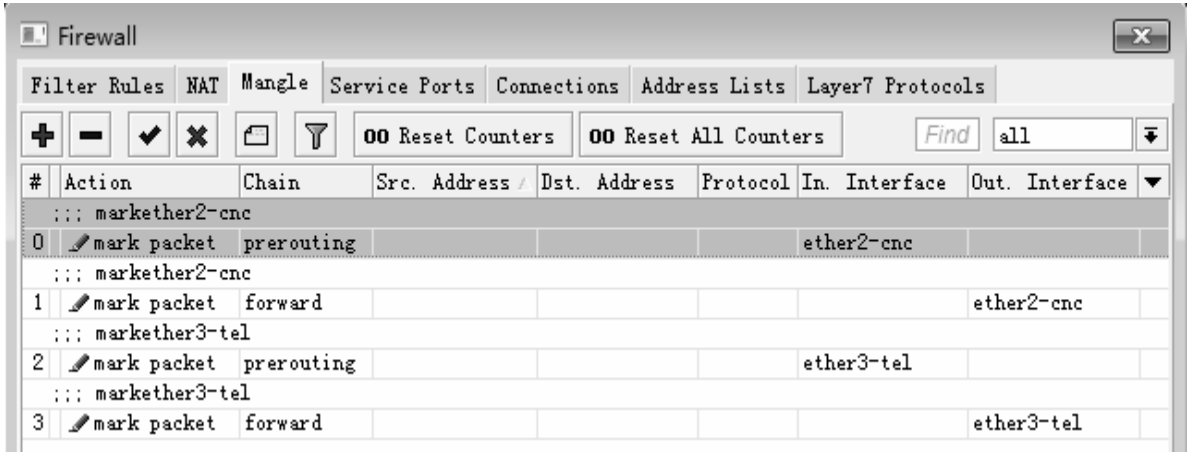
3、在queue tree定义总带宽和流量控制规则

Mangle数据包标记

/ip firewall mangle

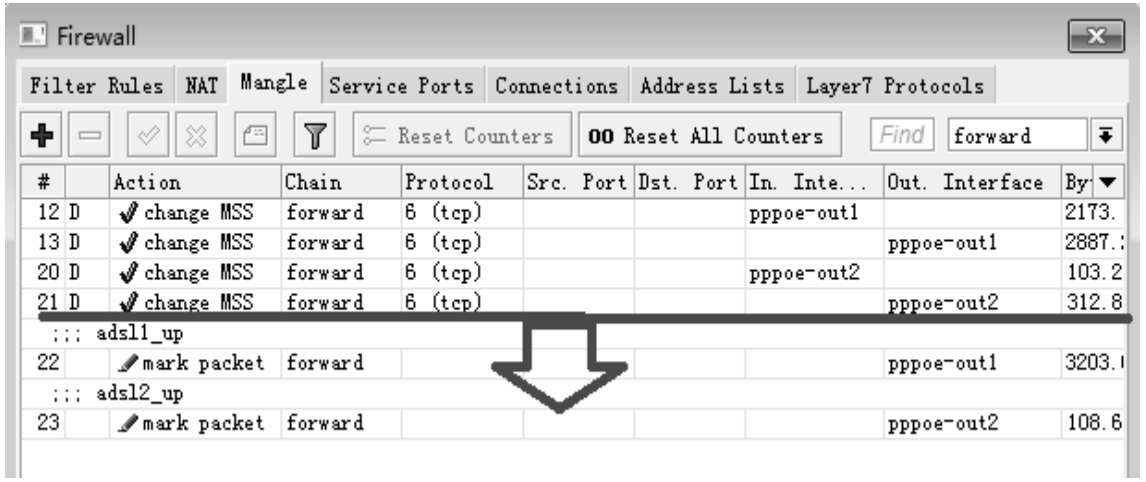
- add action=mark-packet chain=prerouting comment=markether2-cnc in-interface=ether2-cnc new-packet-mark=ether2-cnc_down passthrough=no
- add action=mark-packet chain=forward comment=markether2-cnc new-packet-mark=ether2-cnc_up out-interface=ether2-cnc passthrough=no

- add action=mark-packet chain=prerouting comment=markether3-tel in-interface=ether3-tel new-packet-mark=ether3-tel_down passthrough=no
- add action=mark-packet chain=forward comment=markether3-tel new-packet-mark=ether3-tel_up out-interface=ether3-tel passthrough=no



Mangle标记要点

- 注意：如果是ADSL，我们需要将上行数据的标记放到ADSL的change MSS的规则之后，否则会出现部分网站无法打开的问题。



Queue type设置

- 在Queue type中定义PCQ规则，为电信和网通分别定义各自的上下行带宽
- 电信下载最大1M，网通最大2M

Queue tree

我们在定义是需要考虑预留带宽，即电信为6M带宽，我们在给定电信总带宽max-limit为4800k(约4.8M)，网通设置为10M

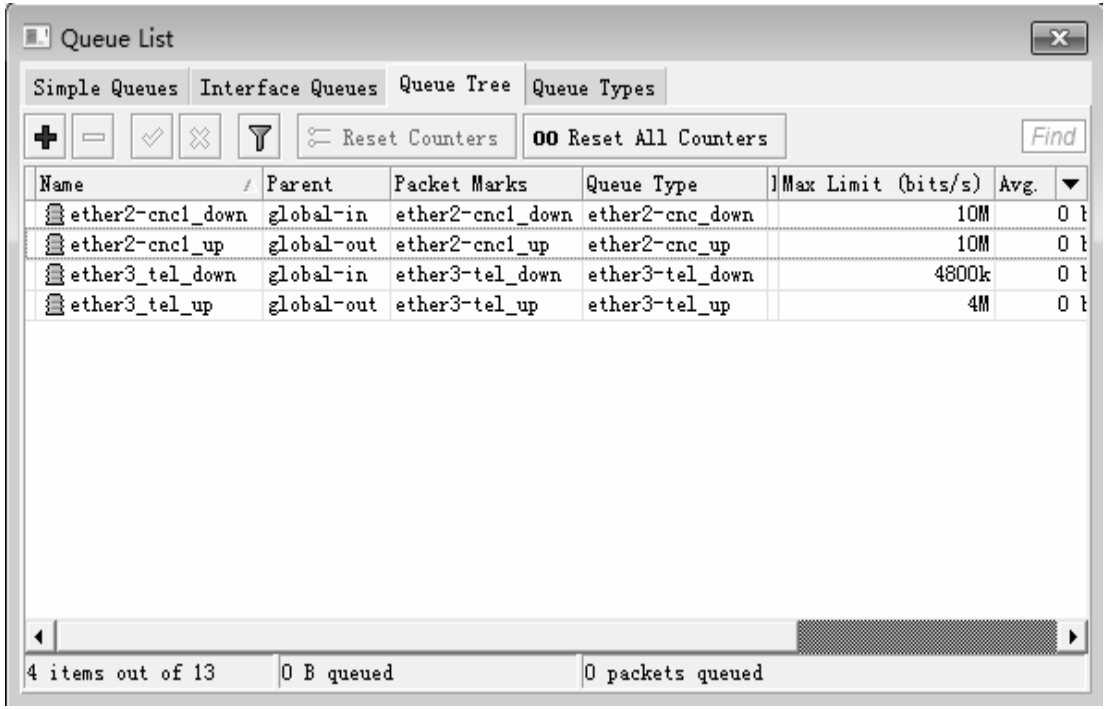
/queue tree

- add max-limit=4800k name=ether3_tel_down packet-mark=ether3-tel_down parent=global-in queue=ether3-tel_down
- add max-limit=4M name=ether3_tel_up packet-mark=ether3-tel_up parent=global-out queue=ether3-tel_up

- add max-limit=10M name=ether2-cnc1_down packet-mark=ether2-cnc1_down parent=global-in queue=ether2-cnc_down
- add max-limit=10M name=ether2-cnc1_up packet-mark=ether2-cnc1_up parent=global-out queue=ether2-cnc_up

Winbox配置

- 在queue tree中定义电信和网通线路的带宽

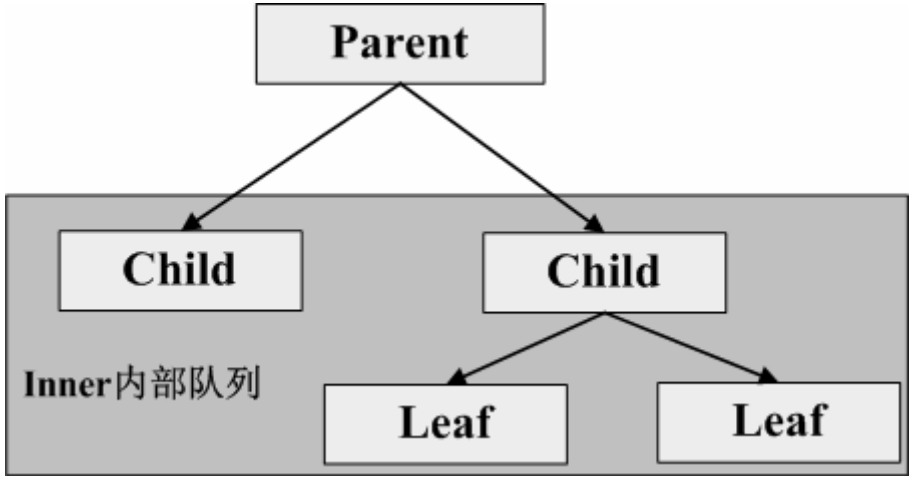


Queue Tree

- Queue Tree是唯一能直接被HTB所调用
- Queue Tree没有像Simple Queue是从上往下有序执行，他没有次序。
- 所有子队列都要从”/ip firewall mangle”中提取标记数据
- 当Simple Queue处理与Queue Tree相同数据时，那Simple Queue会优先执行。

案例配置 - HTB

- HTB等级令牌桶是创建一个等级队列结构，并确定队列之间的关系，就像“父亲与儿子”或“兄弟之间”
- 一旦队列添加了一个**Child**（子队列）将会变为**inner(**内部队列**)**，所有向下没有**Children**（子队列）称为**Leaf**队列（叶队列）。
- HTB是必须指定**Parent(**父级**)**选项，并指定一个从属的队列为子队列 。



案例配置 - 双重限制

每个队列在**HTB**有**2**个带宽限制：

- **CIR** (约定信息速率Committed Information Rate) – (在RouterOS中的参数为**limit-at**) 最坏的情况下，无论如何每个队列都会得到CIR带宽

- **MIR** (最大信息速率Maximal Information Rate) – (在RouterOS中的参数为**max-limit**) 最好的情况下，如果父级有剩余带宽，才能获得这部分剩下的带宽 。

- 换句话说，首先Limit-at（CIR）都会被满足，仅当子队列尝试借调父级剩余带宽时，才可以达到最大的带宽 **max-limit** (**MIR**).

案例配置 - CIR与MIR

- 在HTB中，无论如何**CIR** 带宽都将会得到满足 (即使父级的max-limit满载)，那就是为什么，确保最佳的使用双重限制功能，我们建议坚持这些规则：

- **CIR**约定带宽之和，即所有子级带宽必须小于或等于可获得父级带宽：

  CIR(parent)∑ ⩾ CIR(child1) +...+ CIR(childN)

- 子父级与主父级允许设置为： **CIR(parent)=MIR(parent)**

- 任何子级的最大带宽必须小于或者等于父级的最大带宽

案例配置 - Priority 优先级

这里我们知道，所有队列的**limit-at (CIR)**都有可能被耗尽，优先级则主要负责分配父级队列剩余的带宽给**Child**（子队列）达到max-limit。队列高的优先级最优先达到max-limit，优先级低的则不会。8是最低优先级，1则最高。

注意，优先级工作环境:

- 对于leaf叶队列的优先级对于自己inner（内部队列）没有任何意义，即inner内部队列与其所属的leaf（叶队列）的优先级是不可比较。

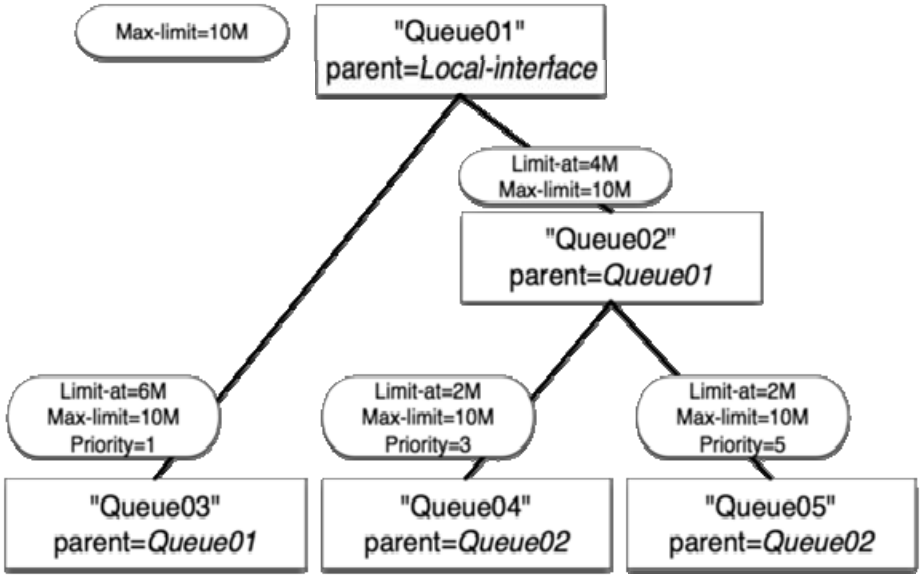
案例配置 - HTB事例

下面这部分我们将分析HTB的操作，将演示一个HTB结构并将涵盖可能出现的所有情况和功能，我们的HTB结构由下面5个队列构成：

- **Queue01** 内部队列有2个子级 - **Queue02**和**Queue03**
- **Queue02** 内部队列有2个子级 - **Queue04**和**Queue05**
- **Queue03** 叶队列
- **Queue04** 叶队列
- **Queue05** 叶队列

Queue03，**Queue04**和**Queue05**分别需要**10Mbps**，我们总出口为**10Mbps**的带宽

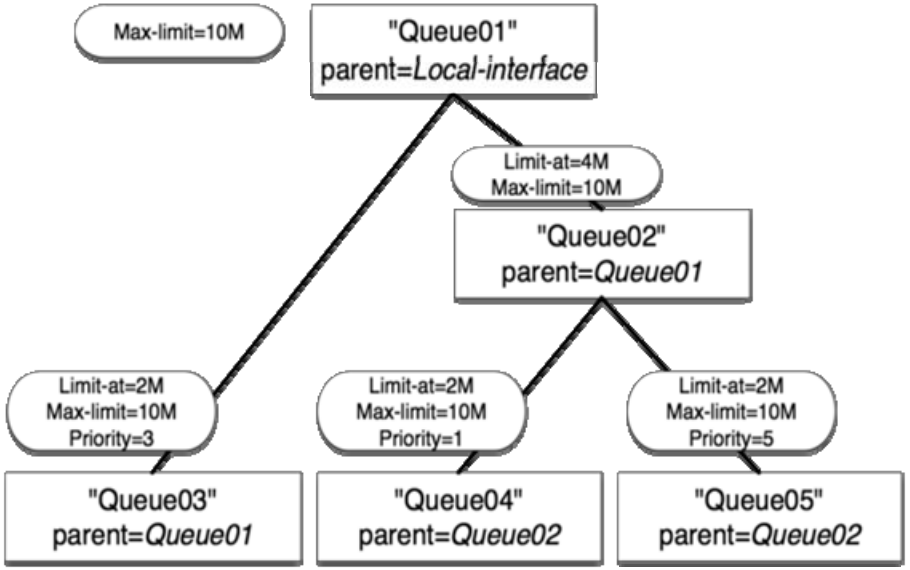
普通事例



Queue01 limit-at=0Mbps max-limit=10Mbps
Queue02 limit-at=4Mbps max-limit=10Mbps
Queue03 limit-at=6Mbps max-limit=10Mbps priority=1
Queue04 limit-at=2Mbps max-limit=10Mbps priority=3
Queue05 limit-at=2Mbps max-limit=10Mbps priority=5

结论：
Queue03 得到6Mbps
Queue04 得到2Mbps
Queue05 得到2Mbps
HTB通过满足所有的**limit-at**，父级队列已没有带宽进行分发。

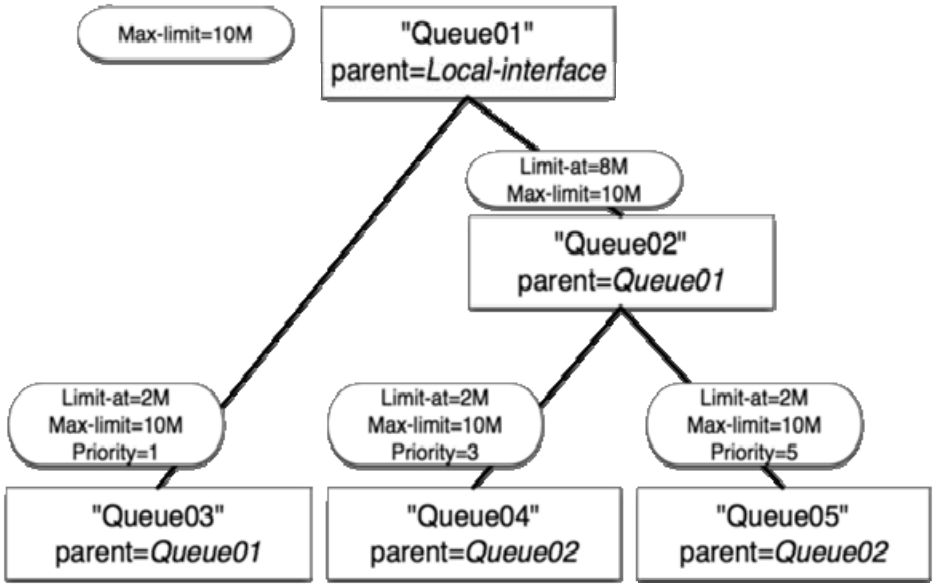
max-limit事例



Queue01 limit-at=0Mbps max-limit=10Mbps
Queue02 limit-at=4Mbps max-limit=10Mbps
Queue03 limit-at=2Mbps max-limit=10Mbps priority=3
Queue04 limit-at=2Mbps max-limit=10Mbps priority=1
Queue05 limit-at=2Mbps max-limit=10Mbps priority=5

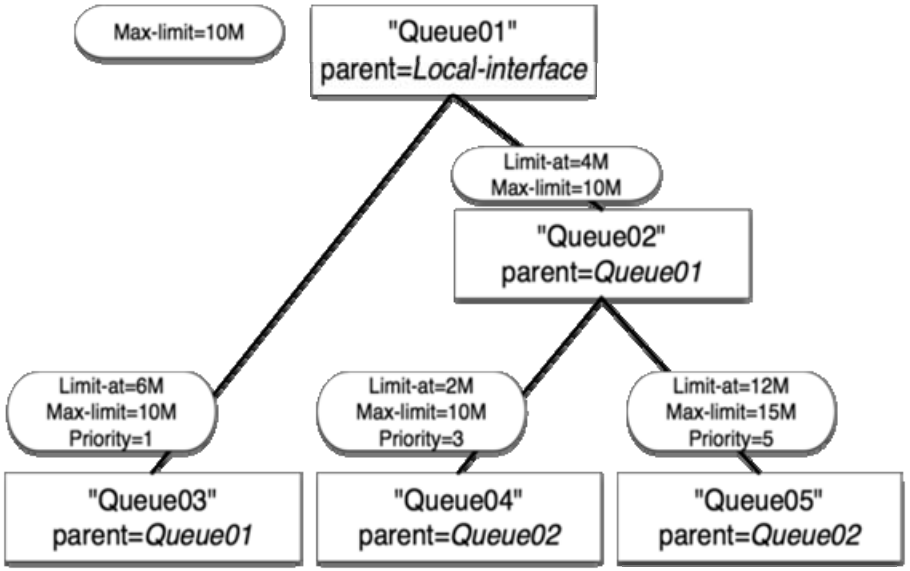
结论：
Queue03 得到 2Mbps
Queue04 得到 6Mbps
Queue05 得到 2Mbps
在满足所有的**limit-at**后，**HTB**将把剩余的带宽分配给优先级高的队列。

inner队列limit-at



Queue01 limit-at=0Mbps max-limit=10Mbps
Queue02 limit-at=8Mbps max-limit=10Mbps
Queue03 limit-at=2Mbps max-limit=10Mbps priority=1
Queue04 limit-at=2Mbps max-limit=10Mbps priority=3
Queue05 limit-at=2Mbps max-limit=10Mbps priority=5
结论：
Queue03 得到2Mbps
Queue04 得到6Mbps
Queue05 得到2Mbps
在满足所有的**limit-at**后，**HTB**将分配剩余带宽给优先级高的，但在这个事例中，内部对列**Queue02**指定了**Limit-at**， 这样他会保留**8Mbps**的流量给**Queue04**和**Queue05**，**Queue04**有更高的优先级，那就是为什么会得到更高的带宽。

leaf队列的Limit-at

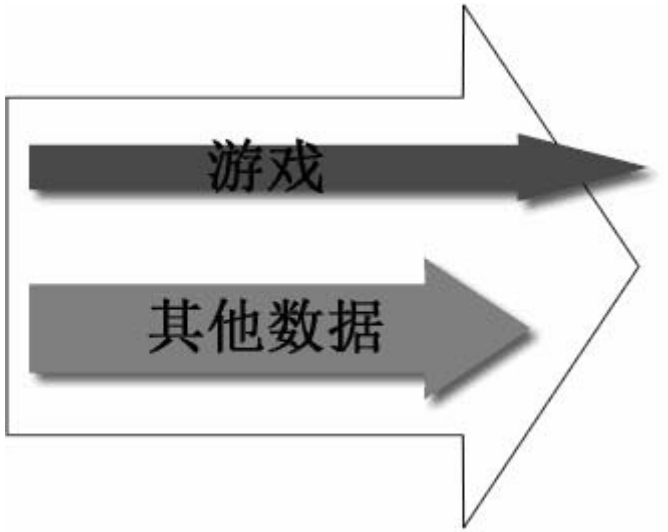


Queue01 limit-at=0Mbps max-limit=10Mbps
Queue02 limit-at=4Mbps max-limit=10Mbps
Queue03 limit-at=6Mbps max-limit=10Mbps priority=1
Queue04 limit-at=2Mbps max-limit=10Mbps priority=3
Queue05 limit-at=12Mbps max-limit=15Mbps priority=5
结论：
Queue03 得到3Mbps
Queue04 得到1Mbps
Queue05 得到6Mbps
为了满足所有的**Limit-at**，**HTB**被强迫分配**20Mbps**，**Queue03**为**6Mbps**，**Queue04**为**2Mbps**，**Queue05**为**12Mbps**，但我们的接口只能处理**10Mbps**，因此接口队列通常**FIFO**带宽发分配将保持比例 **6:2:12**，即**3:1:6**。

游戏优先配置

- PCQ完成了动态的流控控制，已经达到对带宽的充分利用，但如果一台主机在看视频和下载的时候，通常认为是不能玩游戏的，这时我们可以通过设置游戏优先让这台主机下载游戏同时进行，与动态流控组合使其更完美。

- 将采用Queue Tree的HTB令牌桶流量控制，进行优先级排序。



案例配置 - 游戏优先 （1）

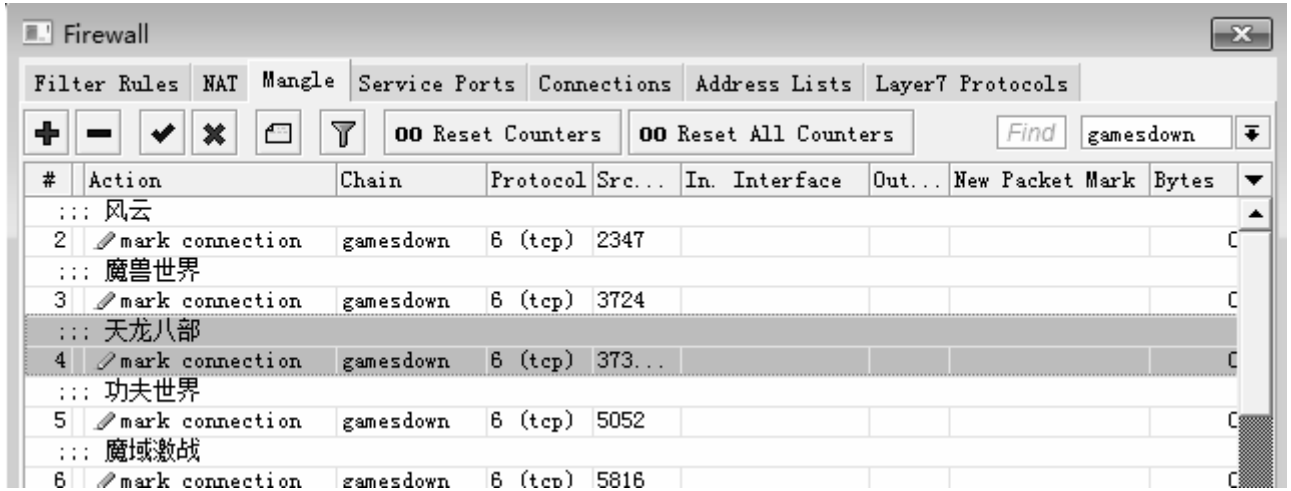
- 在理解了HTB的原理后，我们接下了就要对之前的动态流量控制进行改进
- 在ip firewall mangle加入游戏端口列表的数据标记
- 在queue tree增加对下行数据的HTB和游戏队列规则

案例配置 - 游戏优先 （2）

- 通过，import命令导入游戏列表，脚本都可以在ip firewall mangle中找到gamesdown的链表；

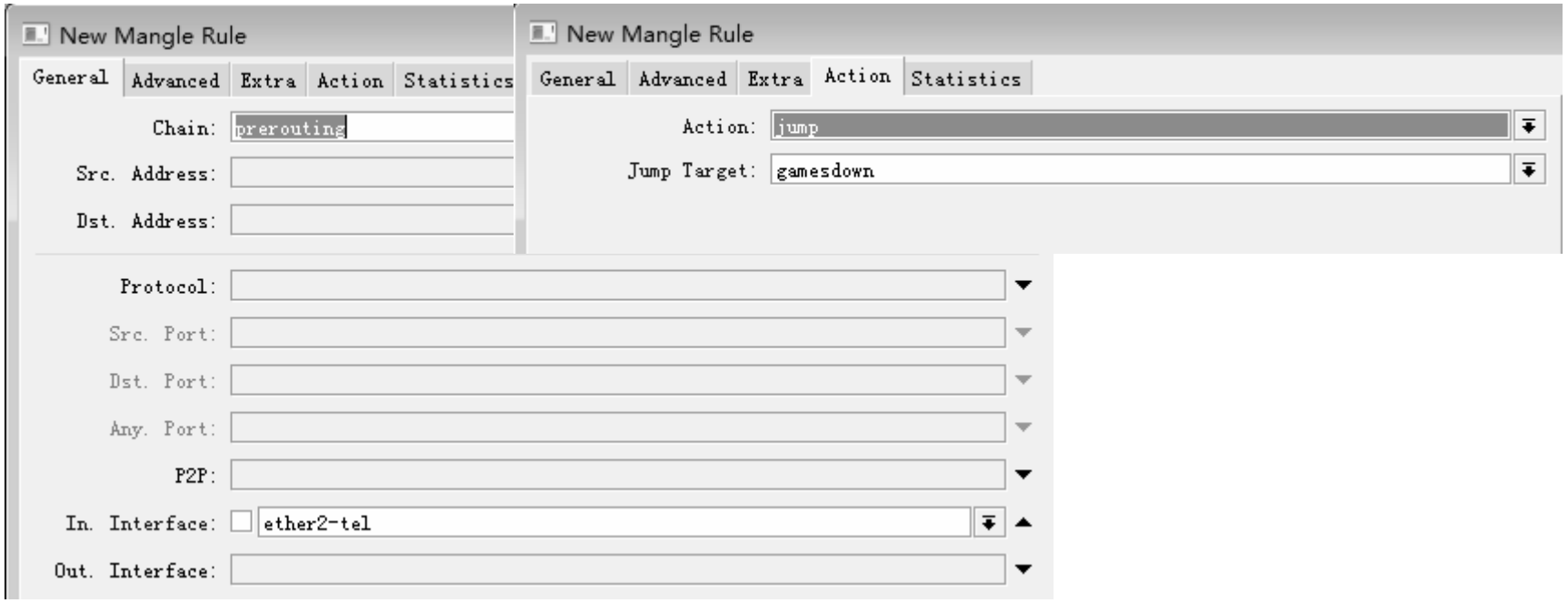
```
[admin@MikroTik] > import games320.rsc
Opening script file games320.rsc

Script file loaded and executed successfully
[admin@MikroTik] >
```

- 在ip firewall mangle显示gamesdown链表



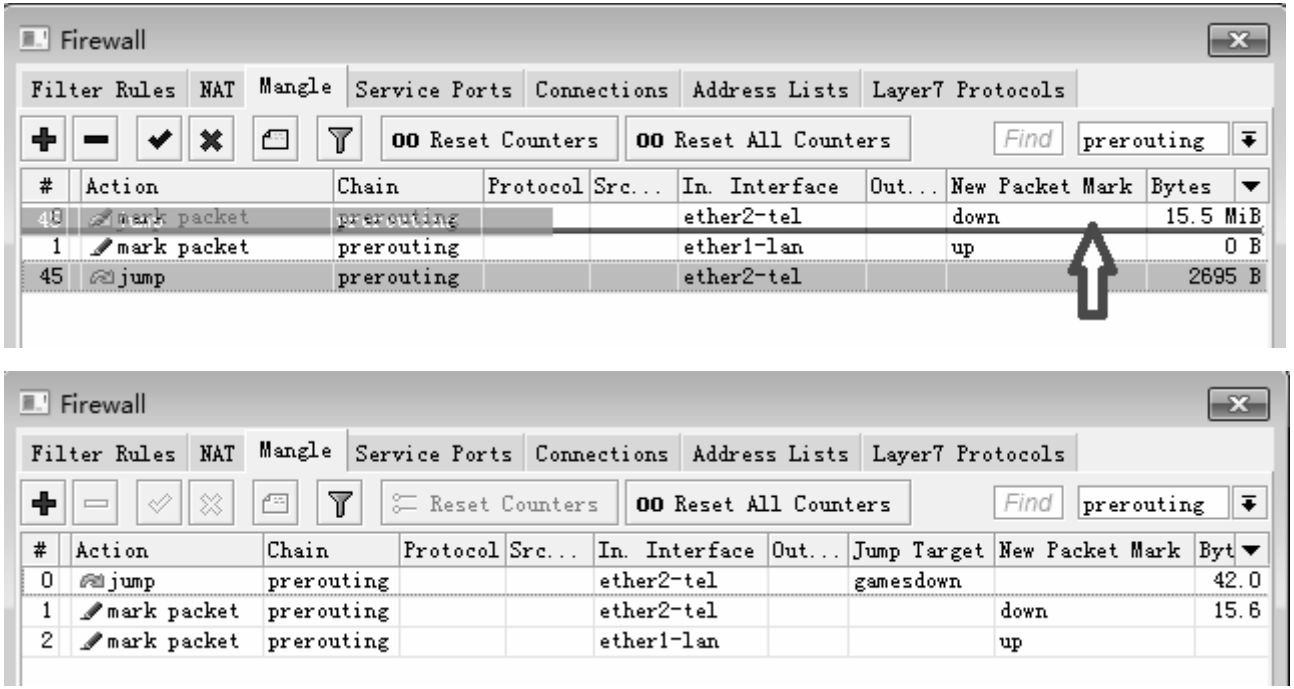
案例配置 - 游戏优先 （3）

- 我们仍然在prerouting里配置规则，将从电信in-interface=ether2-tel进入的数据跳转到gamesdown链表中进行处理，设置action=jump，jump-target=gamesdown

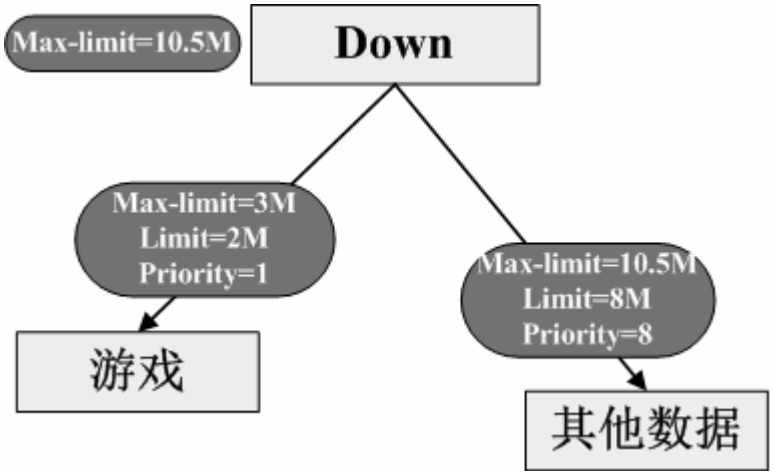


案例配置 - 游戏优先 （4）

- 将定义好的游戏跳转规则放到电信下行标记之前，通过鼠标拖动规则，移动到下行down规则之前，是游戏在prerouting链表里需要最先处理



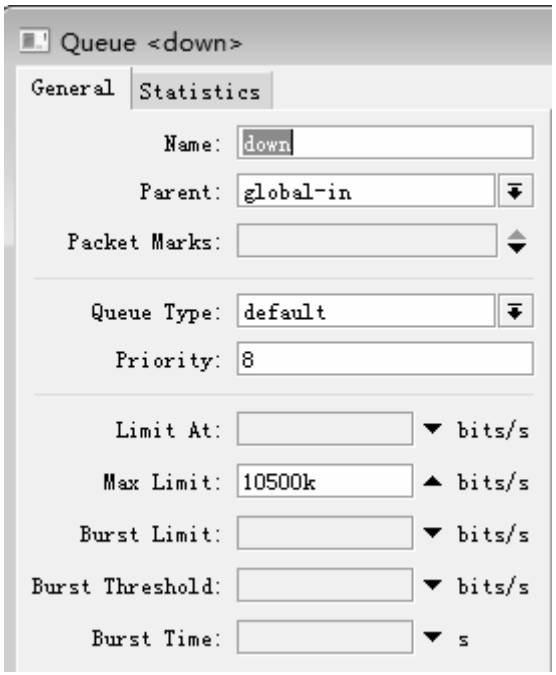
案例配置 - 游戏优先 （5）



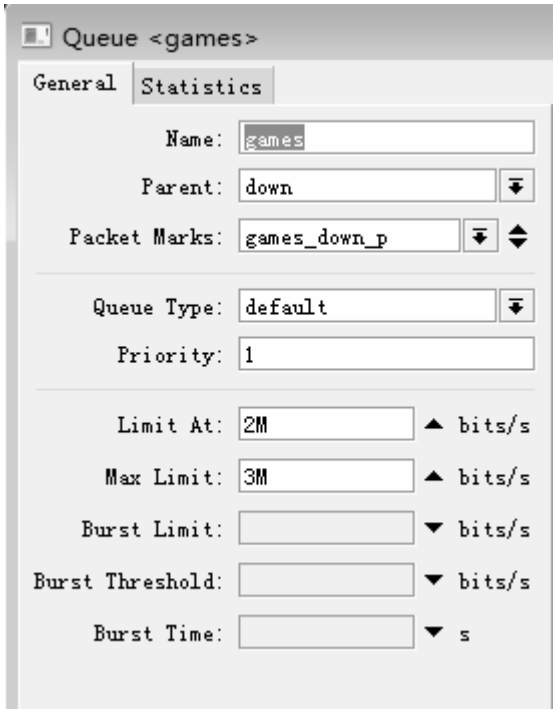
- 在mangle标记完成后，我们接下来将在Queue tree里跳转现有的流控规则为HTB控制
- 将原有的下行控制down规则，修改为父级队列，在父级队列下增加两个子队列others和games
- 在大多网络中带宽消耗来至于下行，所以不不考虑对上行的游戏带宽处理，主要对下行的数据进行优化
- 请再次观察左边的流量分配图

案例配置 - 游戏优先 （6）

- 进入Queue tree，修改down为父级队列，取消之前的packet-market和修改Queue-type=default

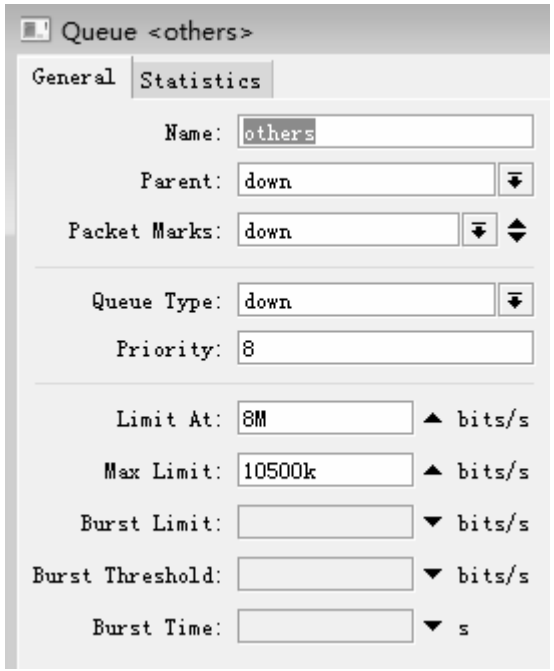


案例配置 - 游戏优先 （7）



- 添加游戏子队列，定义名称为games，设置父级从属于down，即parent=down；
- 由于游戏不需要太多带宽，一般100台的网吧最多耗用1M游戏带宽，我们设置queue-type=default，即单机在游戏端口下不限制任何带宽；
- 这里最大只分配了3M的max-limit，最低保证2M的limit-at
- 优先级priority=1，优先级最高

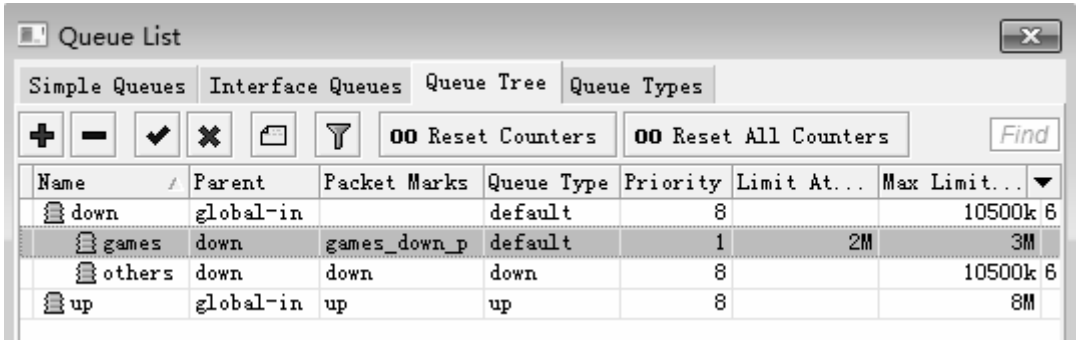
案例配置 - 游戏优先 （8）



- 添加除游戏外的其他数据，取名为others；
- 之前我们已经在游戏队列中分配了2M的limit-at，总带宽我们设置的10.5M，还剩下8.5M；
- 根据HTB原则父级带宽必须大于或者等于子队列带宽之和，即我们应该给其他数据的limit-at分配8.5M带宽，但是为了考虑带宽灵活分配，我们预留了500k出来，作为灵活分配，给其他数据分配8M带宽
- 优先级priority=8，优先级最低

案例配置 - 游戏优先 （9）

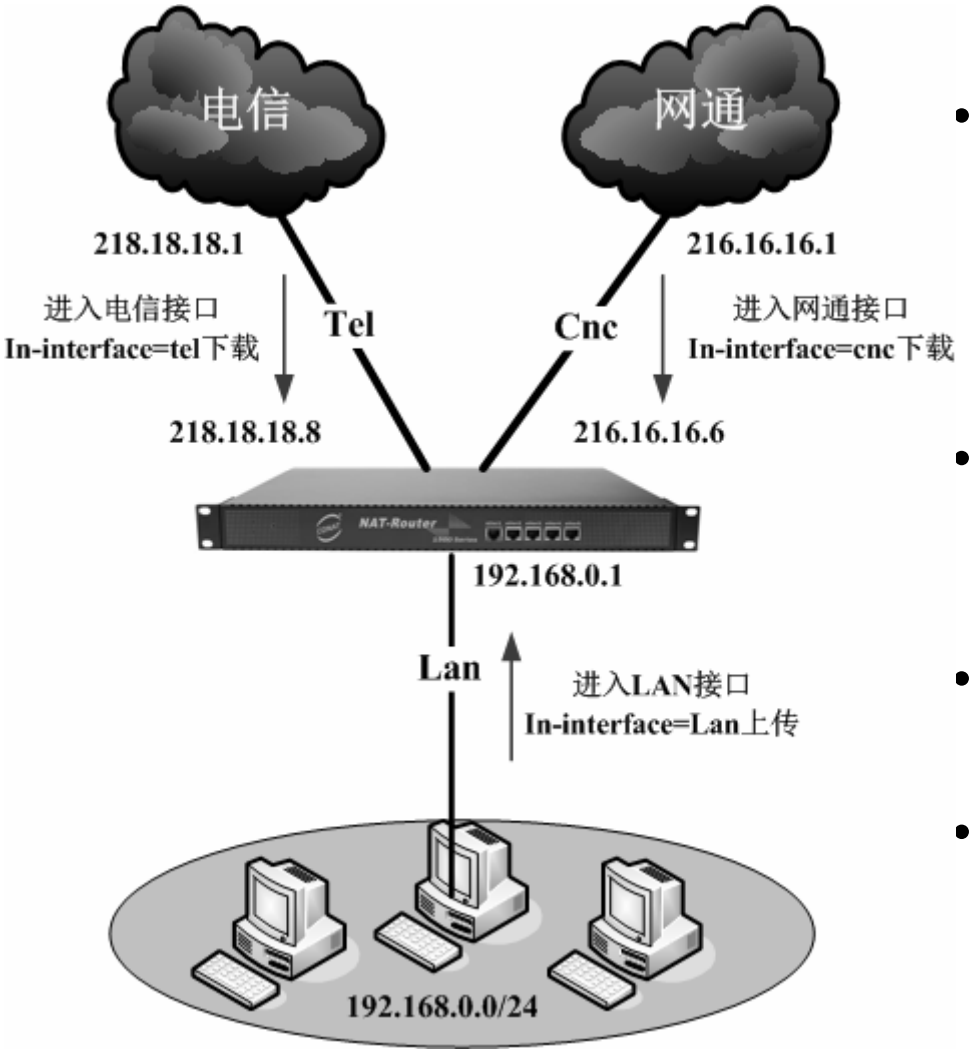
- 游戏优先的HTB配置完成后，我们观察Queue tree的情况，games和others从属于down父级队列，他们都从父级中获取10.5M带宽，只是games队列的优先级高于others：



电信网通双线案例

- 网吧应用最多的为电信与网通的双线案例
- 这个案例里，我们需要做的是导入电信或者网通的路由表，根据不同线路做路由策略
- 一般情况下网通带宽高于电信，所以用网通走默认路由，电信走策略路由表
- 之后对电信和网通做流量控制和游戏优先的处理

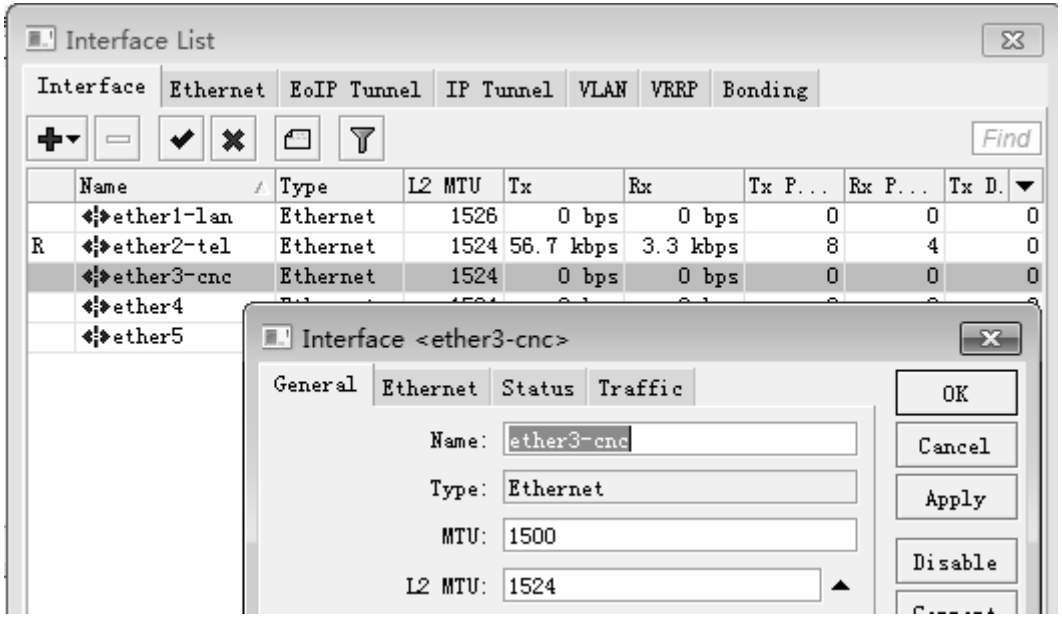
案例分析



- 电信和网通双线的案例在之前单线案例基础上增加了网通线路，我们在配置上只需要增加一些配置
- 增加网通IP地址216.16.16.6，网关216.16.16.1
- 需要导入电信和网通路由表
- 流量控制增加网通流量控制规则

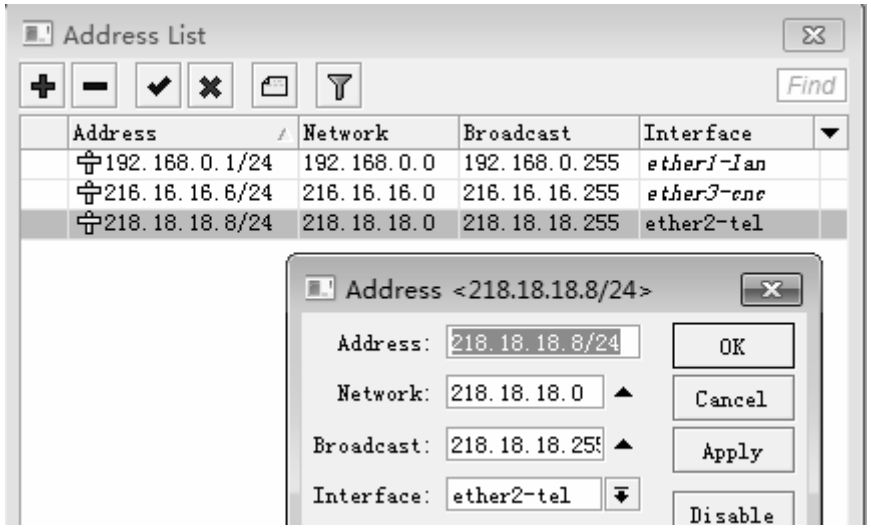
案例配置 – IP地址 （1）

- 在interface里修改ether3接口，取名ether3-cnc



案例配置 – IP地址 （2）

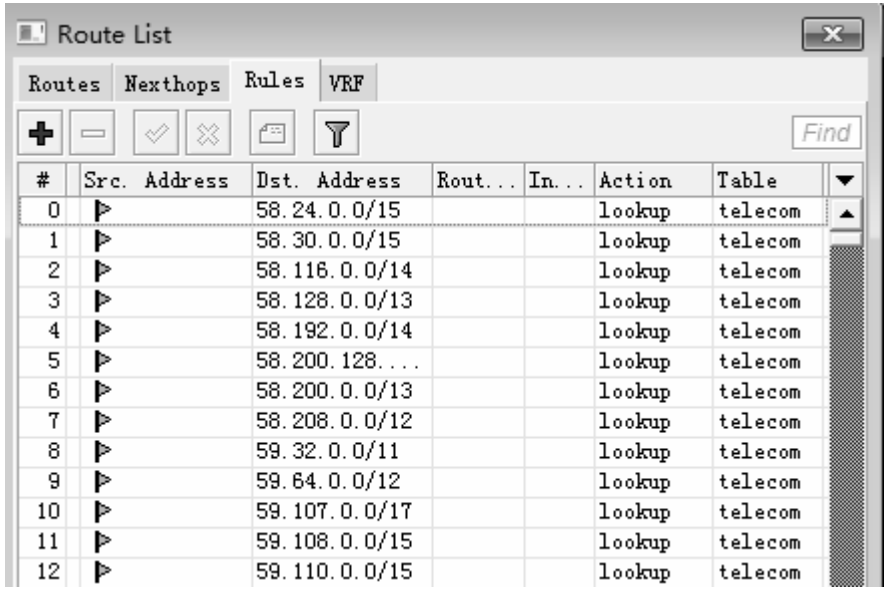
- 在原有的单线网络上增加网通的IP
- 进入ip address添加网通的IP地址216.16.16.6/24



案例配置 – 双线路由（1）

- 首先我们需要导入电信或者网通的路由表，在导入前我们将分析用什么线路做默认路由，什么线路做路由表
- 通常情况下网通带宽高于电信，所以我们选择网通走默认路由，导入电信路由表
- 上传路由表到files目录下，用import命令导入，导入后我们可以在ip route rules里看到电信路由表规则

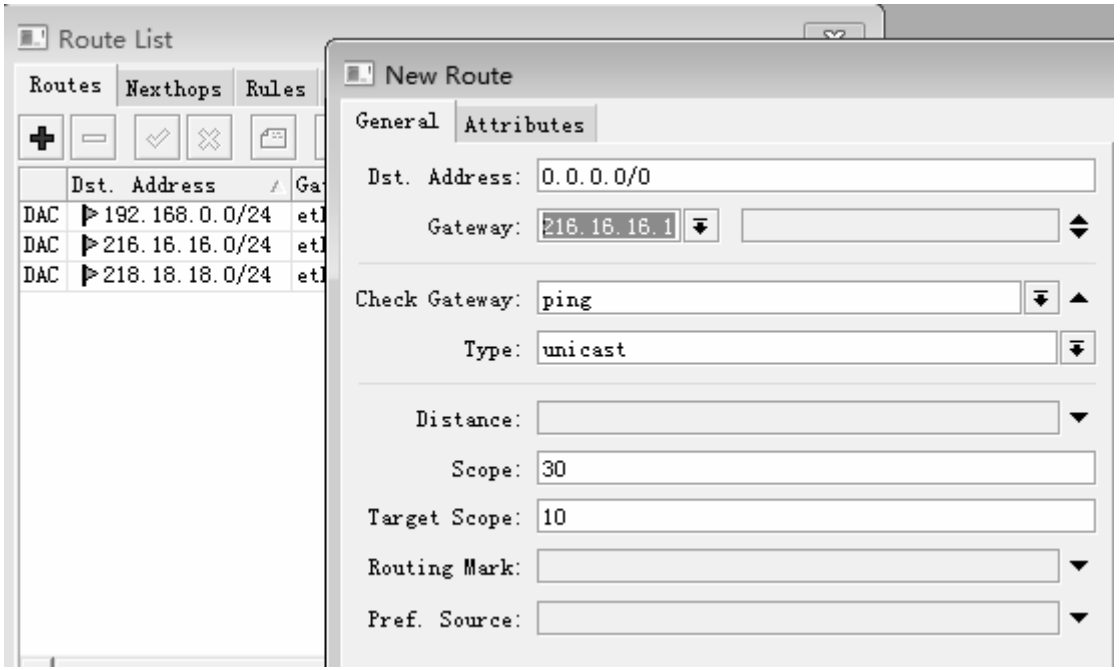
```
[admin@MikroTik] > import tel10.rsc
Opening script file tel10.rsc

Script file loaded and executed successfully
[admin@MikroTik] >
```



#	Src. Address	Dst. Address	Rout...	In...	Action	Table
0		58.24.0.0/15			lookup	telecom
1		58.30.0.0/15			lookup	telecom
2		58.116.0.0/14			lookup	telecom
3		58.128.0.0/13			lookup	telecom
4		58.192.0.0/14			lookup	telecom
5		58.200.128....			lookup	telecom
6		58.200.0.0/13			lookup	telecom
7		58.208.0.0/12			lookup	telecom
8		59.32.0.0/11			lookup	telecom
9		59.64.0.0/12			lookup	telecom
10		59.107.0.0/17			lookup	telecom
11		59.108.0.0/15			lookup	telecom
12		59.110.0.0/15			lookup	telecom

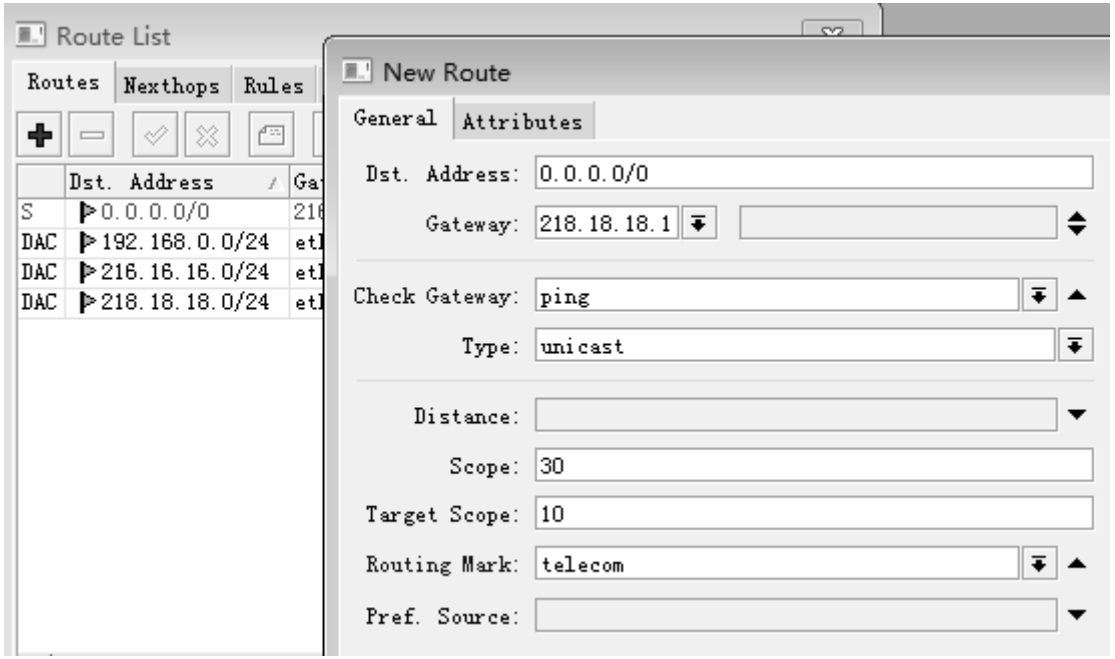
案例配置 – 双线路由（2）

- 进入ip route修改网通为默认路由，添加电信路由标记，



案例配置 – 双线路由（3）

- 添加电信网关和调用路由表，设置routing-mark=telecom，指向在ip route rules中的电信路由表



案例配置 – 双线备份（1）

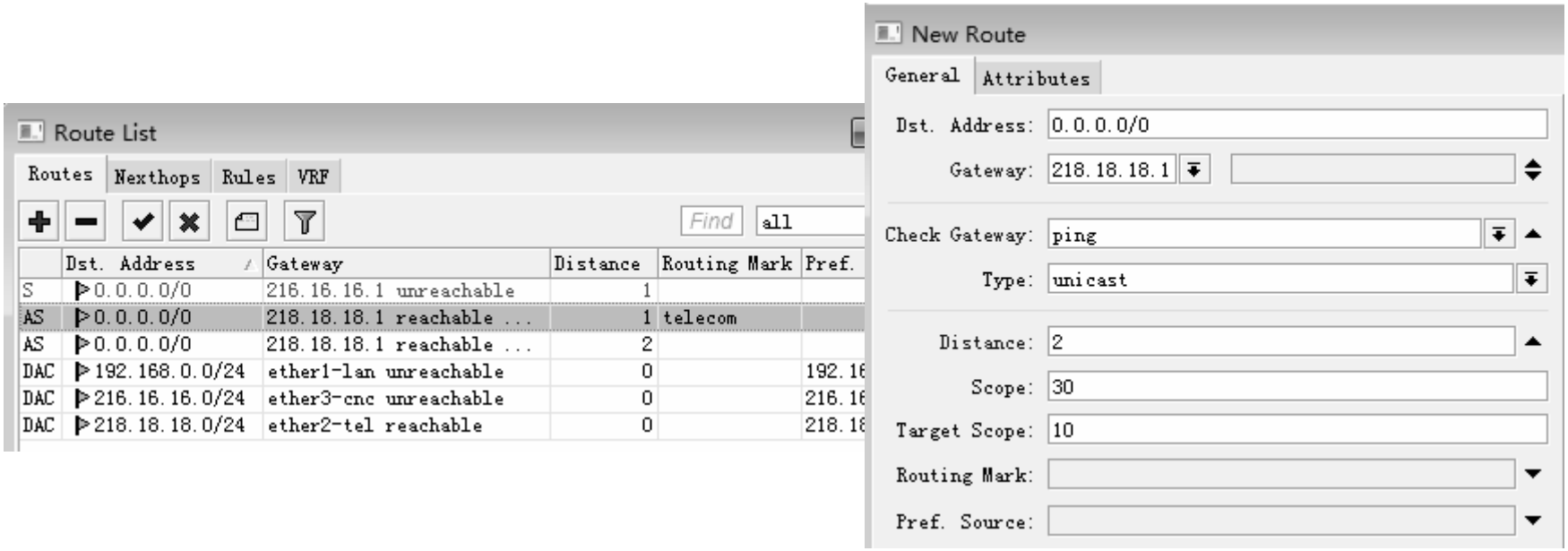
- 在双线事例中，可能出现某条线路出现故障一时无法恢复，那就需要使用另外一条线路进行备份

- 之前我们在每条路由网关设置时都会设置check-gateway=ping的参数，这里是监测网络运行情况，如果网关无法ping通表示路由失效，即该规则将会变为无效，RouterOS将不再执行该规则

- 在该事例中，网通为默认路由，那电信将做为备份，启用备份路由，我们只需增加一条电信网关的默认路由规则，只是将distance设置为2，默认的distance=1表示距离为1跳，设置为2表示2跳，这样默认的网通网关是1距离最近优先考虑，电信为2做备份。

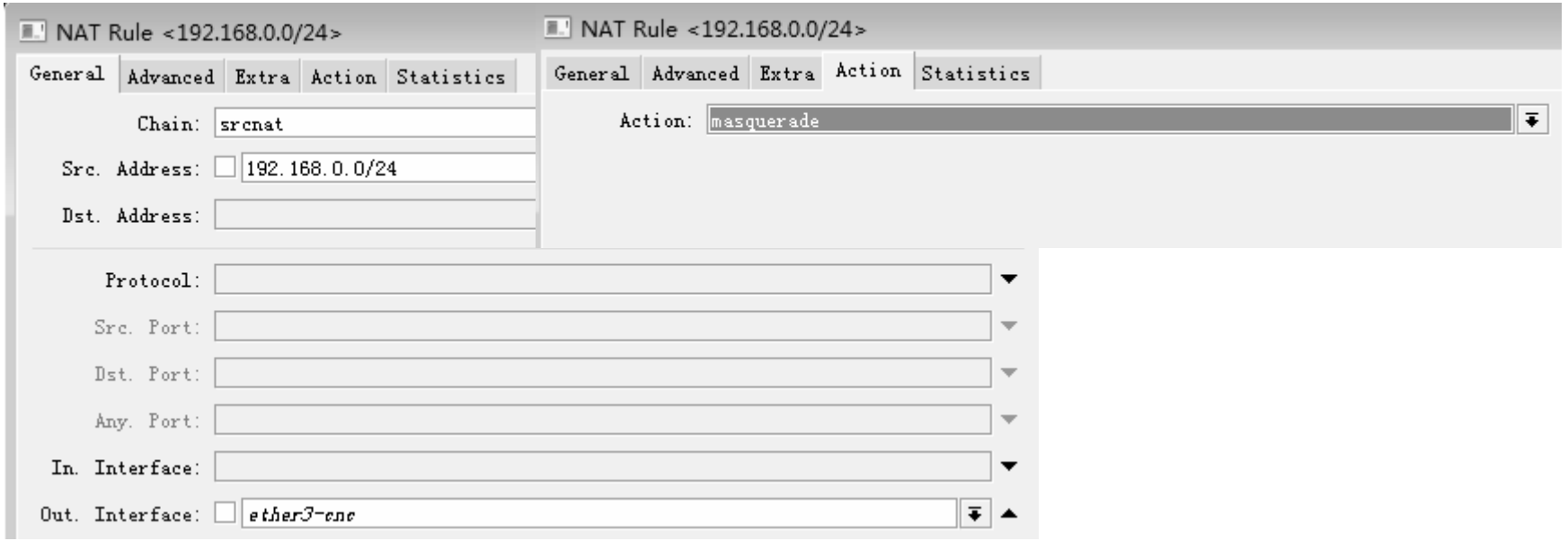
案例配置 – 双线备份（2）

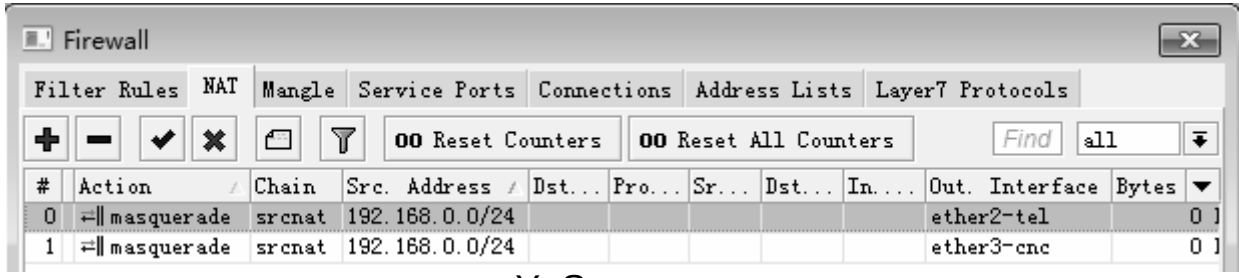
- 在ip route中增加电信备份网关，设置distance=2，这样在网通断线后，自动接替备份



案例配置 - NAT

- NAT设置，我们只需要在原理双线规则中增加一条网通的nat地址转换





案例配置 – 双线流控（1）

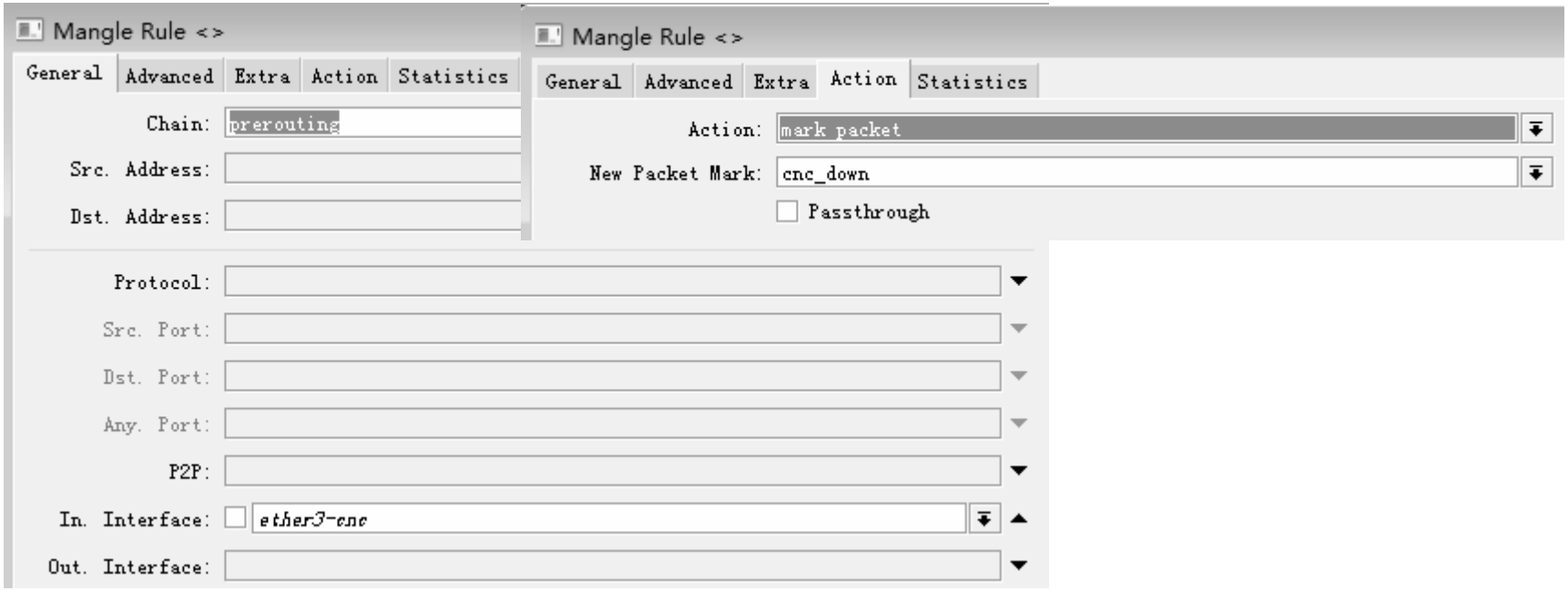
- 双线的流量控制也只需要在原理单线基础上增加对网通线路的标记、PCQ和queue tree的规则

- 这里我们将标记好的名称加上cnc，以示区分线路

- 在这个事例里，电信带宽不变仍然12M，网通15M，主机数为200台

双线流控 – Mangle标记（1）

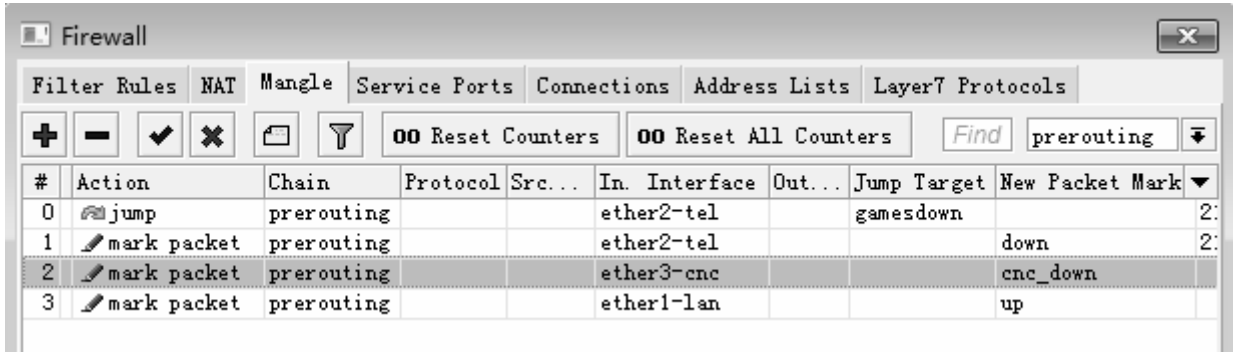
- 首先进入ip firewall mangle标记网通数据，取名cnc_down



双线流控 – Mangle标记（2）

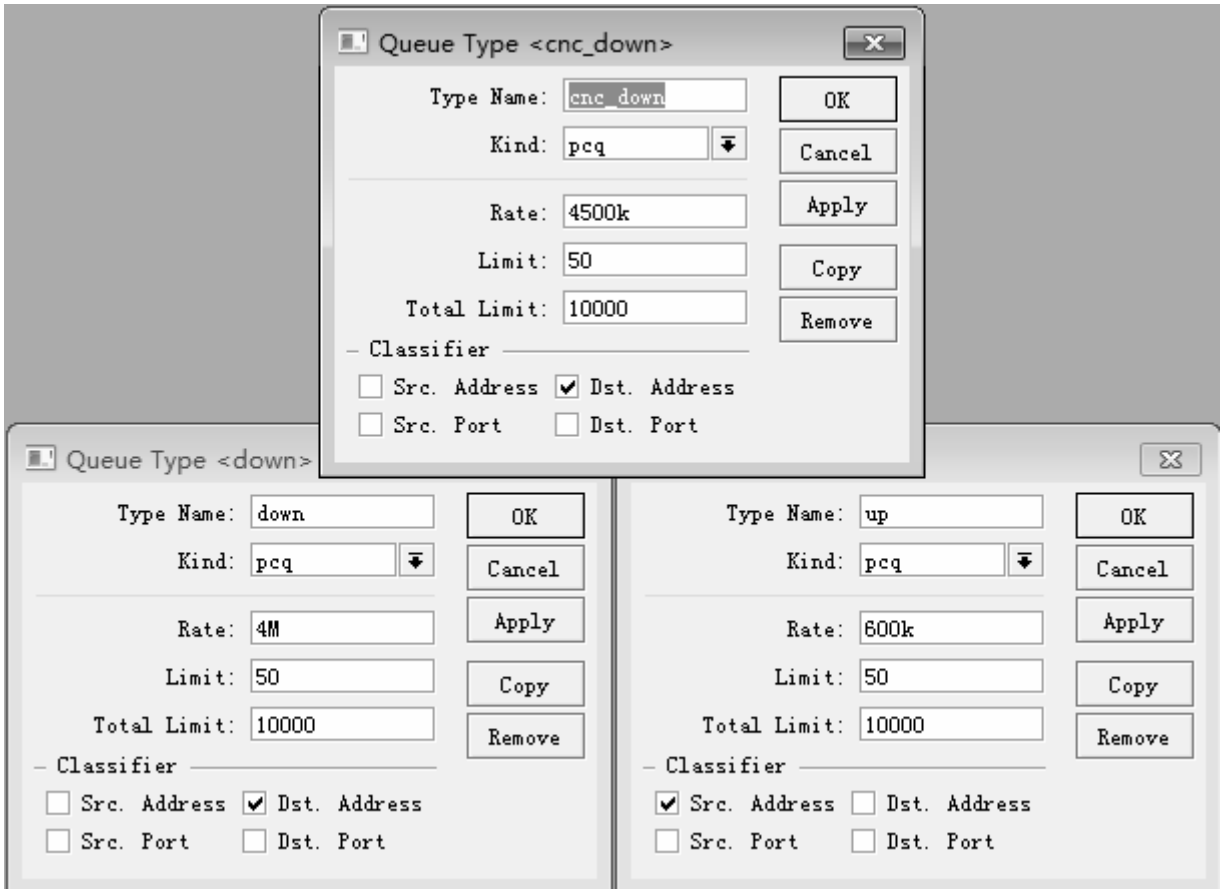
- 设置完成后，将规则移动到内网lan口数据之前

- 这里我们已经对ether1-lan口的内网上行数据进行标记，所以不需要在做其他上行控制



双线流控 – PCQ

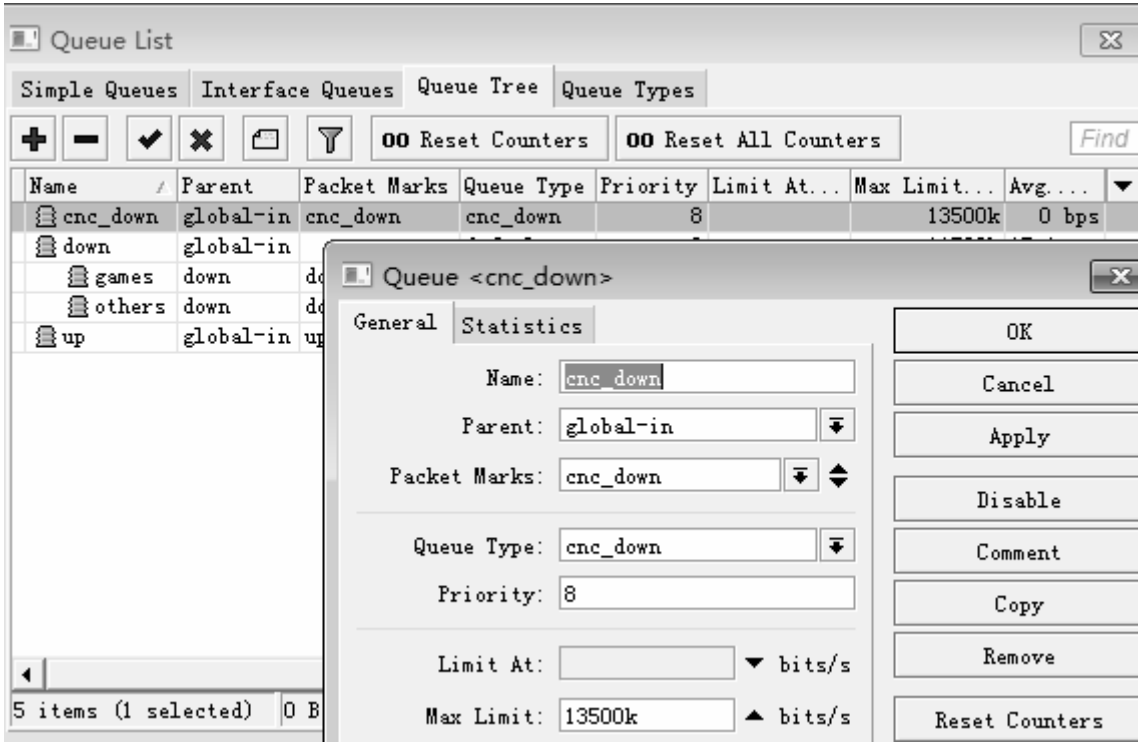
- 设置网通下行PCQ单机带宽为4.5M，由于机器数增加到200台，所以各个total-limit修改为10000



双线流控 – Queue tree

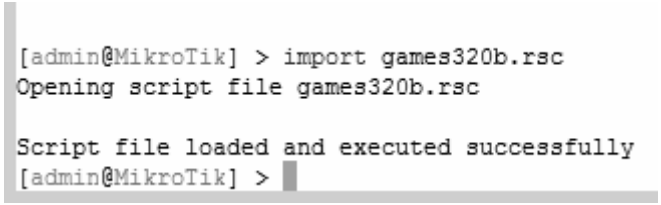
- 在queue tree里增加网通的流量控制规则，取名cnc_down，在prerouting链表定义数据流，所以选择parent=global-in，packet-marks=cnc_down，设置PCQ的queue-type=cnc_down，总带宽Max-limit=13.5M

注意：在这里我们已经将上行做了总体控制，所以不需要区分电信或者网通的上行带宽控制



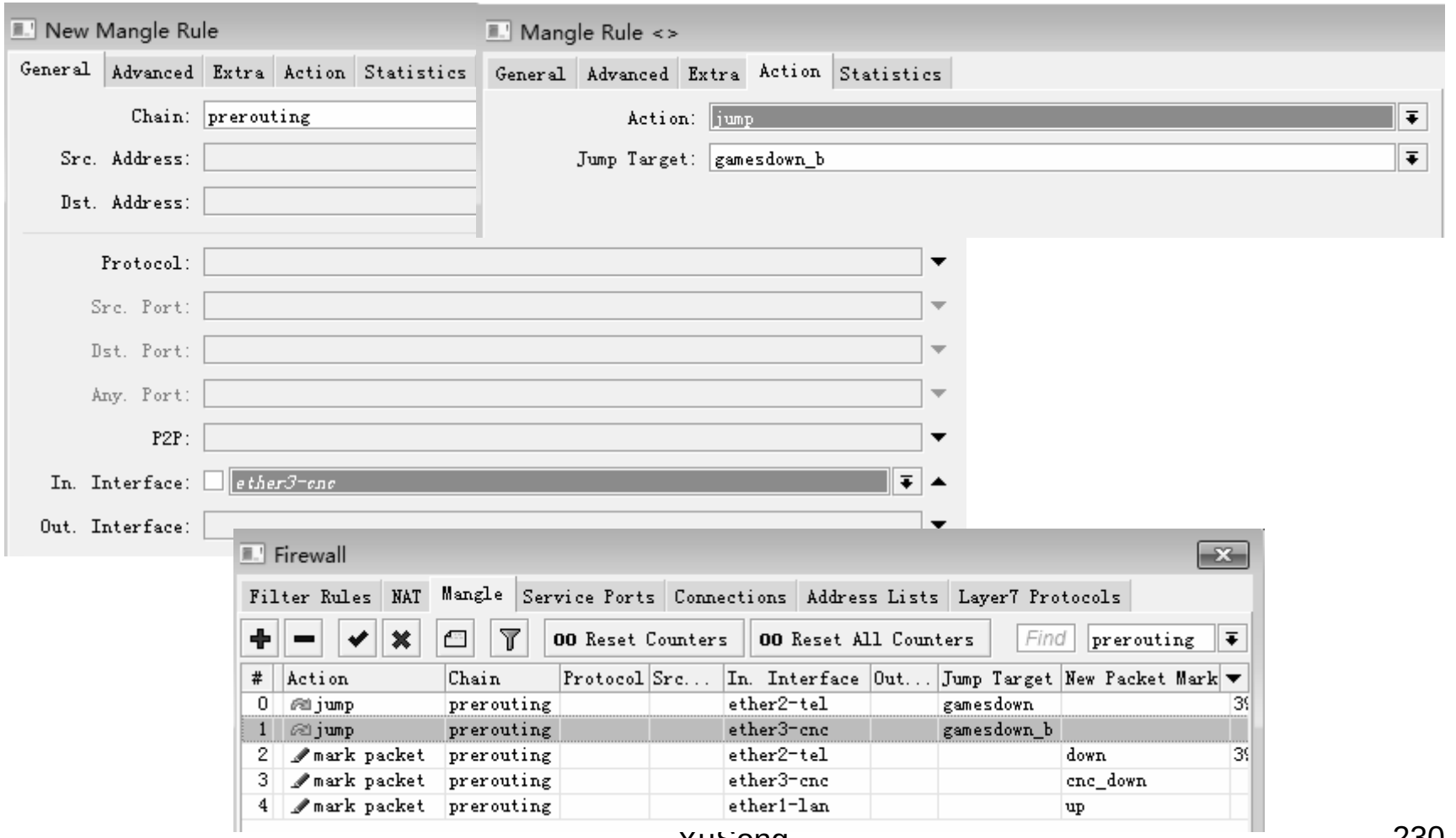
游戏优先（1）

- 在南方通常只需要做电信线路的游戏优先，而北方涉及到网通线路的游戏，所以需要做电信和网通双线的游戏优化；
- 我们只需要增加修改网通的流量控制，增加Mangle的数据标记为gamesdown_b，将进入网通数据跳转到gamesdown_b链表；
- 然后在queue tree配置网通的HTB流量控制规则
- 导入gamesdown_b，该表是在之前的游戏链表上重新取名而成，没有什么差异，只是重复了一个表，以示区分电信和网通的游戏标记

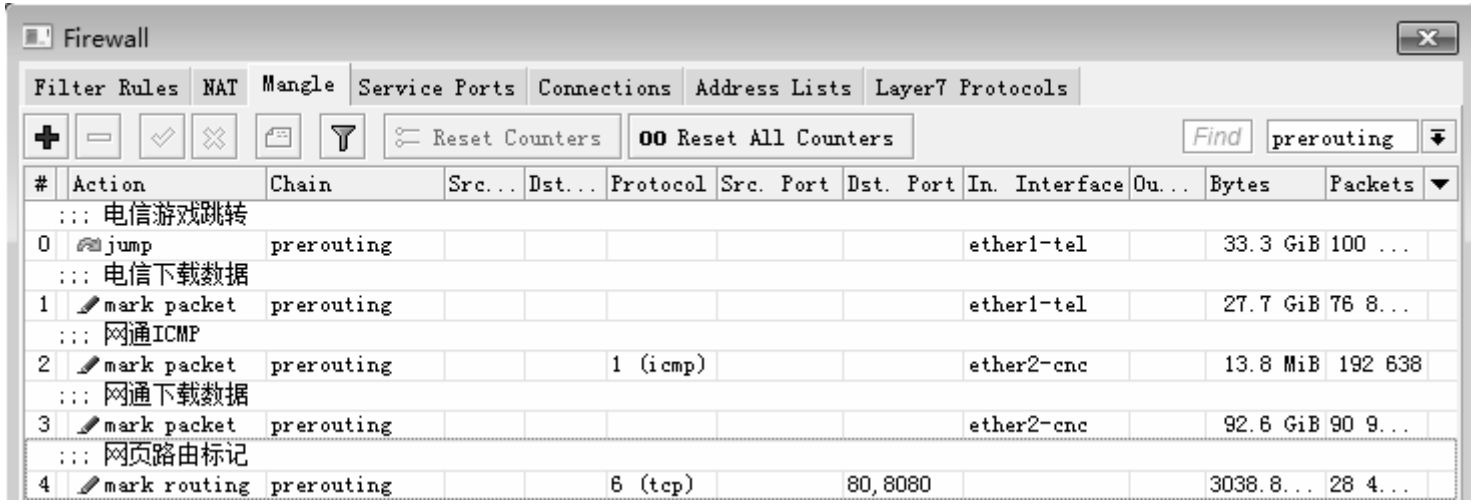


游戏优先 （2）

- 在prerouting增加进入网通的调整规则，并拖动到所有规则之前

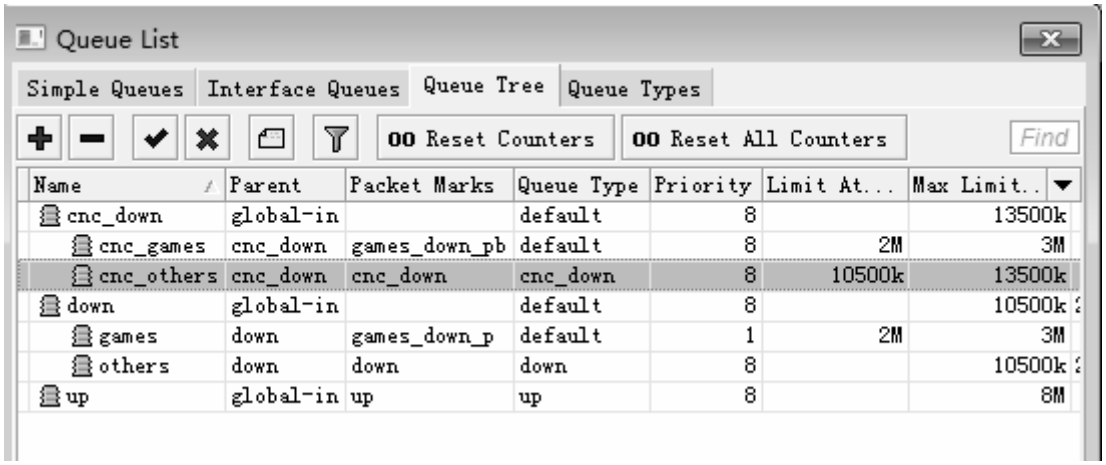


- 根据需要做好相应的注释



双线HTB

- 接下来的HTB如同之前的电信游戏优先一样，设置cnc_games，packet-marks为games_down_pb，其他完全和单线游戏配置一样



Name	Parent	Packet Marks	Queue Type	Priority	Limit At...	Max Limit..
cnc_down	global-in		default	8		13500k
cnc_games	cnc_down	games_down_pb	default	8	2M	3M
cnc_others	cnc_down	cnc_down	cnc_down	8	10500k	13500k
down	global-in		default	8		10500k
games	down	games_down_p	default	1	2M	3M
others	down	down	down	8		10500k
up	global-in	up	up	8		8M

RouterOS PPP与Hotspot

PPTP隧道

- PPTP：点对点隧道协议 (PPTP: Point to Point Tunneling Protocol）

- 点对点隧道协议（PPTP）是一种支持多协议虚拟专用网络的网络技术。通过该协议，远程用户能够通过 windows客户端或者路由器，以及其它装有点对点协议的系统安全访问公司网络，并能拨号连入本地 ISP，通过 Internet 安全链接到公司网络。

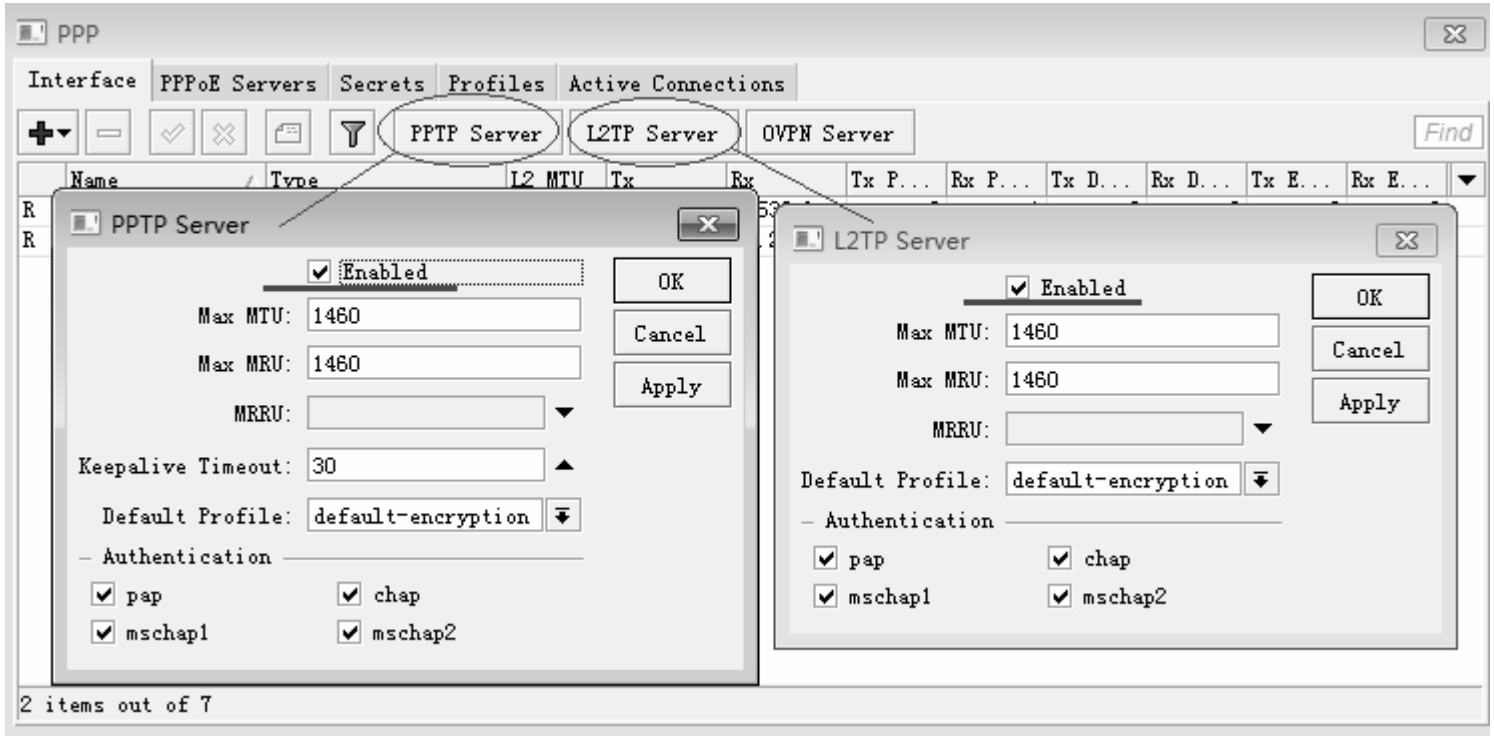
PPTP与L2TP

PPTP和L2TP都使用PPP协议对数据进行封装，然后添加附加包头用于数据在互联网络上的传输。尽管两个协议非常相似，但是仍存在以下几方面的不同：

- PPTP要求互联网络为IP网络。L2TP只要求隧道媒介提供面向数据包的点对点的连接。
- PPTP只能在两端点间建立单一隧道。L2TP支持在两端点间使用多隧道。使用L2TP，用户可以针对不同的服务质量创建不同的隧道。
- L2TP可以提供包头压缩。当压缩包头时，系统开销（overhead）占用4个字节，而PPTP协议下要占用6个字节。
- L2TP可以提供隧道验证，而PPTP则不支持隧道验证。但是当L2TP或PPTP与IPSEC共同使用时，可以由IPSEC提供隧道验证，不需要在第2层协议上验证隧道

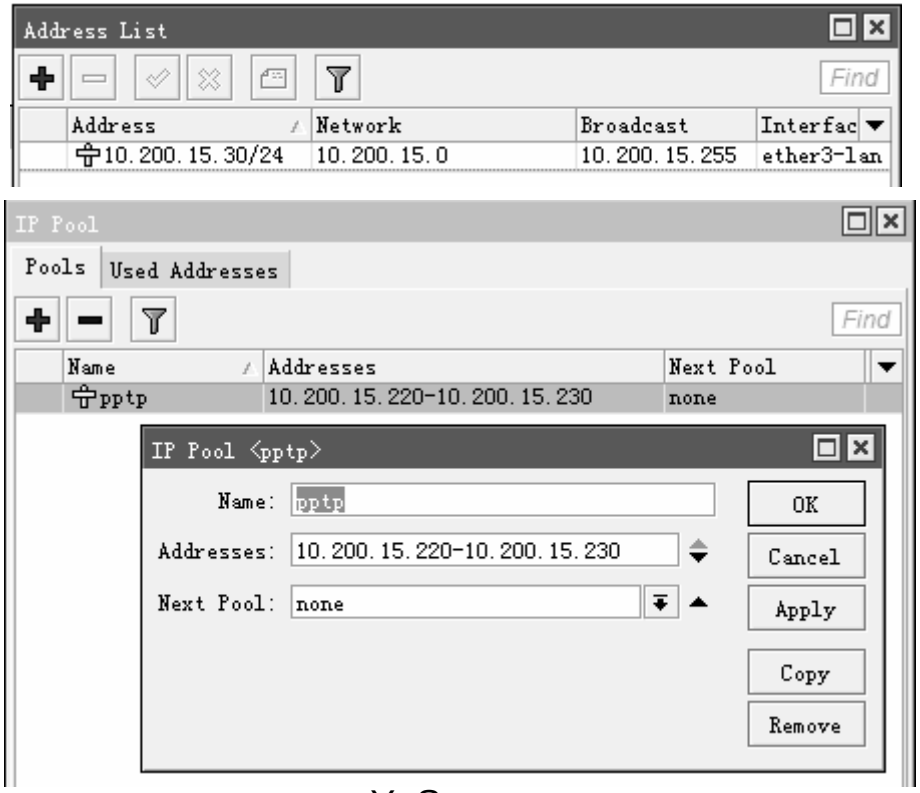
启用PPTP和L2TP服务

- 进入ppp启用PPTP server和L2TP Server



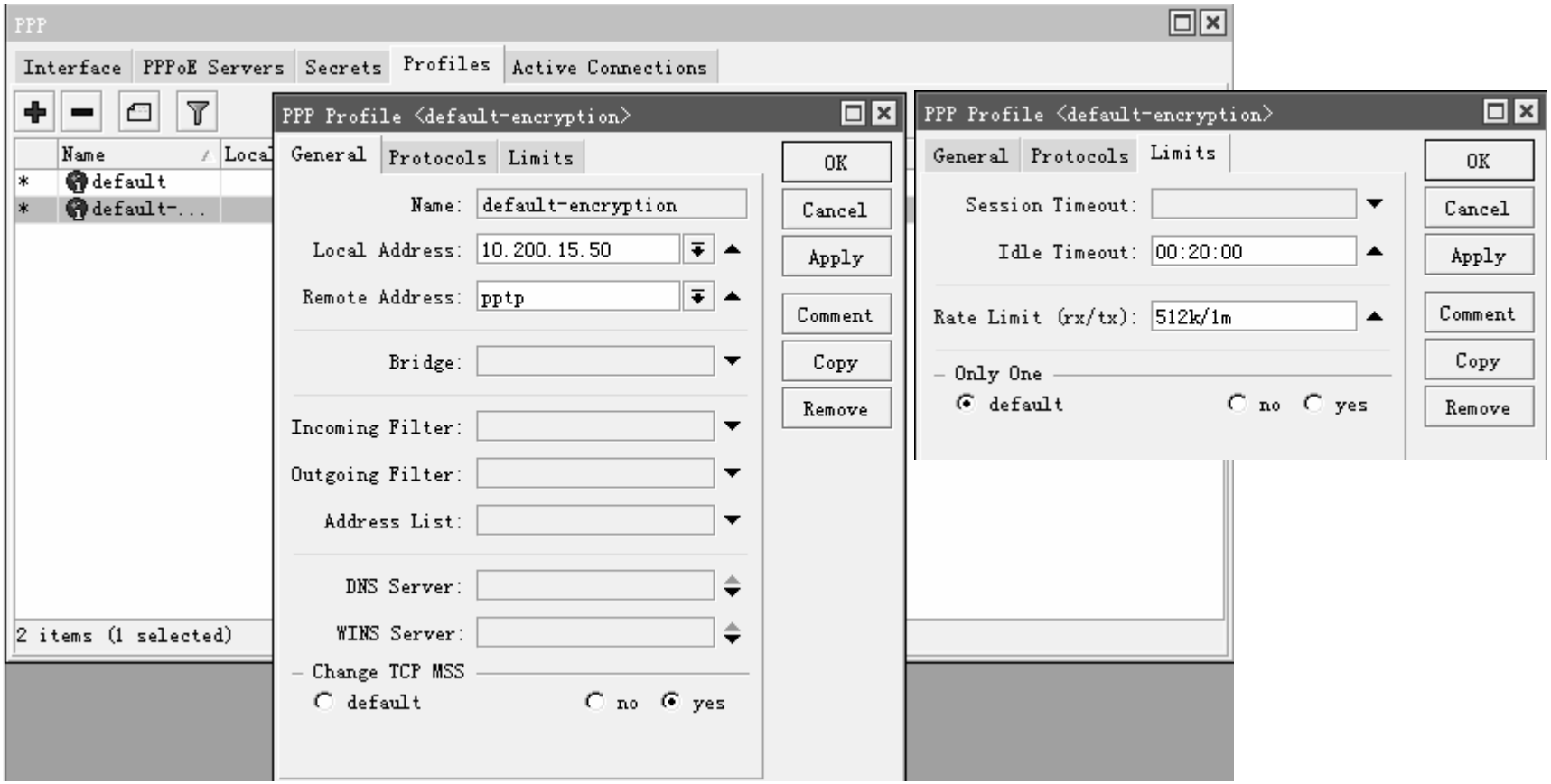
配置PPTP服务器

- 通过RouterOS架设一个PPTP服务器，假设我们的路由器IP地址是10.200.15.50
- 并在ip pool里分配地址池，地址范围10.200.15.220-10.200.15.230用于客户端地址分配



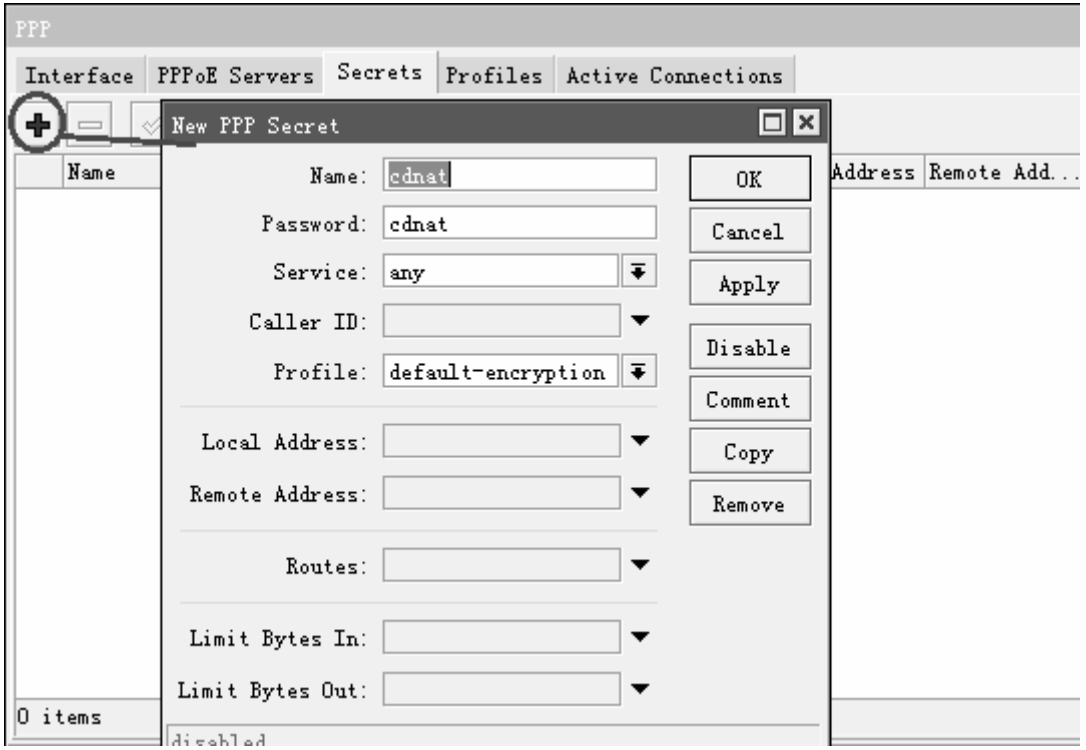
配置PPTP服务器

- 启用服务器前，我们首先配置profile规则，设置local-address=10.200.15.50，remote-address=pptp，设置相应的Idle-timeout为20分钟，带宽上行512k，下行1M



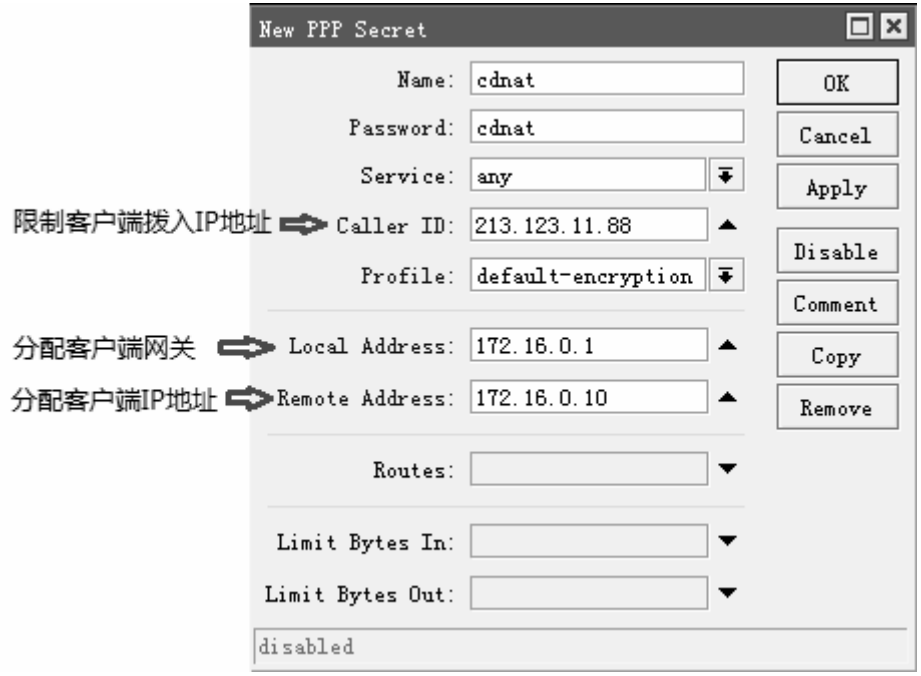
配置PPTP服务器

- 添加PPTP的客户端账号，name=cdnat，password=cdnat，profile=default-encryption



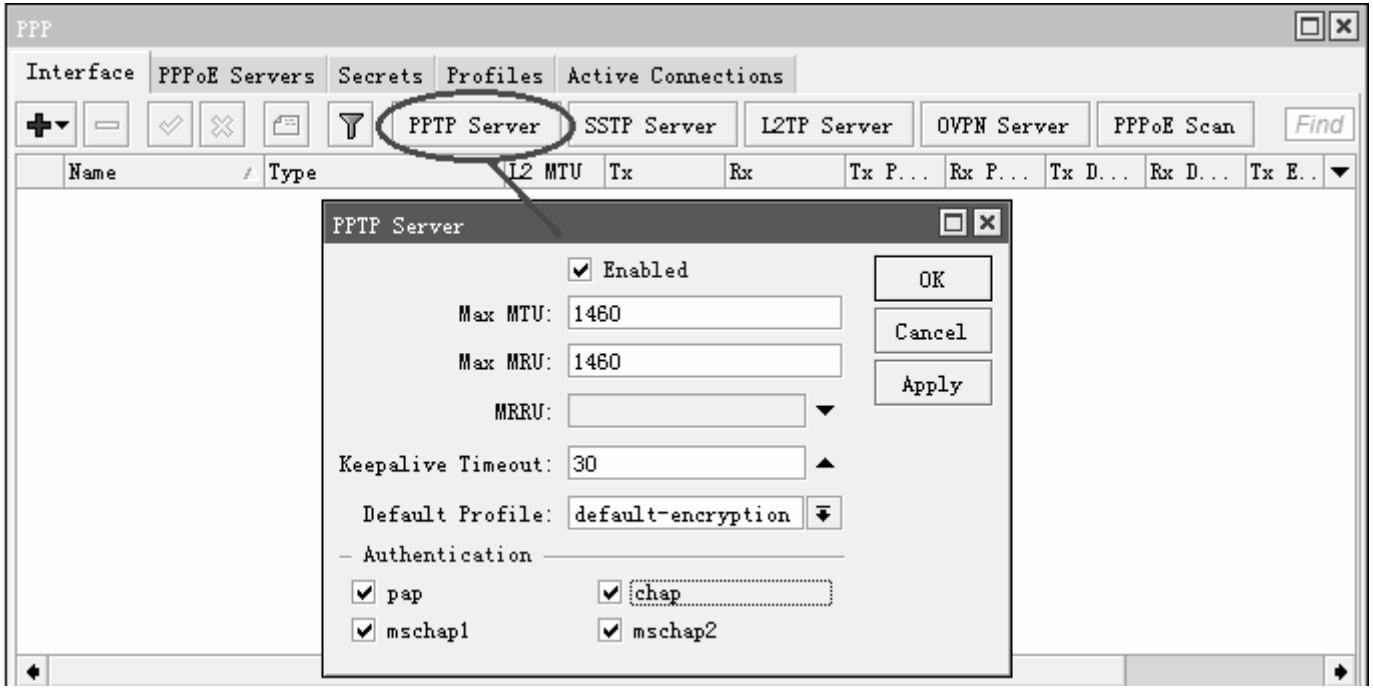
配置PPTP服务器

- Caller-ID，绑定客户端拨号IP地址（PPPoE账号是绑定MAC地址）
- Local-address，可以给账号分配固定网关
- Remote-address，可以给账号分配固定IP
- 在secret里设置local-address和remote-address优先级高于profile规则



配置PPTP服务器

- 我们进入winbox的ppp菜单，启用PPTP服务器



Windows拨号连接

- 帐号建立完成后，通过Windows创建客户端拨号程序，连接到RouterOS服务。我们可以在windows的网上邻居-属性中新建一个PPTP拨号连接。

L2TP Windows连接

L2TP修改注册表

缺省的Windows XP L2TP 传输策略不允许L2TP 传输不使用IPSec 加密。可以通过修改Windows XP 注册表来禁用缺省的行为：

手工修改：

1) 进入Windows XP 的“开始” “运行”里面输入“Regedt32”，打开“注册表编辑器”，定位“HKEY_Local_Machine \ System \ CurrentControl Set \ Services \ RasMan \Parameters ”主键。

2) 为该主键添加以下键值：

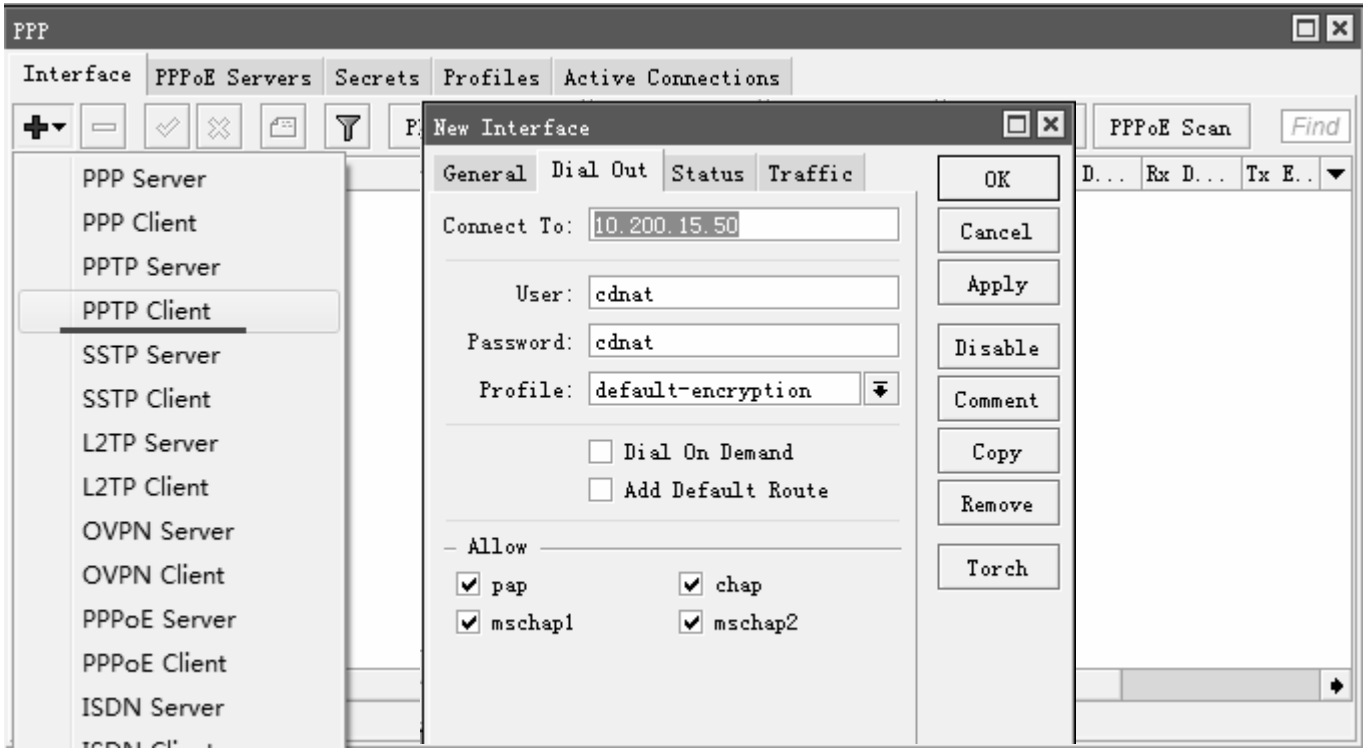
a. 键值：ProhibitIpSec

b. 数据类型：reg_dword

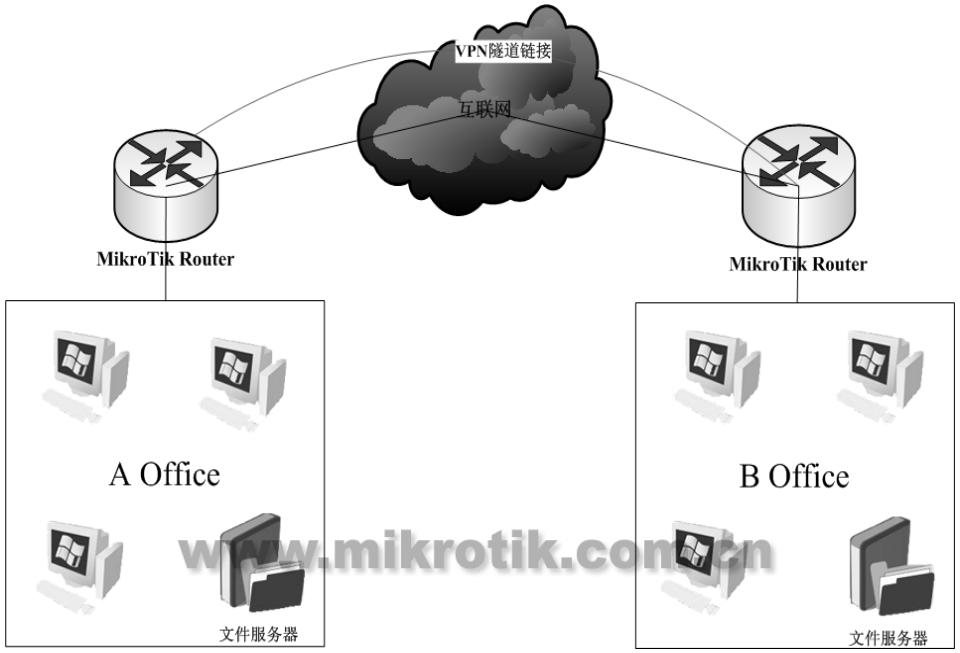
c. 值：1

RouterOS Client拨号连接

- 拨号客户端我们在interface中分别选择 pptp-client和L2TP-client，并设置相应的拨号地址和帐号密码。
- Profile的规则必须为默认配置，否则出现拨号失败。

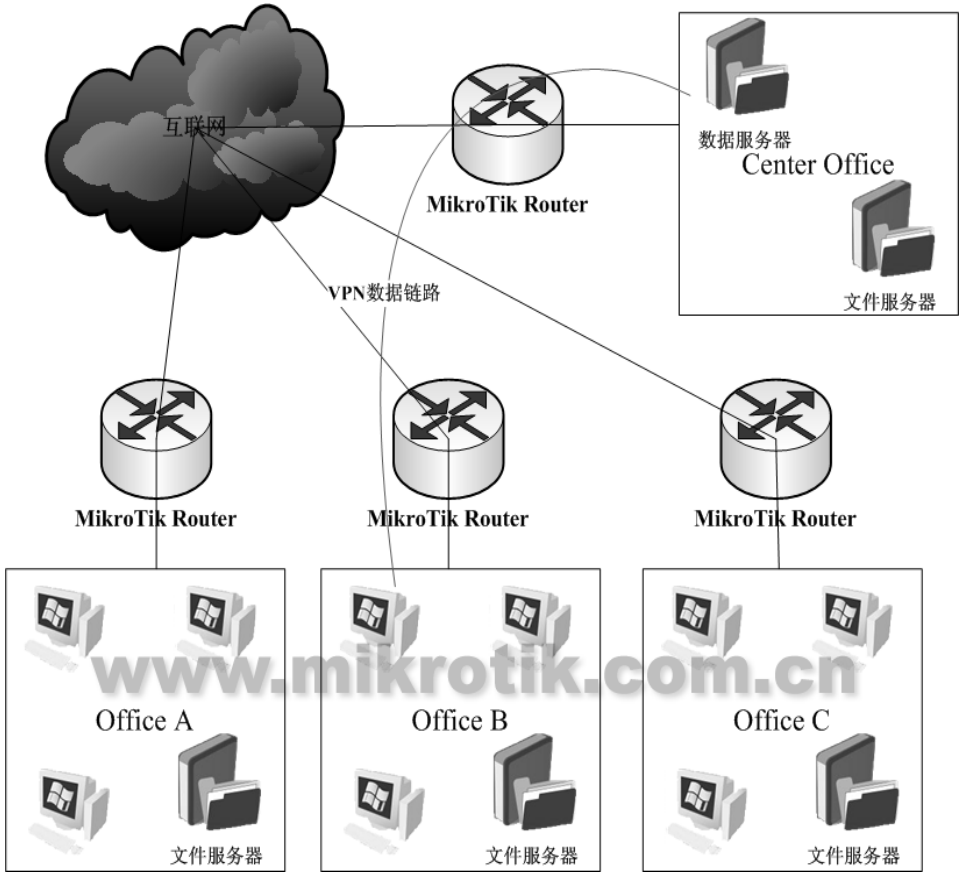


企业间对等互访



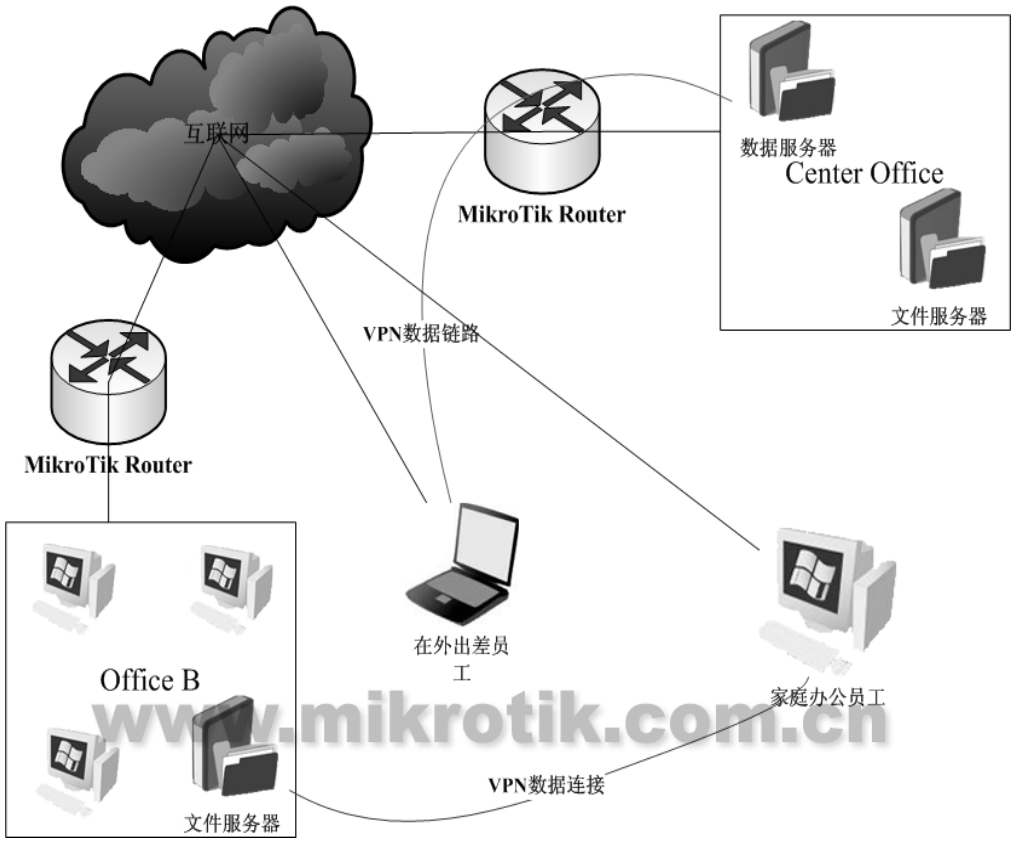
- 建立PPP服务器，并给客户端分配帐号和固定IP地址
- 建立PPP的拨号，并分配IP地址
- 设置两个远程局域网的IP地址路由

企业与分支点总分连接



- 分配PPP帐号，给没一个帐号分配固定IP地址
- 并建立多个点的VPN连接
- 设定每个路由器之间的路由规则

企业办公综合应用



- 这种方式为前两种的综合应用
- 移动外出办公分配一组profile
- 固定办公则使用不同的profile规则

ISP网络应用

- 综合考虑网络内接入用户数
- 采用什么样做骨干连接，如光纤或者WLAN连接
- 根据用户数量和网络构建选择相应的硬件设备
- 网络采用二层传输，还是多级路由连接
- 根据网络环境采用PPPoE，还是Hotspot认证

HotSpot热点网关

HotSpot热点认证网关特征：

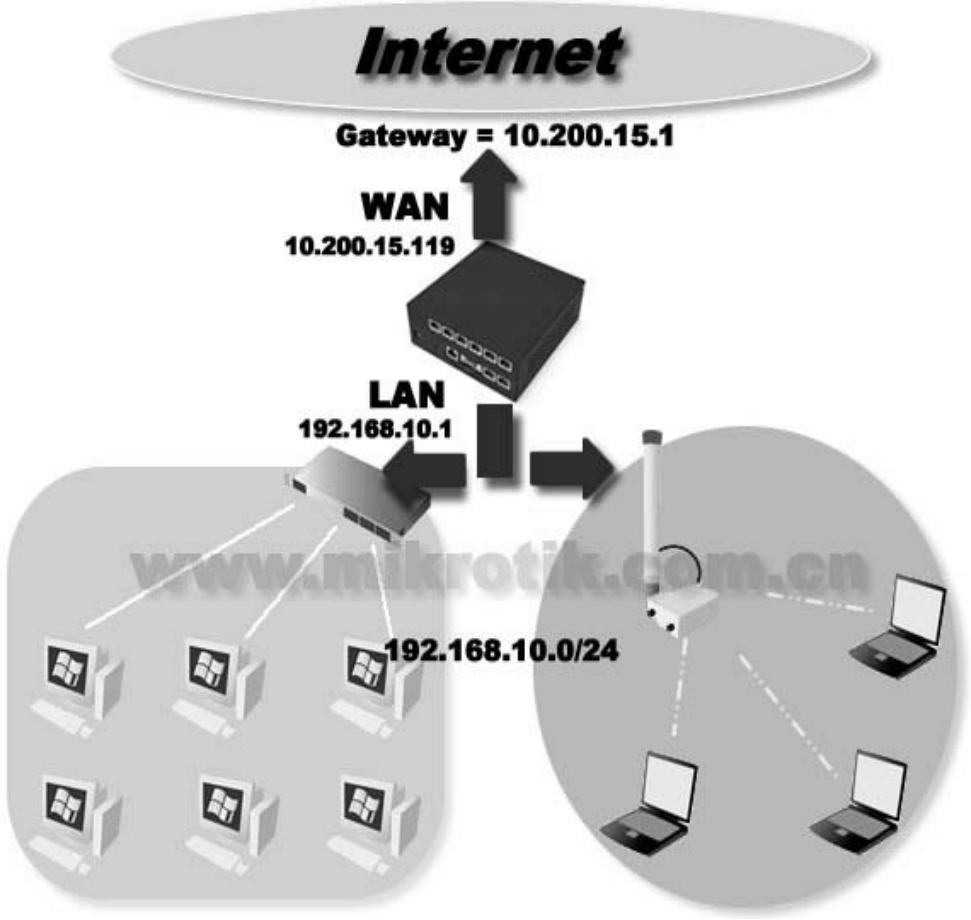
- 不需要用户安装客户端软件
- 提供一个友好的web认证接口
- 提供开放的html文件，编辑自己的认证页面
- 自带用户帐号管理、计时与流量记录功能
- 支持UpNp即插即用功能，方便移动用户上网
- 支持Radius管理

Hotspot应用范围

- 小范围的ISP运营商的认证系统
- 与WiFi结合提供公共场所的热点认证，如图书管、机场和学校
- 商务酒店的上网管理
- Hotspot的缺点：容易受到ARP病毒的攻击

Hotspot配置事例 1

- WAN口对应外网IP为10.200.15.119/24
- 网关：10.200.15.1
- LAN口对应内网IP：192.168.10.1/24
- DNS：61.139.2.69



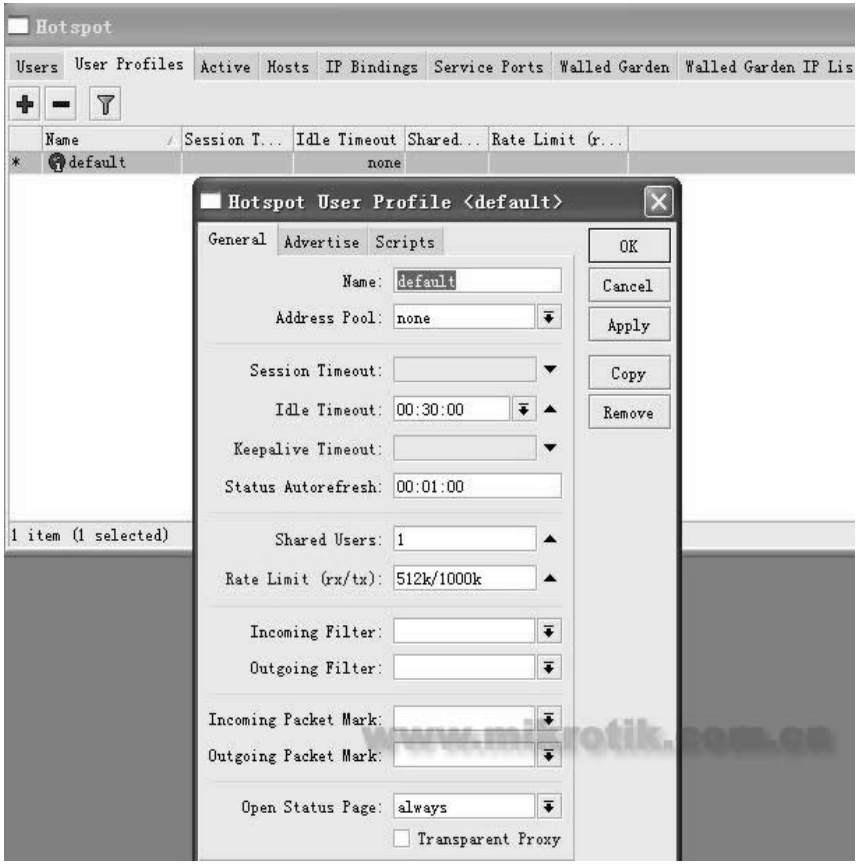
Hotspot配置事例 2

在配置好基本的网络参数后，Hotspot配置如下：

- 1、先进入ip hotspot user profile设置用户分组规则；
- 2、然后在ip hotspot user添加用户的帐号；
- 3、进入ip hotspot server profile配置服务器规则；
- 4、在ip pool中分配IP地址段，根据需要启用DHCP服务；
- 5、在ip hotspot server添加并启用hotspot服务。

Hotspot配置事例 3

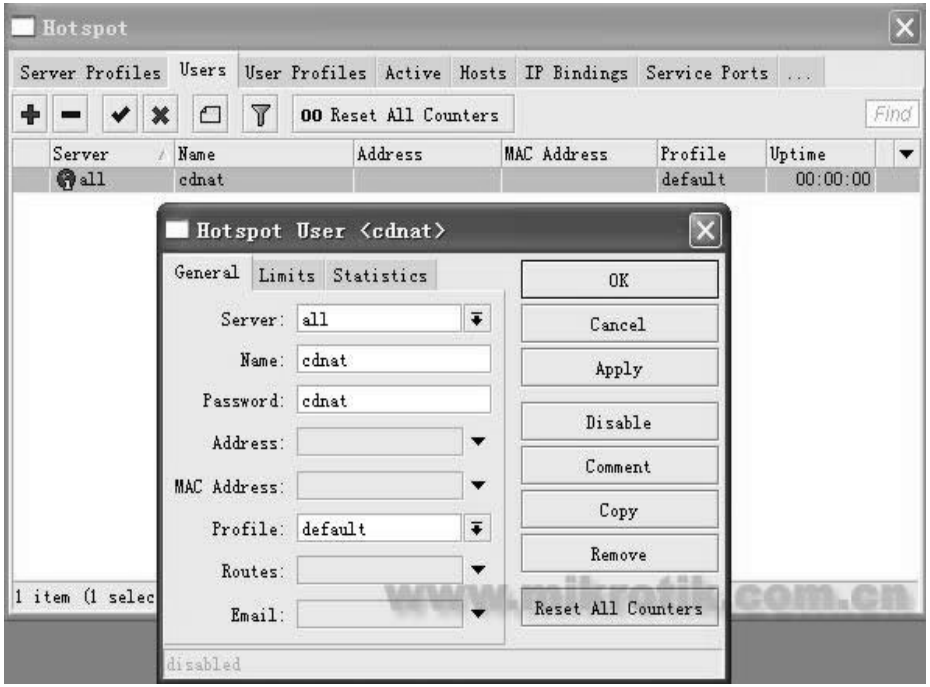
Hotspot user profile配置



- Idle-Timeout：用户在一定时间内没有任何流量发出后自动注销连接
- Keepalive-Timeout：路由器主动通过ICMP探测主机是否在线，如果在一定时间为探测到自动注销连接（如果用户机开启防火墙，路由器无法探测到）
- Shared-users：帐号的分享用户多少，默认为1，即仅一个用户使用该帐号。
- Rate-Limit：分配每个帐号带宽，格式为“上行 / 下行”
- Transparent-proxy：透明代理功能是否开启，一般使用Hotspot认证建议不用打开此参数。

Hotspot配置事例 4

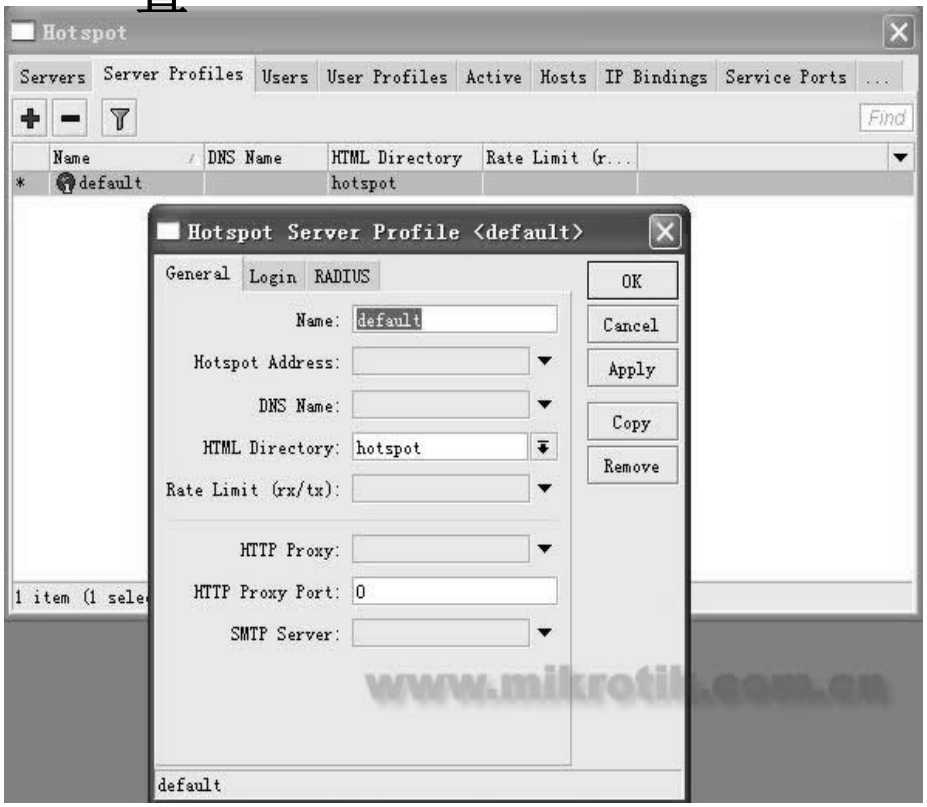
Hotspot user配置



- 这里默认server 服务器为all，也可分配相应服务器。
- Name 用户名：cdnat
- Password：cdnat
- Profile：用户组规则，这里选择我们之前设置的default规则
- 配置完用户规则后，进入ip hotspot server profile，配置服务器器规则。

Hotspot配置事例 5

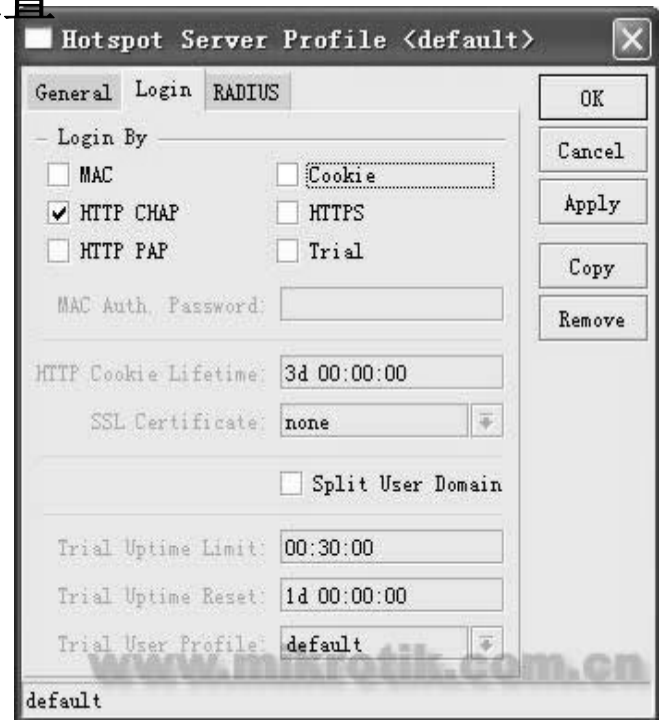
Hotspot server profile配置



- server profile配置服务器参数
- 在General选项中选择HTML Directory 为默认的hotspot文件路径，同时也可以选择自己定义的文件名路径。

Hotspot配置事例 6

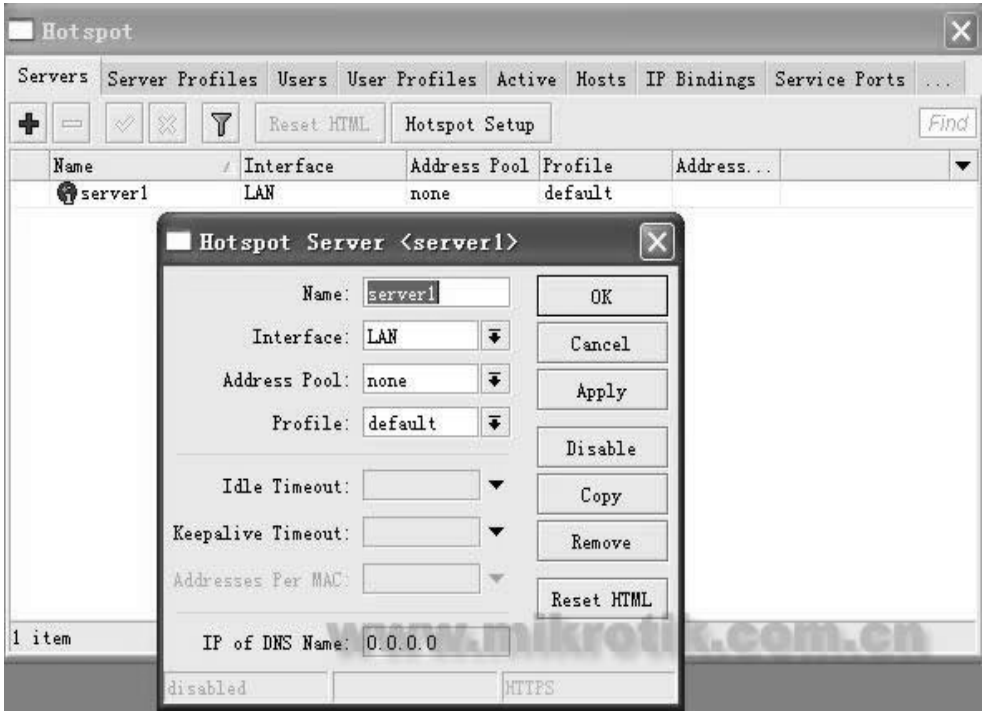
Hotspot server profile login 配置



- 配置login登录方式，一般只启用http chap即可，其他选项根据需要开启。
- Cookie值根据需要开启

Hotspot配置事例 7

Hotspot server 配置

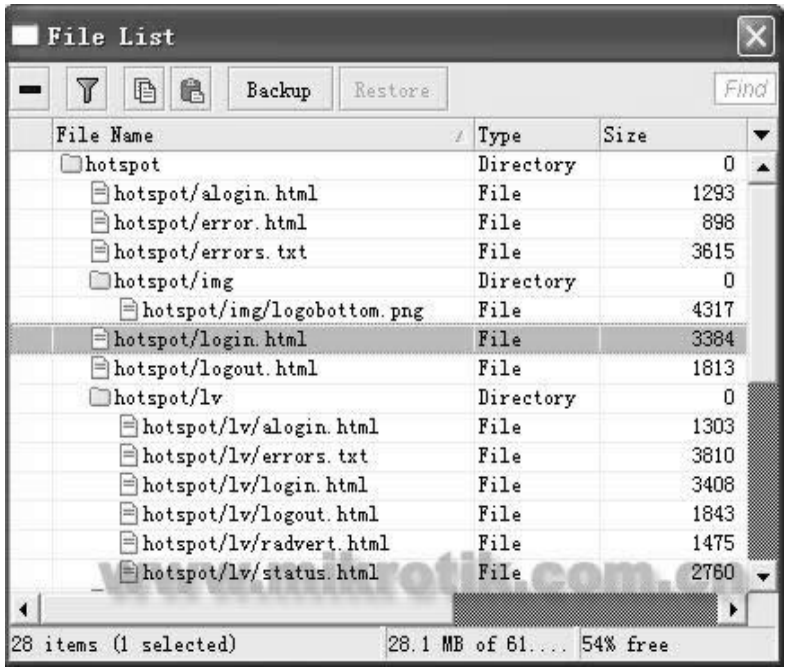


- 最后我们启用Hotspot服务器；
- 选择内网LAN口，作为认证接口；
- 当启用完成后，所有对路由器或者外网访问数据，都需要通过web认证。

Hotspot配置事例 7

- 打开认证也输入帐号和密码，通过认证后，在hotspot active中会显示状态

Hotspot页面修改



- Hotspot的认证登录页面是开放式的，即可以通过RouterOS的files目录下找到这些文件，在files中的默认文件名“Hotspot”

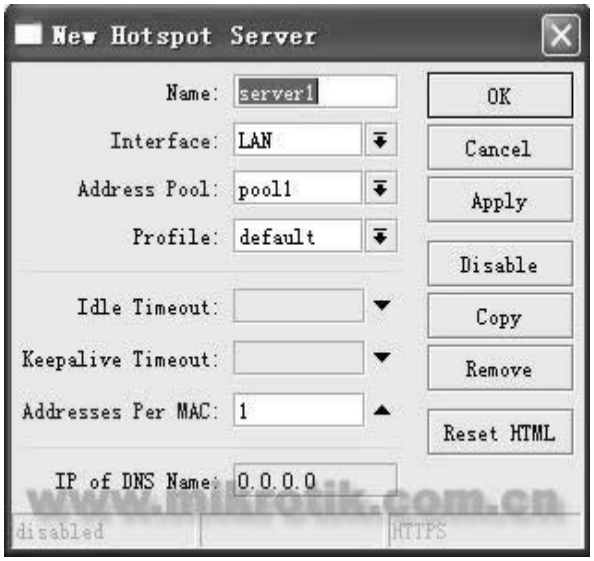
- 认证页面我们可以通过修改login.html、logout.html和status.html的web界面得到你想要的网页画面或者log。

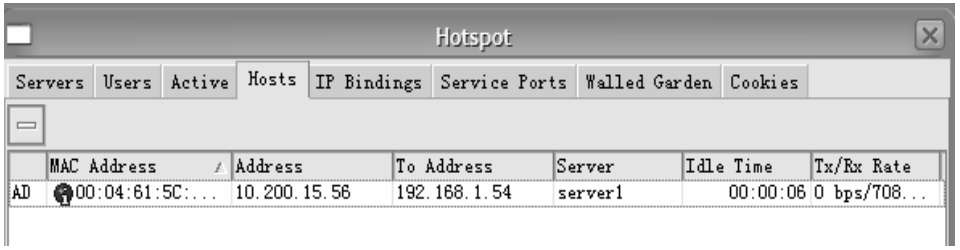
Hotspot UpNp

- 从2.7的版本就开始支持upnp的即插即用功能，即当用户和Hotspot认证服务器在同一局域网内，不管局域网用户设置任何的IP地址（前提是用户必须设置任意的IP地址、网关和DNS）都可以被Hotspot认证服务器获取，并在Hotspot的Host中分配一个新的虚拟IP地址，并对用户作一对一的NAT转换。

- Hotspot的即插即用方式分成适用于：流动性较强的公共场所，如机场、车站、公园，也可以应用到酒店和小区中。

Hotspot UpNp配置





- 在2.9和3.0的Hotspot启用server服务后，即插即用功能默认是打开的，但配置Hotstop需要在hotspot server中将address pool的地址池设置好
- Addresses Per MAC这个是每个IP对应的MAC地址，这里我们设置为1，即一个IP对应一个MAC地址。

PPPoE隧道协议

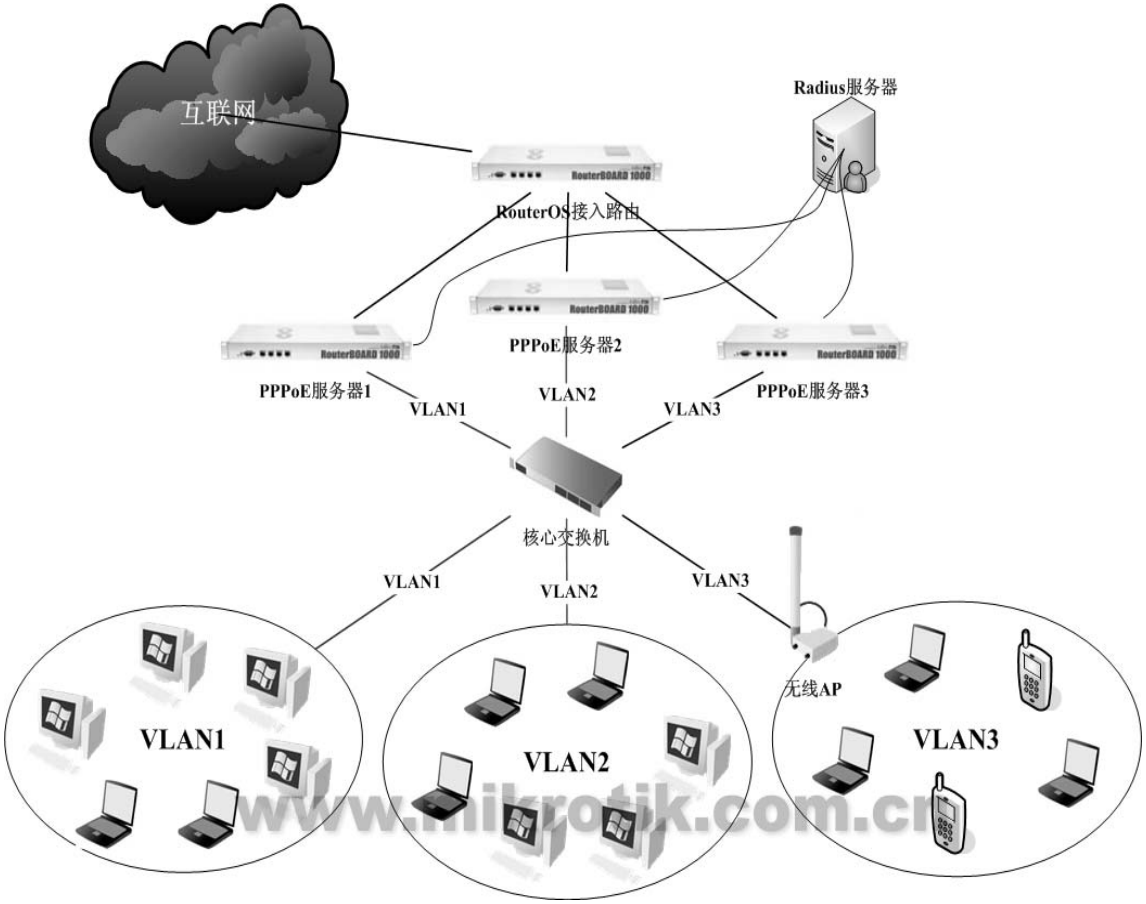
- PPPoE 是一种标准的点对点协议(PPP) 他们之间只是传输上的差异：PPPoE隧道建立是基于二层链路，一般来说, PPPoE是基于与用户认证后通过分发IP地址给客户端。
-
- 基于二层隧道认证，更加安全；完全解决ARP病毒问题

- 要求用户配置客户端拨号程序，只能基于二层的网络认证

PPPoE应用范围

- ISP运营商的认证系统（PPPoE能有效解决ARP问题被大多运营商所采用）
- 小区和学校网络的认证系统
- 在一些网吧为了避免ARP病毒的侵扰，也在网吧内部建立的PPPoE认证方式，避免ARP对网吧带来上网电脑频繁掉线问题。

PPPoE认证系统构建



大型的PPPoE系统

- 接入路由器负责nat、防火墙和路由
- PPPoE服务器提供PPPoE拨号认证
- VLAN核心交换机负责链接和隔离内部网络

配置PPPoE服务准备

- 确定网络的在线人数（供参考）

  100-500人：采用单台，即nat与PPPoE服务器在一台上

  5**00-1200**人：采用两台设备，nat路由和PPPoE服务器

  1200人以上：采用多台设备，nat路由和多台PPPoE服务器

- 确定网络环境

  小区与校园网络：通过中心机房的核心交换连接到每个终端

  ISP运营网络：核心交换机划分VLAN，通过光纤连接到每个分节点网络， 再从分节点到每个终端

  无线网络环境：通过WLAN连接到每个节点，通过节点连接终端

- 网络结构

  PPPoE是纯二层的隧道链接，所以中间设备不能有三层设备，否则无法到达客户端

PPPoE配置事例

- nat路由器作为接入路由连接Internet
- PPPoE server提供PPPoE的认证服务器
- PPPoE服务器负责认证，nat路由器负责nat运算和处理，在nat路由上配置目标路由将数据指向PPPoE和内网的用户

配置考虑

- nat路由器外网WAN地址：218.11.11.2，内网LAN口配置172.16.0.1/28。
- PPPoE服务器WAN口配置：172.16.0.2/28，PPPoE服务的内网配置192.168.10.1/24，分配给下面客户的IP地址范围192.168.10.10-192.168.10.254。
- 为了内网192.168.10.0/24的用户通过nat路由器访问互联网，需要配置目标路由，dst-address=192.168.10.0/24 gateway=172.16.0.2。
- nat路由配置好外网网关218.11.11.1和nat规则。PPPoE Server配置默认网关为nat路由器的LAN口地址172.16.0.1。

配置nat路由器

- **IP地址**

/ip address

add address=218.11.11.2/24 interface=wan

add address=172.16.0.2/28 interface=lan

- 路由配置

/ip route

add gateway=218.11.11.1

add dst-address=192.168.10.0/24 gateway=172.16.0.2

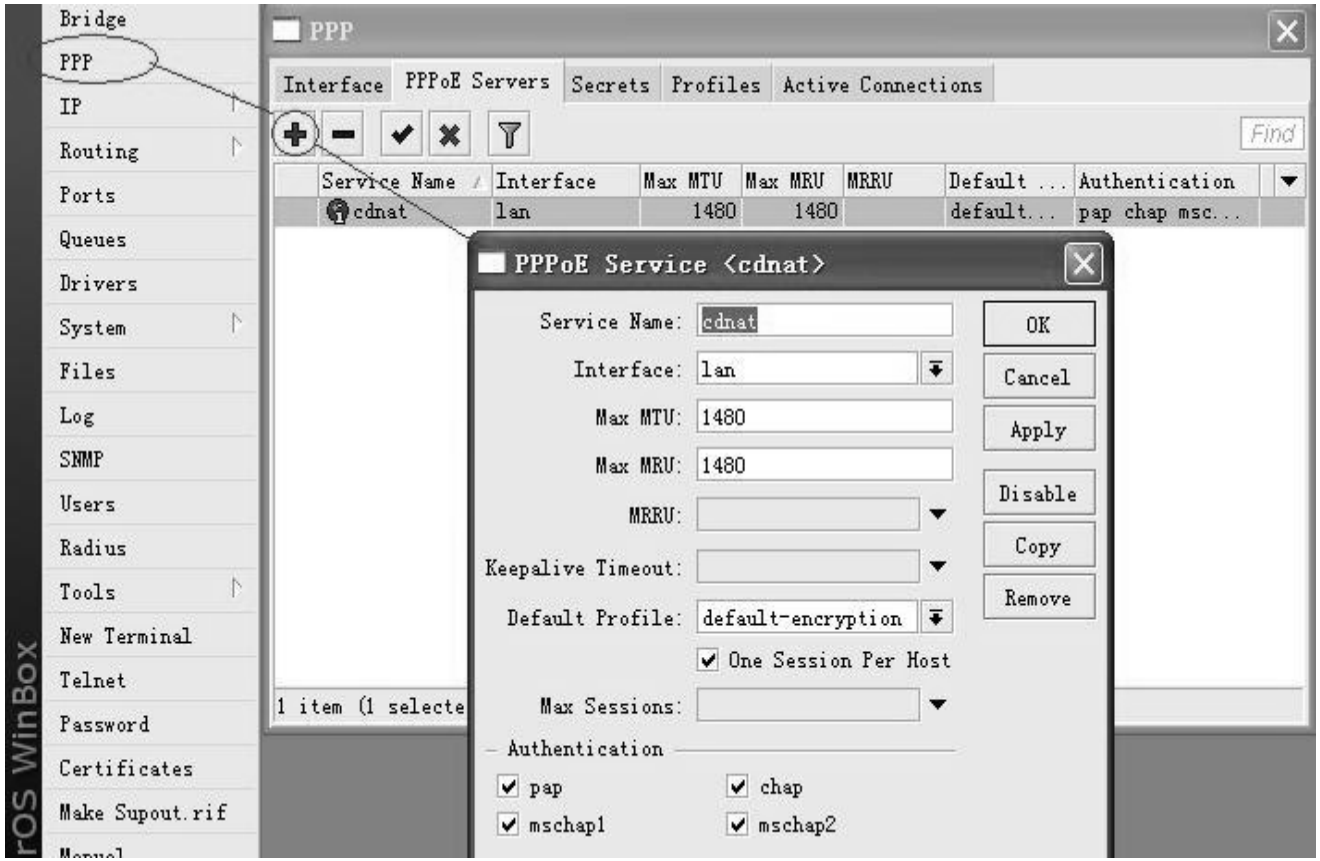
- **nat配置**

/ip firewall nat

add action=masquerade chain=srcnat out-interface=wan

PPPoE服务器配置 1

- 启用PPPoE Server服务，如下



PPPoE服务器配置 2

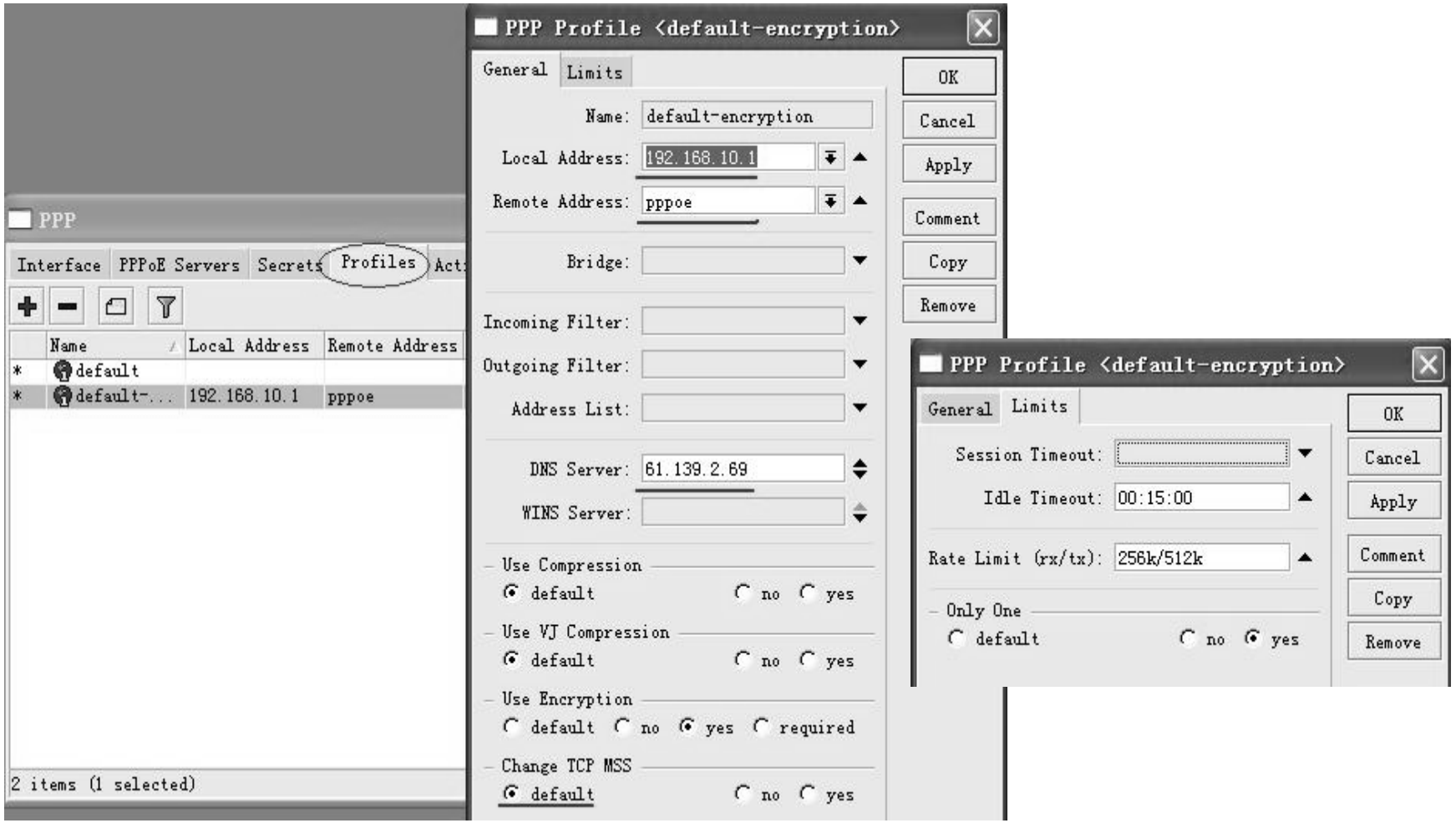
- 首先进入**/ip pool** 建立地址池

```
/ip pool
 add name=pppoe ranges=192.168.10.10-192.168.10.254
```

- 建立用户分组**/ppp profile**，使用**default-encryption**分组

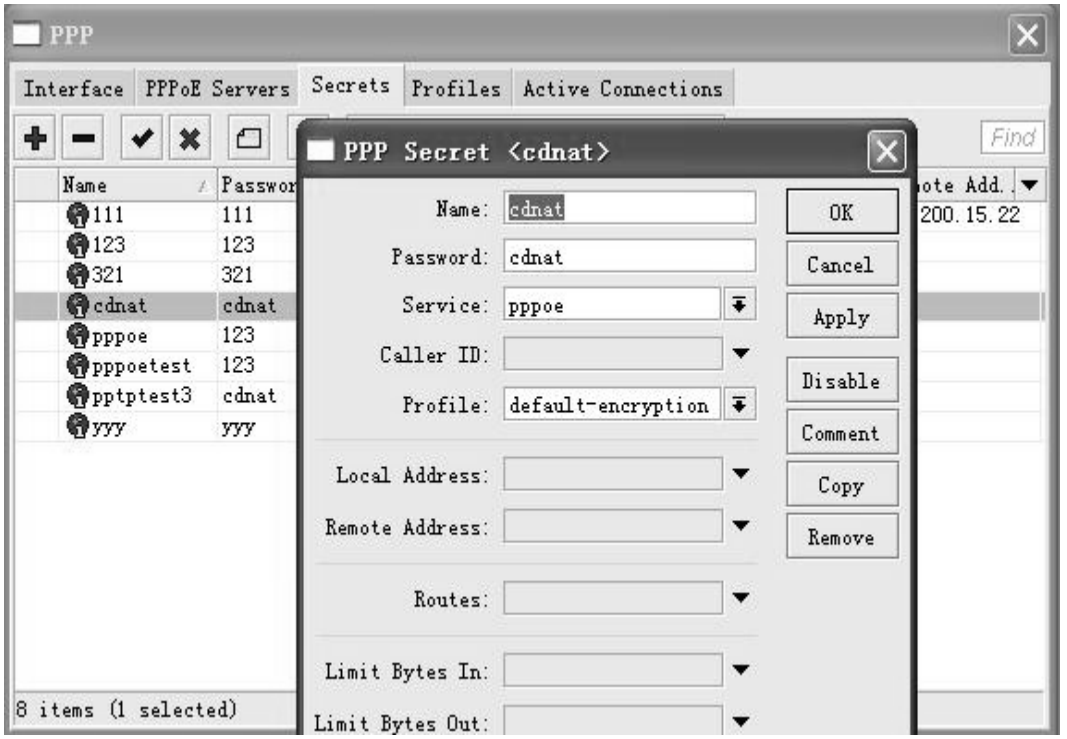
```
/ppp profile
 set default-encryption change-tcp-mss=default comment="" dns-
 server=  61.139.2.69 idle-timeout=15m local-
 address=192.168.10.1 name=  default-encryption only-one=yes
 rate-limit=256k/512k remote-address=pppoe  use-
 compression=default use-encryption=yes use-vj-
 compression=default
```

PPPoE服务器配置 3



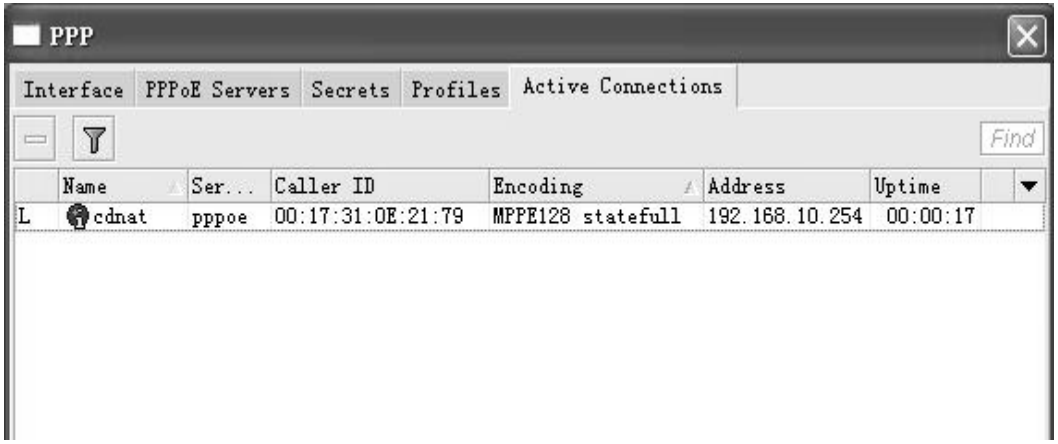
PPPoE服务器配置 4

- 在/ppp sercets添加用户帐号和密码
- 选择profile为default-encryption
- Caller ID 为客户端的MAC地址，可以通过设置与帐号绑定

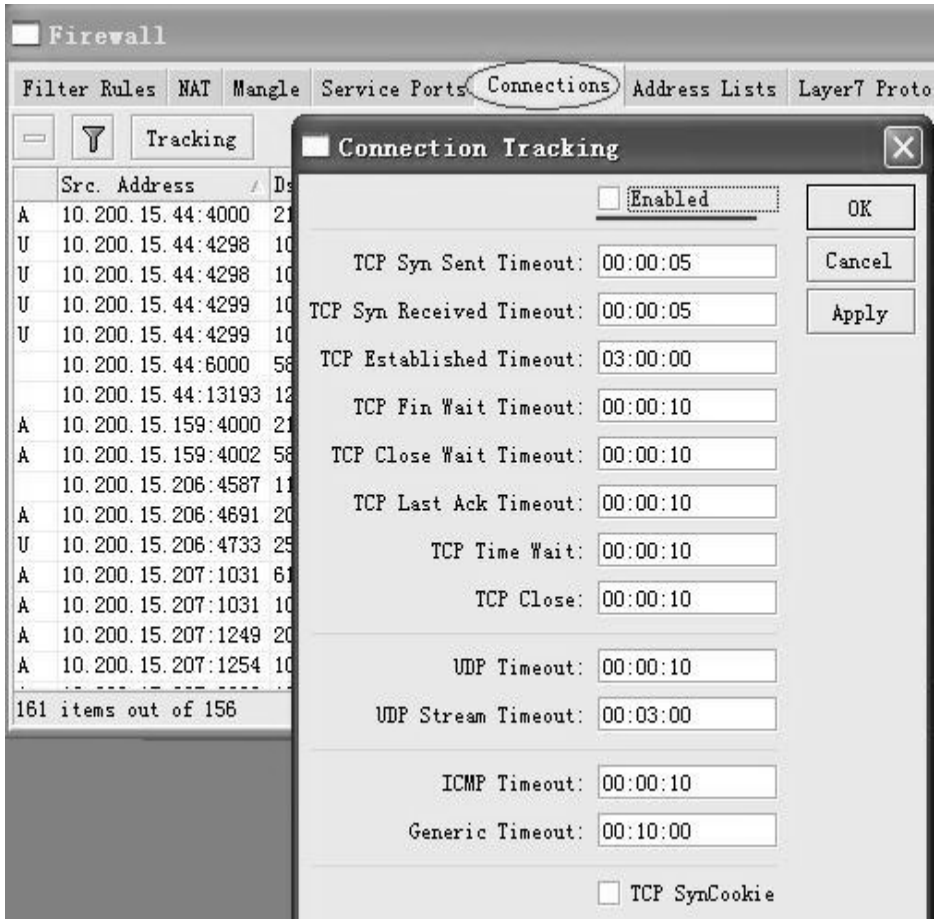


PPPoE服务器配置 5

- 最后我们通过windows拨号，连接到RouterOS的PPPoE服务器
- 我可以在/ppp active connection查看连接状态

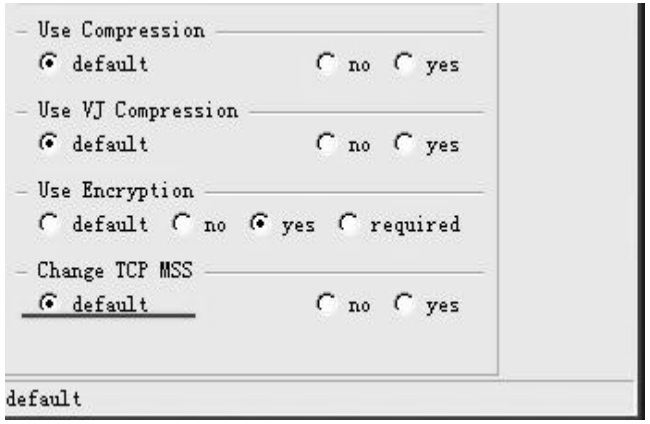


提升PPPoE服务器性能

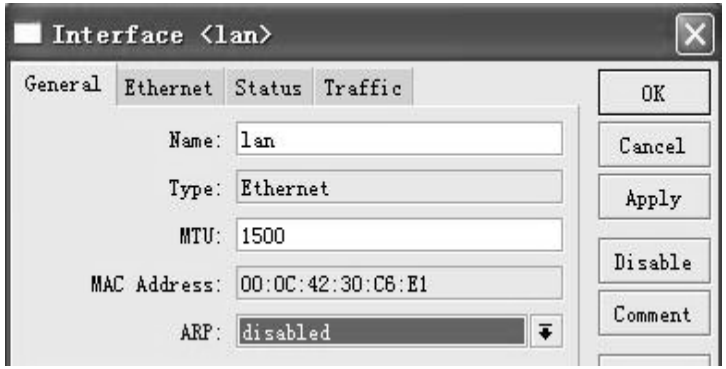


- 由于该案例中的PPPoE服务器，不需要做nat转换，所以我们可以关闭掉多余的功能
- /ip firewall connection tracking 选择关闭减小系统资源耗用
- 启用Radius服务器管理账号，因为本地数据库被设计为100-150个账号的管理。

提升PPPoE服务器性能



- 设置/ppp profile的change tcp mss为default或者no



- 没有以太网连接需要的情况下，可以关闭PPPoE Server接口的arp功能，减小开销。

VLAN

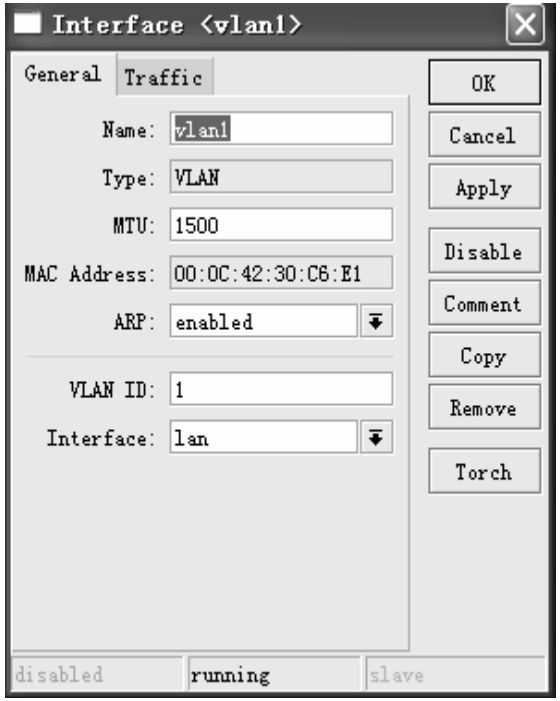
- 限制广播域。广播域被限制在一个VLAN内，减少了多余广播风暴，提高了网络处理能力。

- 增强局域网的安全性。不同VLAN内的报文在传输时是相互隔离的，即一个VLAN内的用户不能和其它VLAN内的用户直接通信，如果不同VLAN要进行通信，则需要通过路由器或三层交换机等三层设备。

- 灵活构建虚拟工作组。用VLAN可以划分不同的用户到不同的工作组，网络构建和维护更方便灵活。

VLAN配置

- 在RouterOS中的VLAN配置是指隧道VLAN，即trunk隧道协议
- 在/interface中，进入vlan或者在winbox中的interface里点加号添加一个vlan
- vlan中我们需要选择vlan的对应网卡，比如lan，然后配置vlan的ID，即trunk隧道的ID号。



- 通过隧道VLAN隔离后，可以在每一个VLAN下建立PPPoE服务

- 客户端通过建立在买个VLAN端口下作的PPPoE服务器拨号

网桥功能

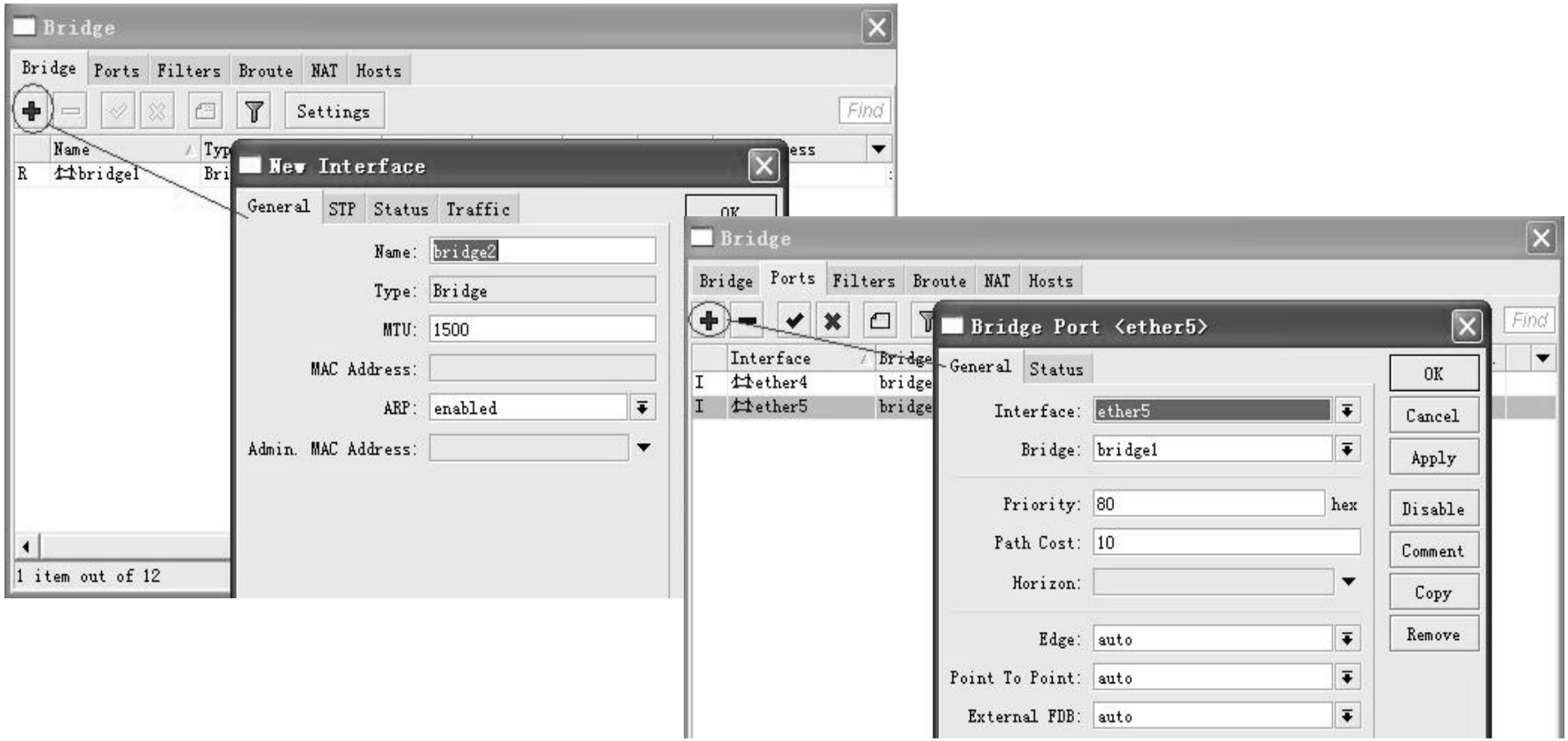
- 桥接基于OSI参考模型的第二层-数据链路层,工作类似于交换机，通过维护局域网中MAC地址列表，学习、存储和转发数据。

- 能透传三层IP数据包，透明连接两个IP网络。

- 常被用Wlan无线桥接、桥接流量控制、二层数据过滤和各种VPN隧道桥接。

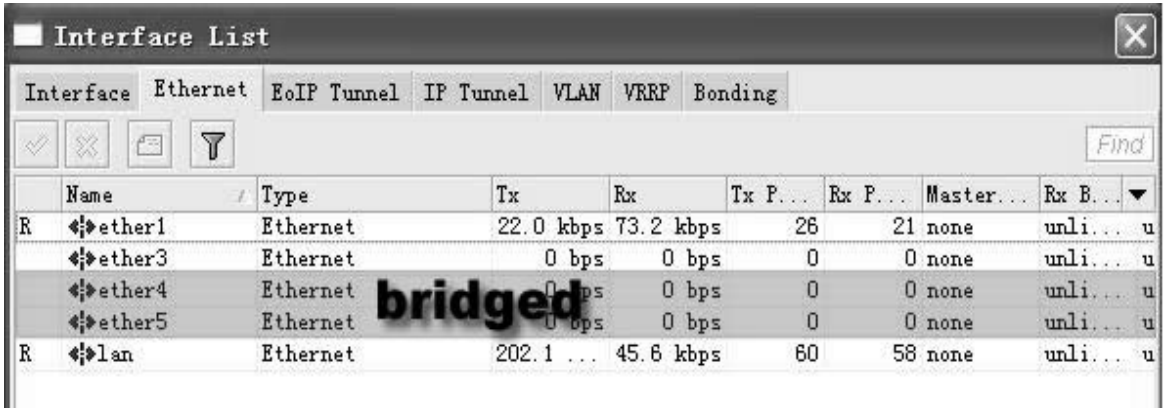
启用桥接功能

- 进入bridge中添加一个bridge接口，然后在bridge port中添加相应的网络接口到bridge中：



Bridge桥接

- 通过桥接后，将Ethernet 中的ether4和ether5设置为桥接模式；
- 我们可以通过添加或者删除管理桥接接口。
- 我们可以通过bridge host查看被桥接两个网络的MAC地址列表



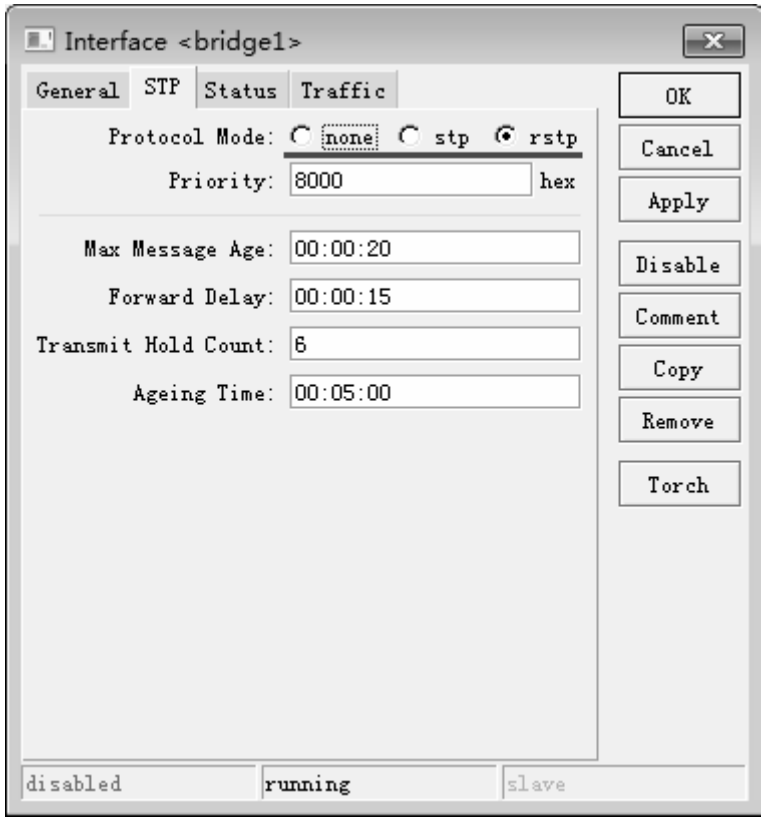
STP与RSTP

- STP（Spanning Tree Protocol）是生成树协议的英文缩写。该协议可应用于环路网络，通过一定的算法实现路径冗余，同时将环路网络修剪成无环路的树型网络，从而避免报文在环路网络中的增生和无限循环。但是它还是有缺点，STP协议的缺陷主要表现在收敛速度上。

- 为了解决STP协议的这个缺陷，在世纪之初IEEE推出了802.1w标准，作为对802.1D标准的补充。在IEEE 802.1w标准里定义了快速生成树协议RSTP（Rapid Spanning Tree Protocol）。RSTP协议在STP协议基础上做了三点重要改进，使得收敛速度快得多（最快1秒以内）。

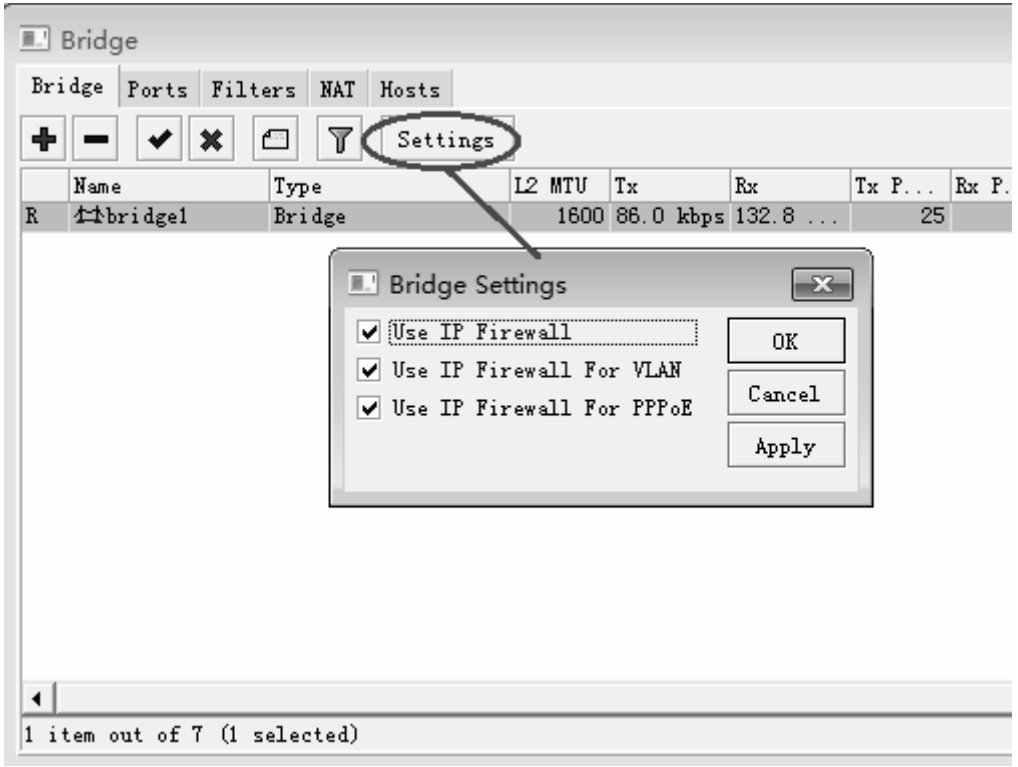
启用STP或RSTP

- 在添加bridge接口时，选择STP标签启用stp或者rstp。



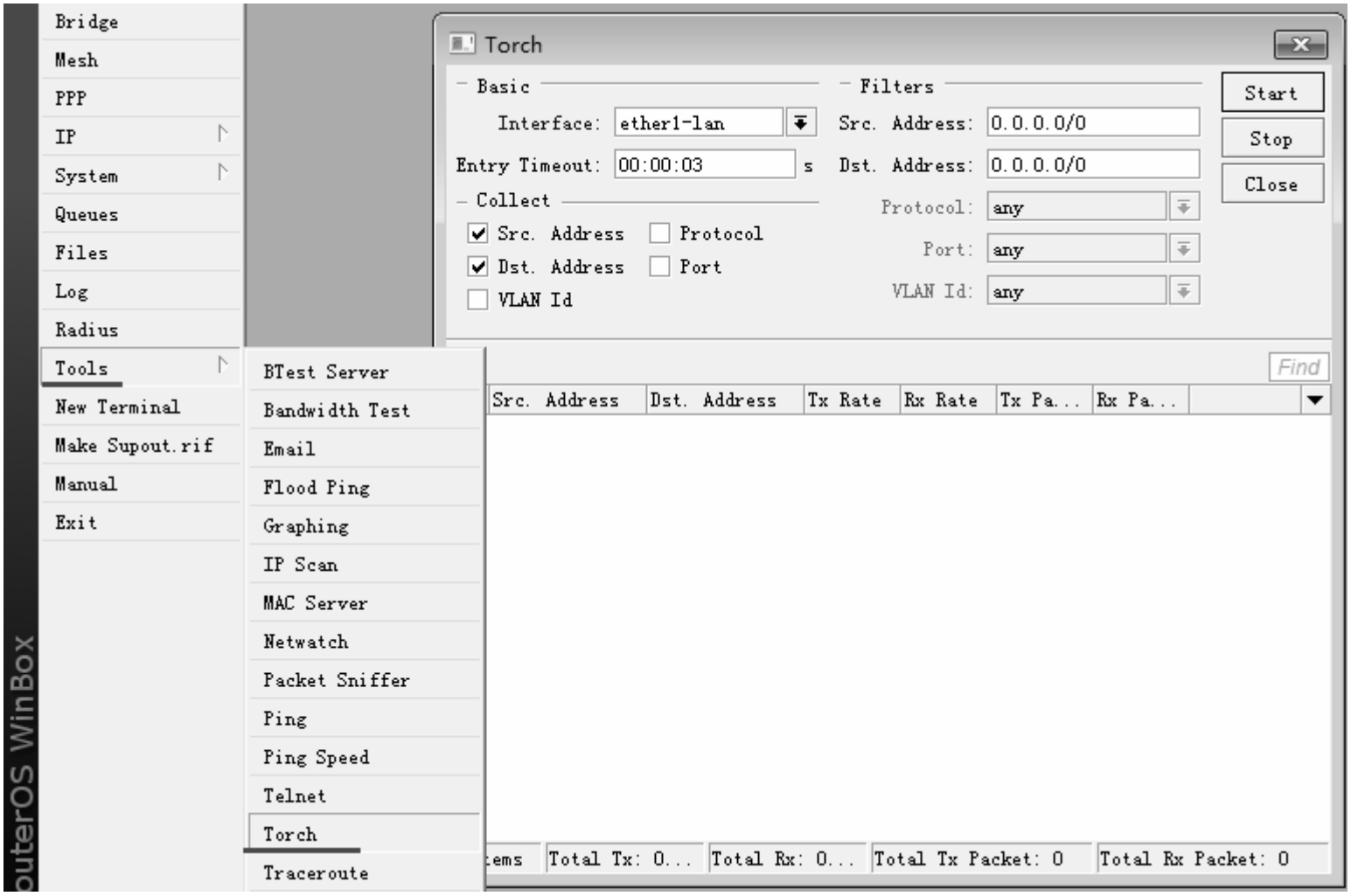
Use ip firewall

- 打开use-ip-firewall，即启用bridge的三层IP过滤和流控
- 当然你不想使用三层过滤，可以关闭掉，减小CPU耗用，提高转发率

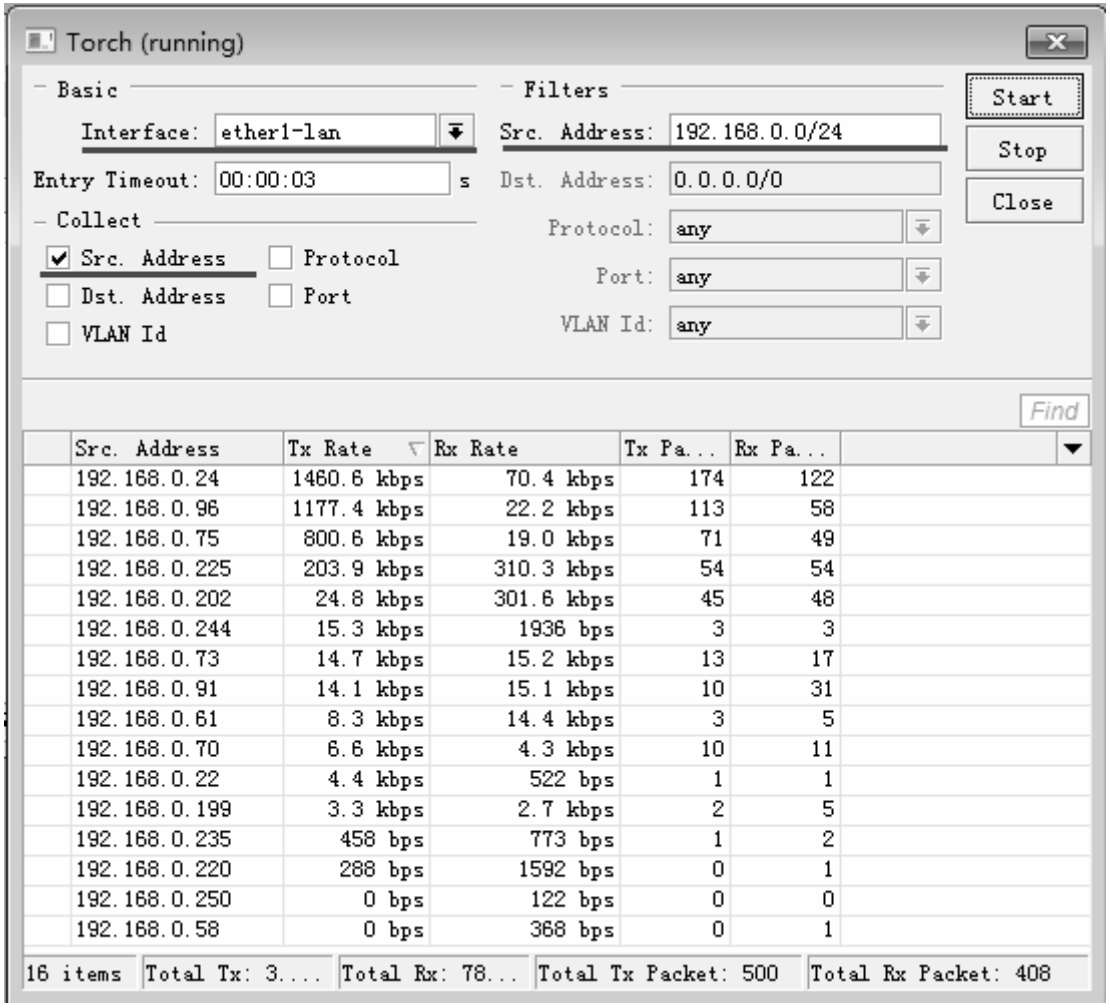


其他应用功能

Torch 实时抓包

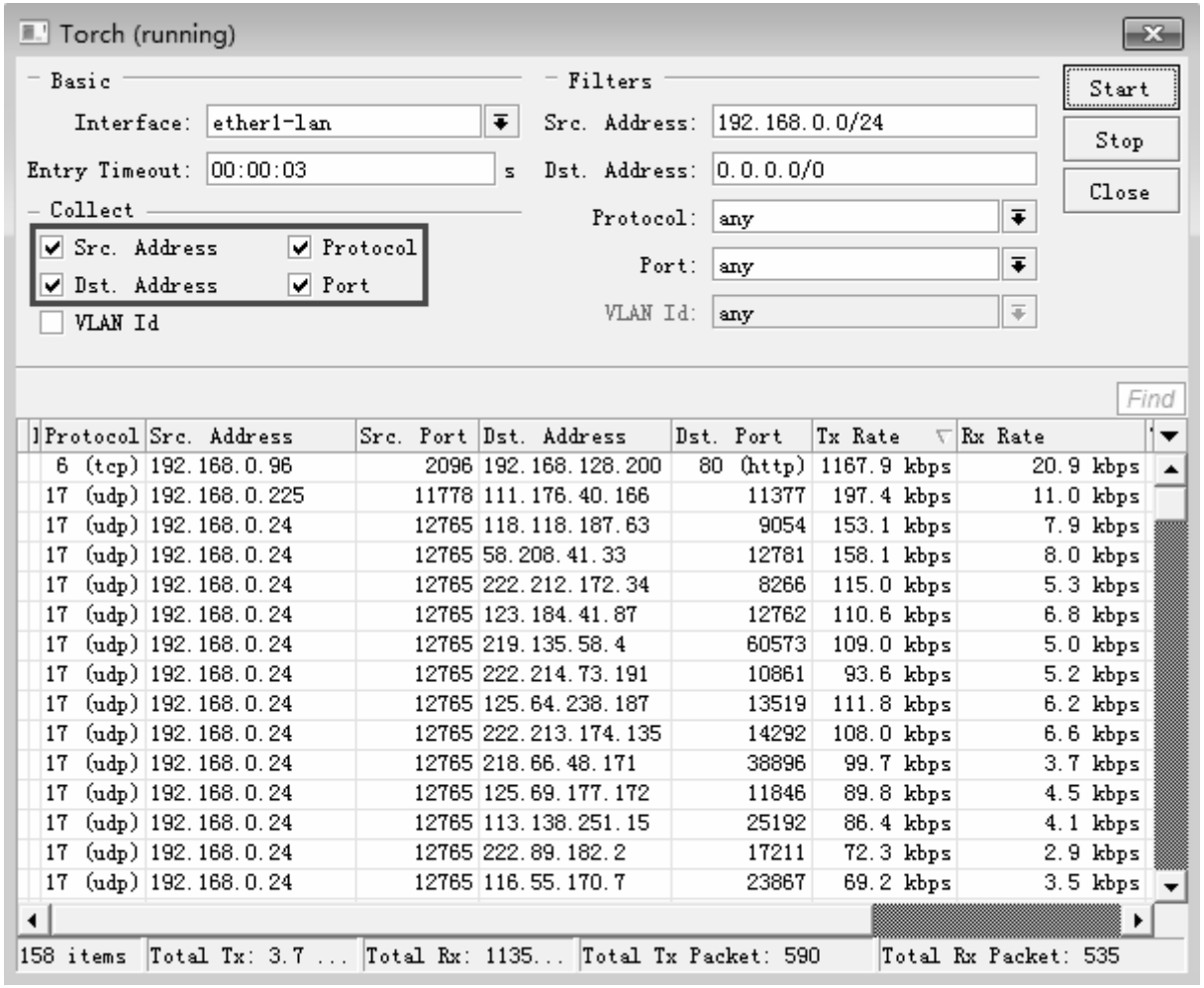

Bridge
Mesh
PPP
IP
System
Queues
Files
Log
Radius
Tools
New Terminal
Make Supout.rif
Manual
Exit
BTest Server
Bandwidth Test
Email
Flood Ping
Graphing
IP Scan
MAC Server
Netwatch
Packet Sniffer
Ping
Ping Speed
Telnet
Torch
Traceroute
outerOS WinBox
Torch
Basic
Interface: ether1-lan
Entry Timeout: 00:00:03 s
Collect
Src. Address
Dst. Address
VLAN Id
Protocol
Port
Filters
Src. Address: 0.0.0.0/0
Dst. Address: 0.0.0.0/0
Protocol: any
Port: any
VLAN Id: any
Start
Stop
Close
Find
Src. Address
Dst. Address
Tx Rate
Rx Rate
Tx Pa...
Rx Pa...
Total Tx: 0...
Total Rx: 0...
Total Tx Packet: 0
Total Rx Packet: 0

Torch 实时抓包



- 监测内网地址的带宽使用情况
- 查看每台主机的实时带宽耗用
- 发现异常客户机，及时分析处理

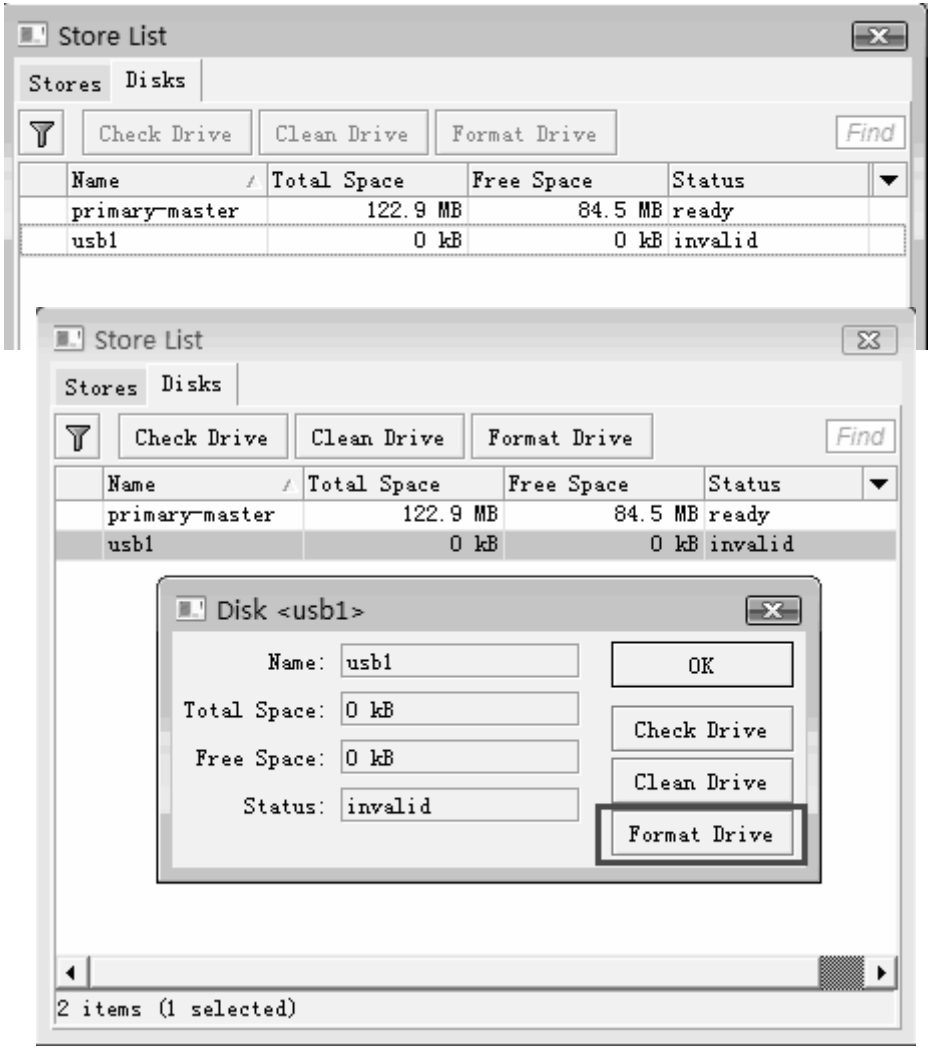
Torch 实时抓包



- 对每台主机的端口和协议占用监测
- 可用于游戏端口和协议分析

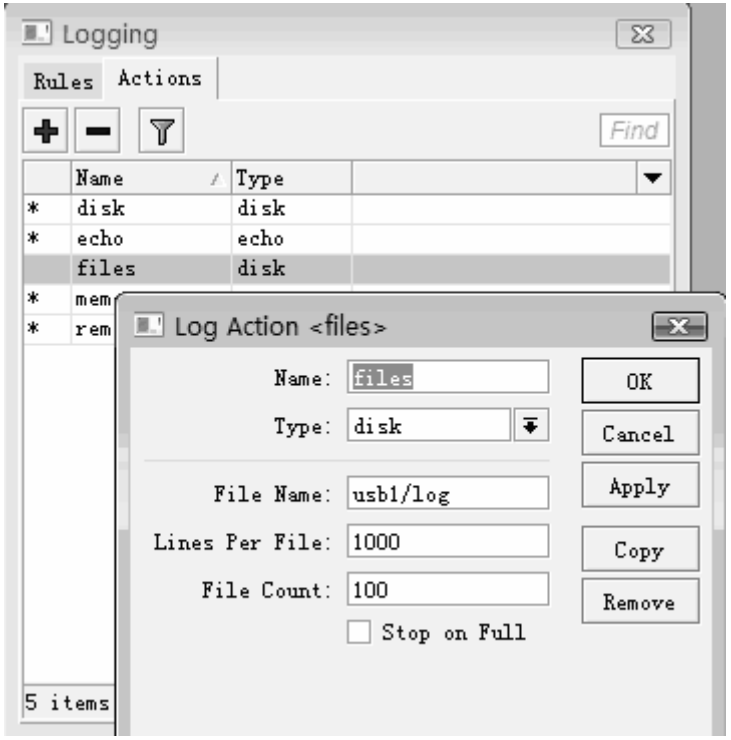
Store扩展存储


IP
System
Queues
Files
Log
Radius
Tools
New Terminal
Make Supout.rif
Manual
Exit
Auto Upgrade
Certificates
Clock
Console
Drivers
Health
History
Identity
License
Logging
Packages
Password
Ports
Reboot
Resources
SNTP Client
Scheduler
Scripts
Shutdown
Stores
RouterOS WinBox


Store List
Stores
Disks
Check Drive
Clean Drive
Format Drive
Find
Name
Total Space
Free Space
Status
primary-master
122.9 MB
84.5 MB
ready
usb1
0 kB
0 kB
invalid
Store List
Stores
Disks
Check Drive
Clean Drive
Format Drive
Find
Name
Total Space
Free Space
Status
primary-master
122.9 MB
84.5 MB
ready
usb1
0 kB
0 kB
invalid
Disk <usb1>
Name:
usb1
Total Space:
0 kB
Free Space:
0 kB
Status:
invalid
OK
Check Drive
Clean Drive
Format Drive
2 items (1 selected)

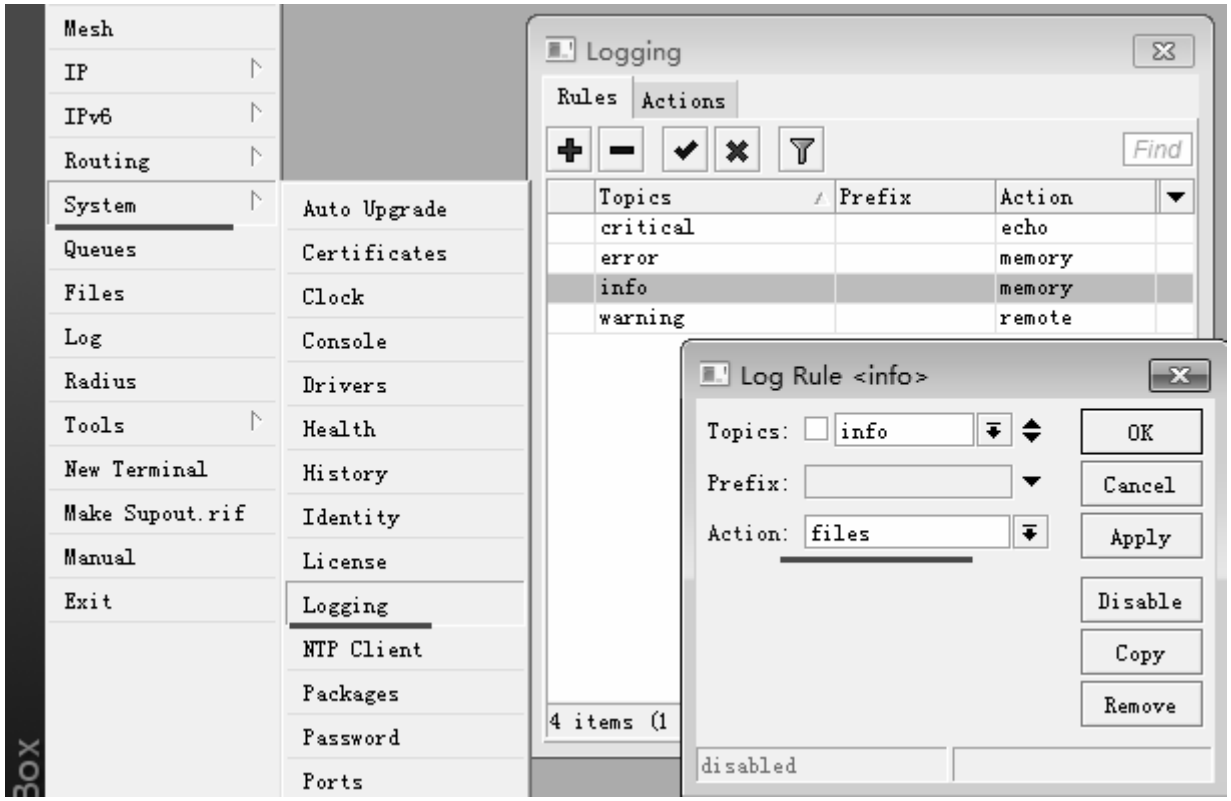
Log日志存储

- 我们到system logging里添加一个action操作，将日志存储到指定的U盘
- 路径是：usb1/log



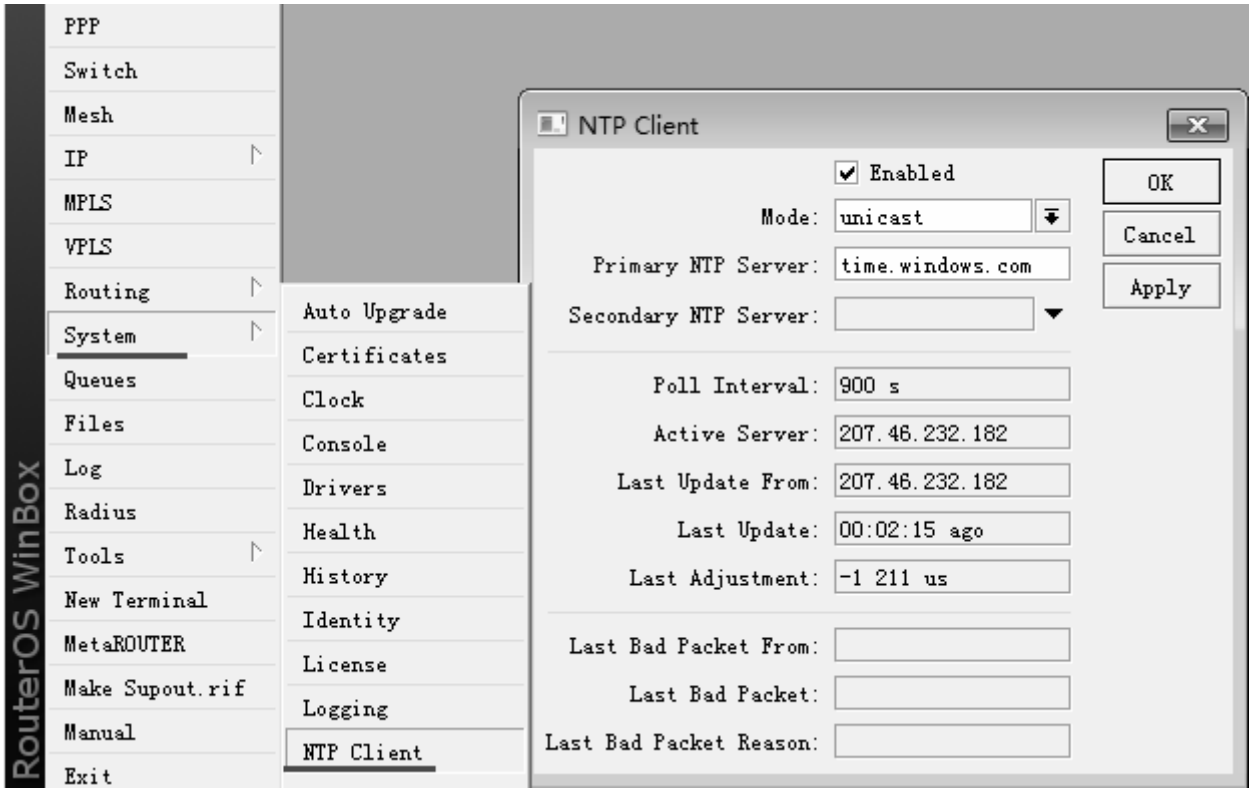
系统日志记录

- 进入system logging将系统信息日志保持到路由闪存中，便于查询记录：



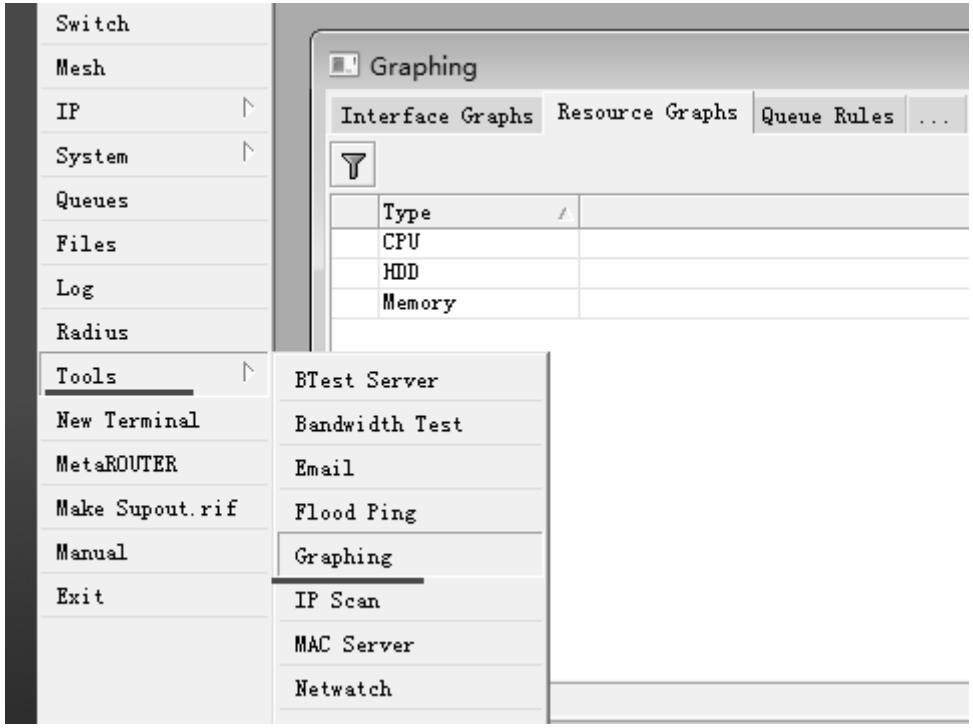
NTP网络对时

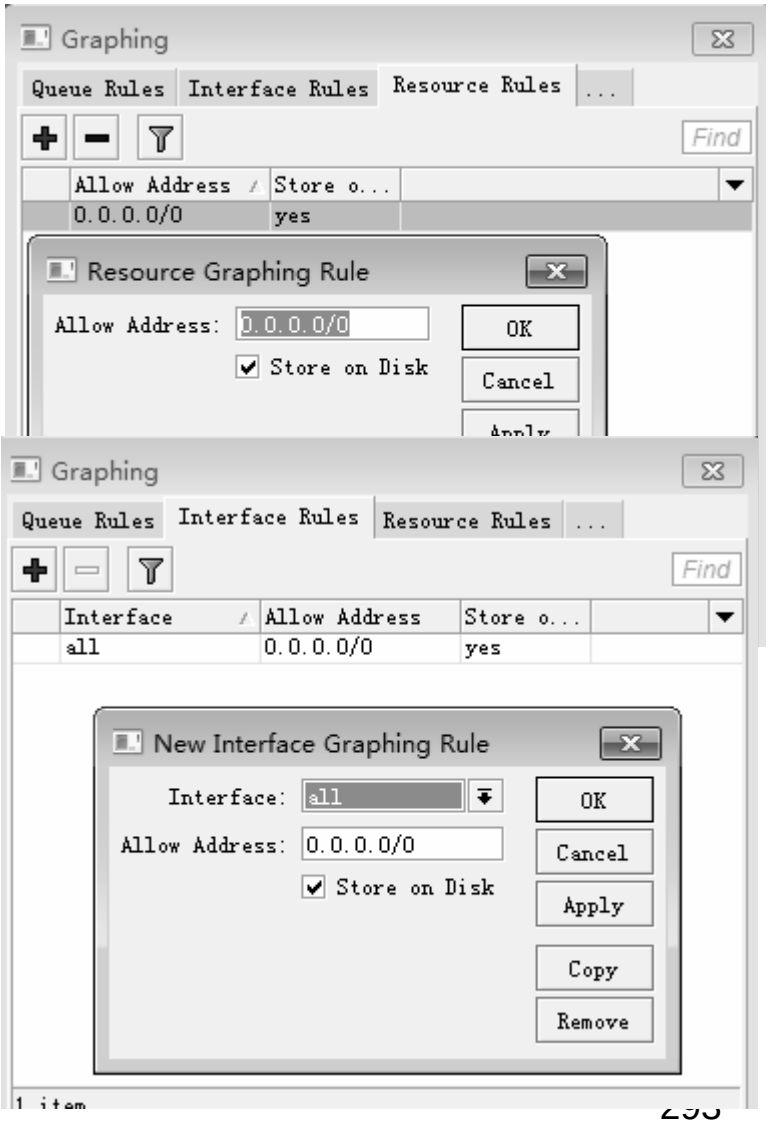
- 通在system NTP-client配置time.windows.com的NTP服务器，对NAT-1500和其他RouterOS设备进行网络对时，配置如下图:



流量图记录

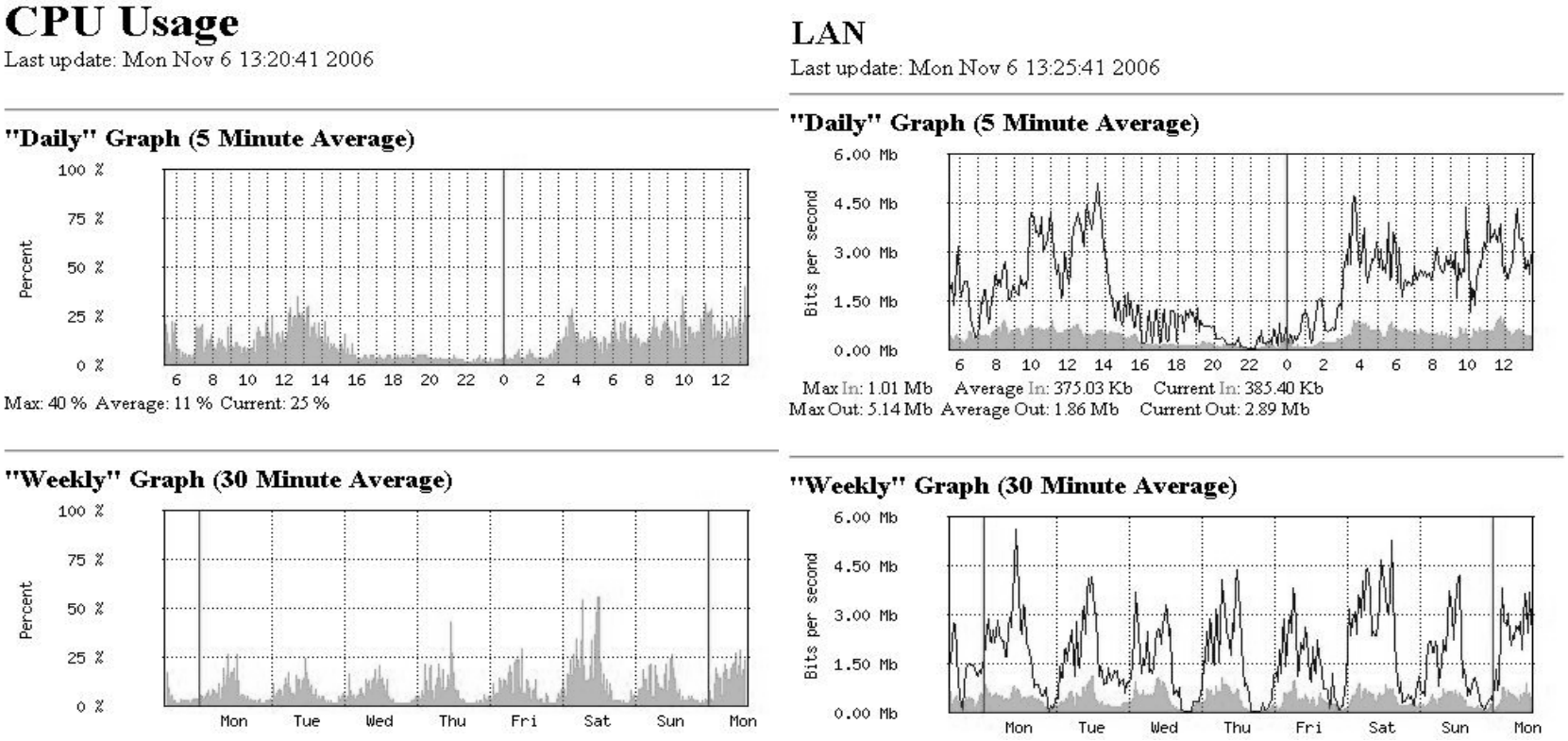
- 在tools Graphing添加记录规则





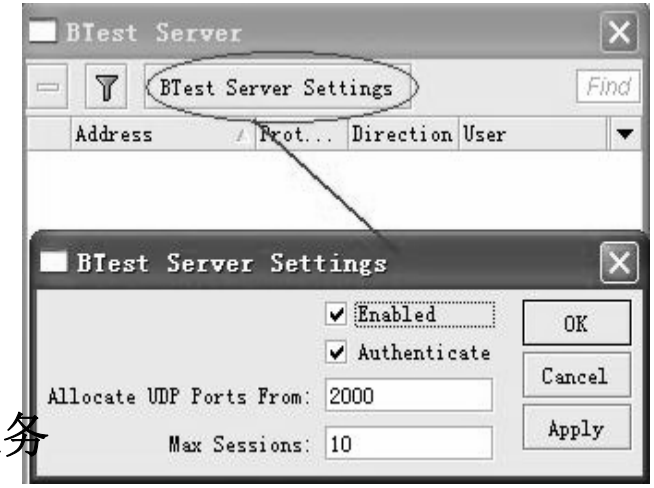
流量图记录

- 通选择lan口查看每天的运行情况,也可以通过网页访问记录的天、周、月、年的日志情况



Bandwidth-text带宽测试

- 带宽测试用于测试MikroTik路由器的吞吐量（有线或无线）和检测网络带宽瓶颈。操作路径/tool bandwidth-test
- 测试路由器吞吐量，需要三台路由器相链接：Bandwidth服务器，测试路由器（Testing Router）和Bandwidth客户端:

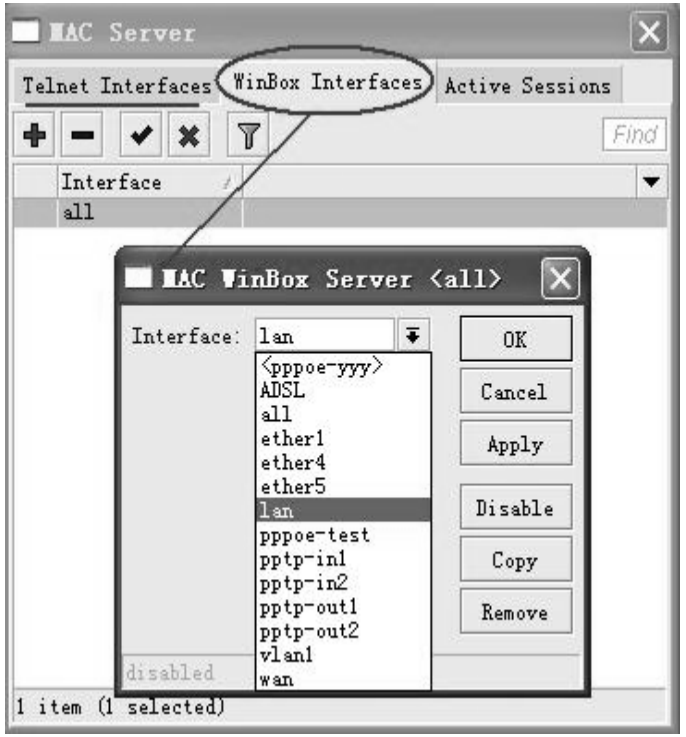


通过**bandwidth-server**启用和配置带宽测试服务

mac-server

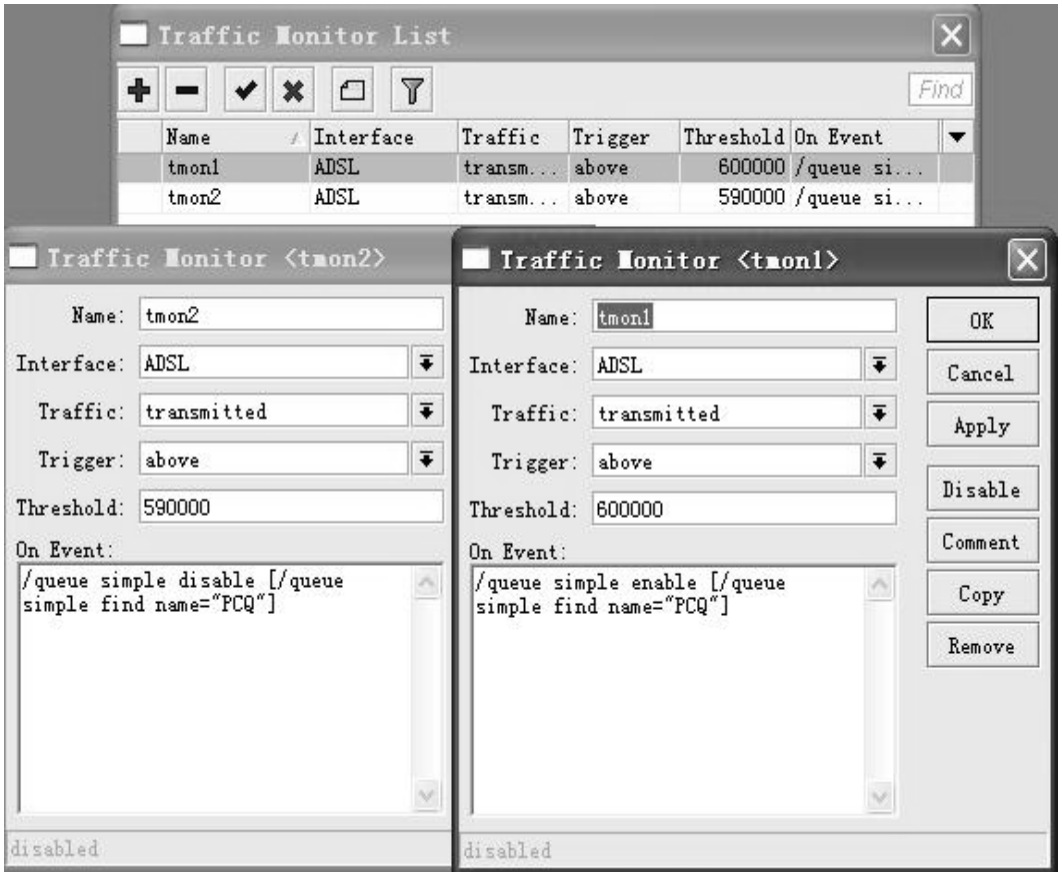
- 控制每个网卡接口的MAC登录，包括（winbox和其它MAC登录工具）

- 操作路径：/tool mac-server



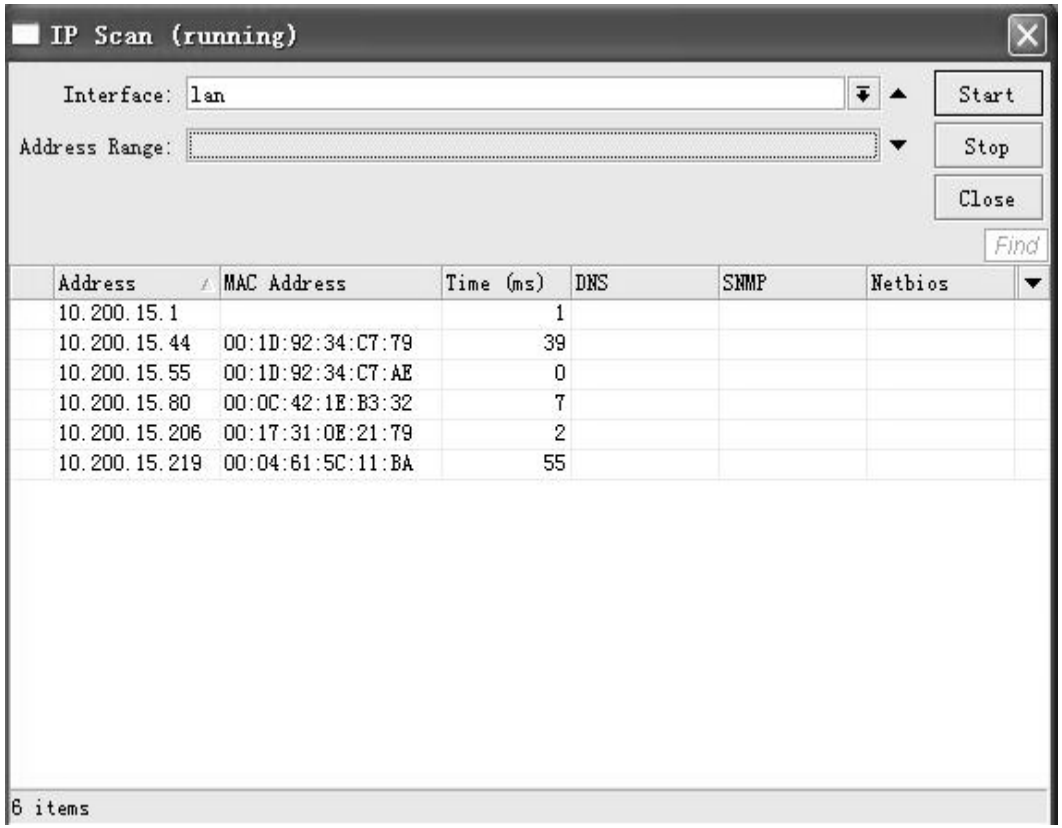
traffic-monitor

- 监控网卡流量，并通过脚本控制事件发生
- 路径：/tool traffic-monitor
- 这里是一个根据流量大小启用和禁用PCQ规则的脚本事例。



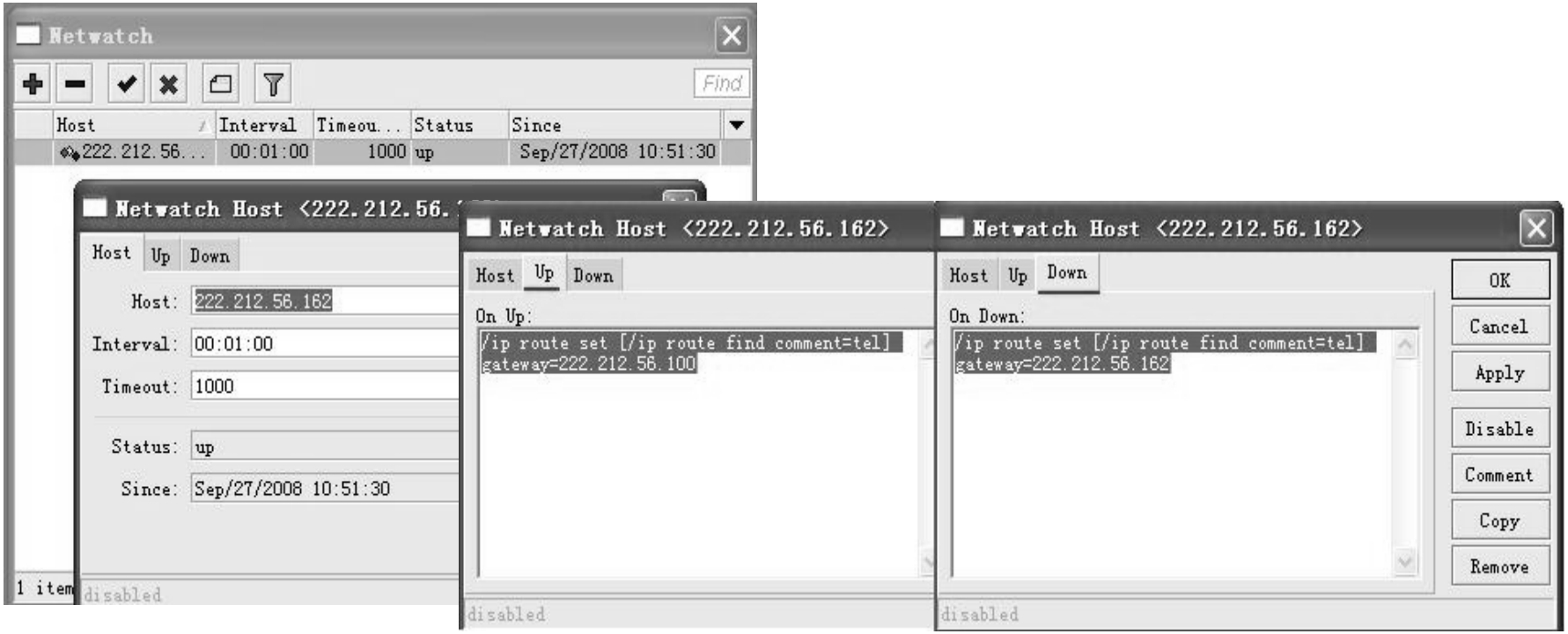
IP Scan

- 通过IP Scan搜索对应网卡上的所有IP和MAC地址
- 路径/tool ip-scan



netwatch

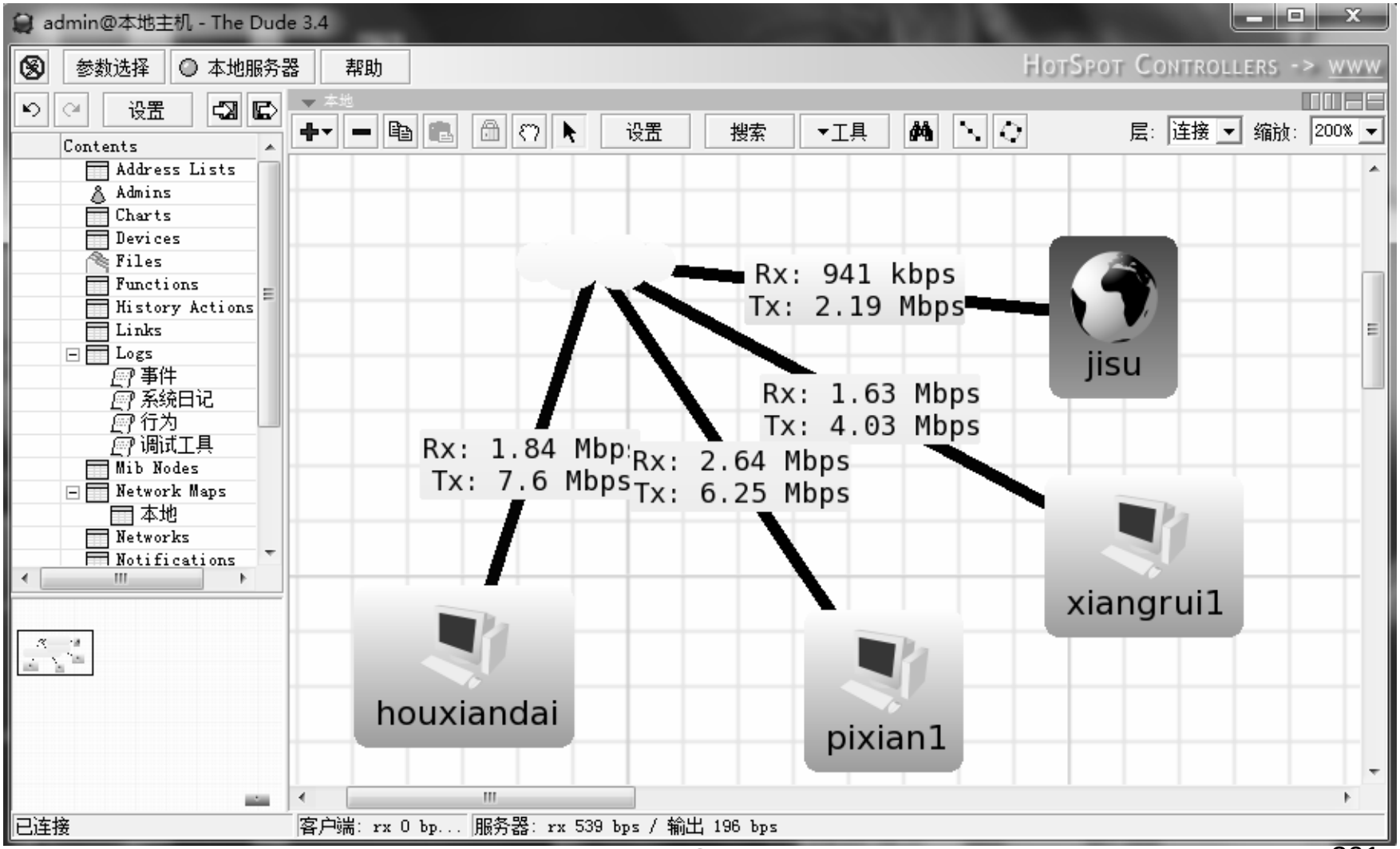
- 通过ICMP协议，监测网络主机IP，通常用于网络主机监测，并做出相应的脚本指令。
- 操作路径：/tool netwatch



The Dude网络管理器

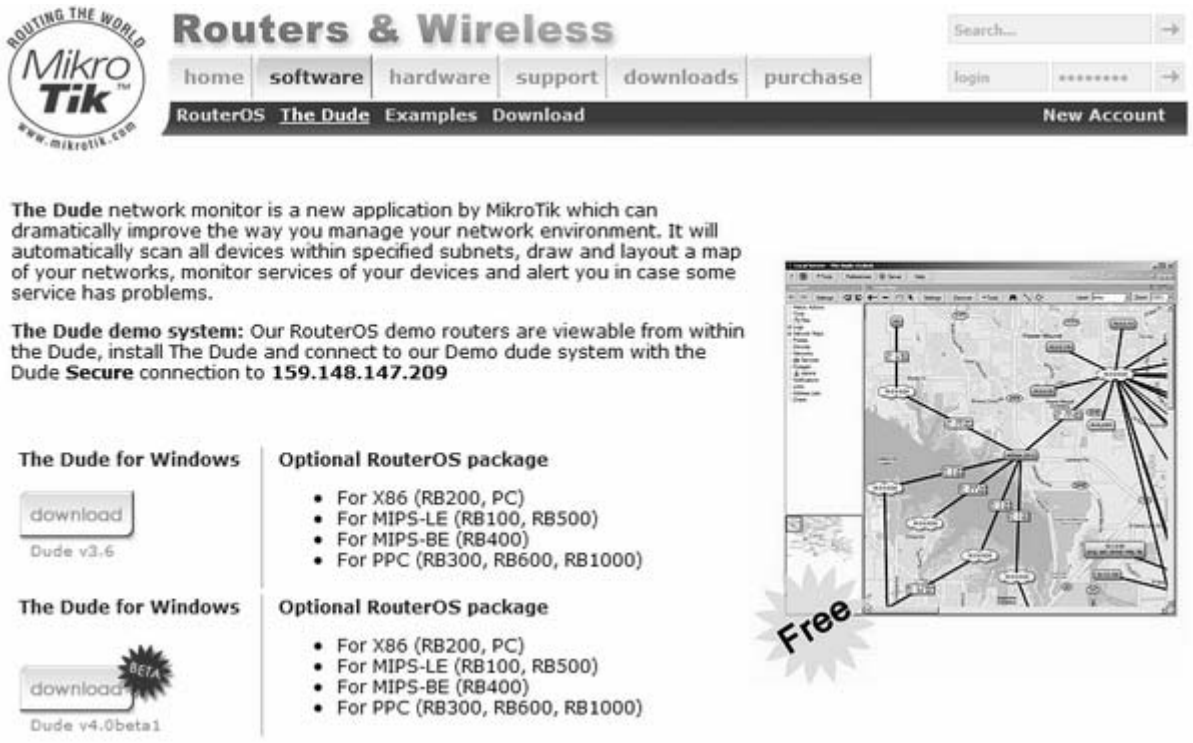
- 免费的网络管理器软件
- 自动网络搜索和布置网络拓扑图；
- 探测任何类型的网络设备；
- 设备的连接监测和状态通知；
- 为设备提供SVG图标，支持用户图标和背景定义；
- 简单的安装与操作和日志系统；
- 允许你绘制你的网络拓扑图和添加需要定义网络设备；
- 支持SNMP, ICMP, DNS和TCP等协议等对设备的监视；
- 独特的连接不间断监视和图像显示功能；
- 设备管理可以通过远程管理工具直接进入；
- 支持远程Dude服务器和本地客户端；
- 能运行在Linux Wine环境、MacOS Darwine环境和Windows下。

The Dude网络监控软件



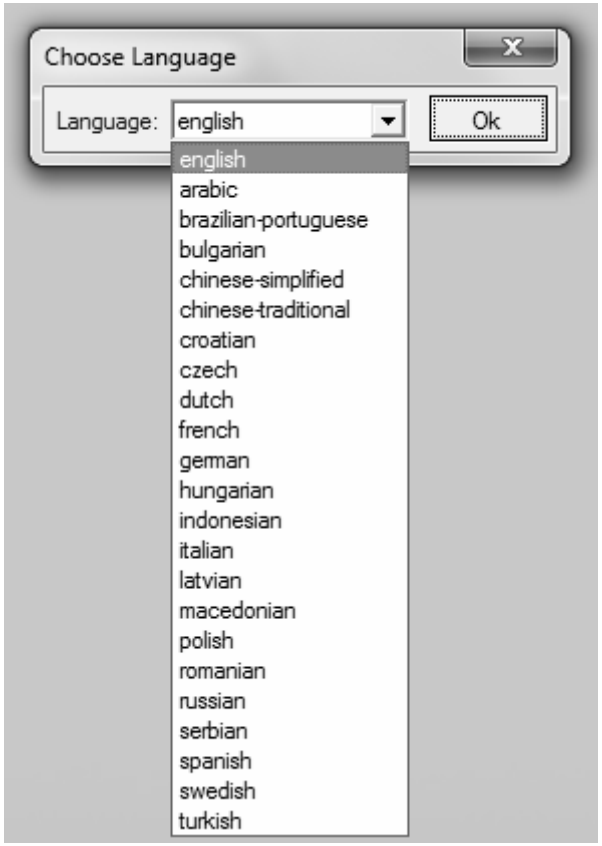
The Dude下载

- 可以到www.mikrotik.com下载最新的Dude软件



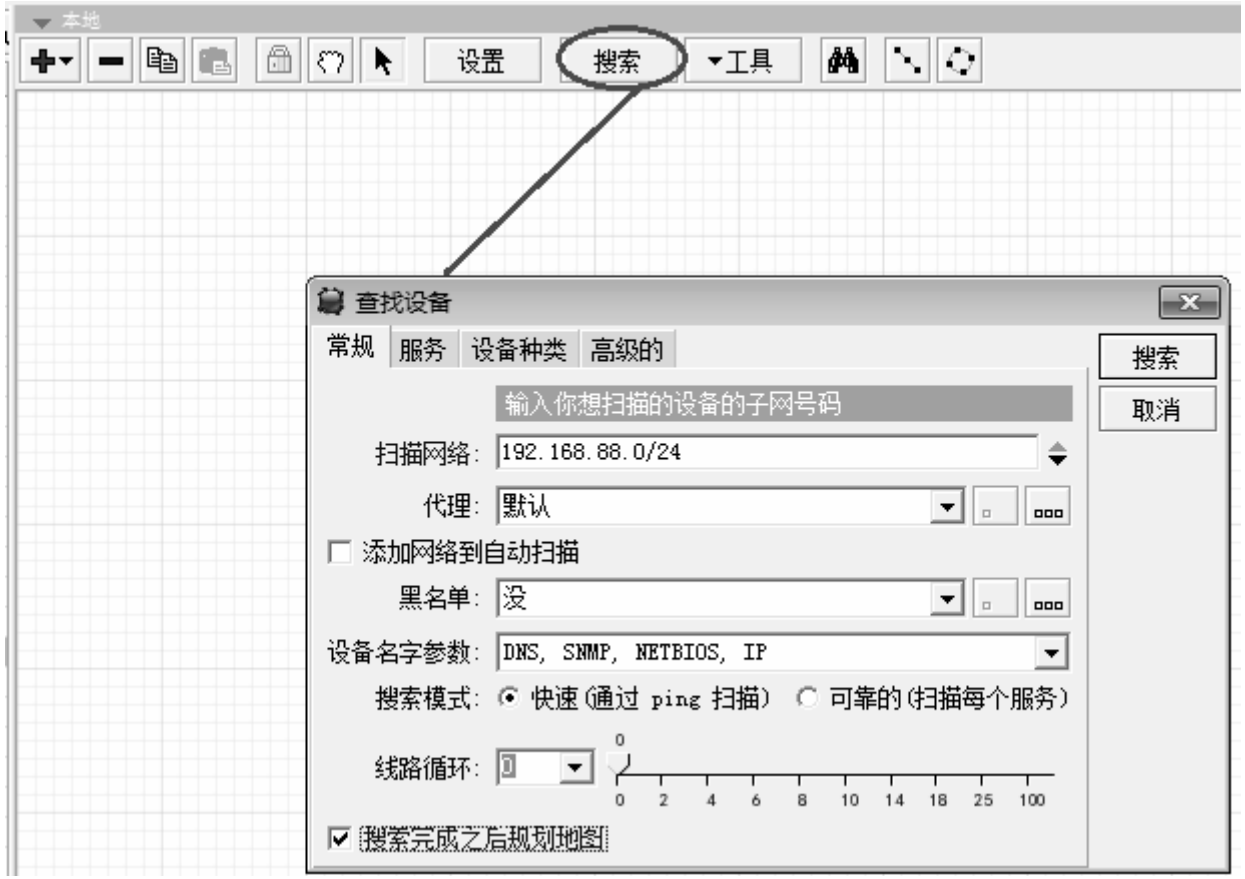
The Dude语言

- 安装后，第一次运行可以选择语言，这里有chinese-simplfied，即简体中文。

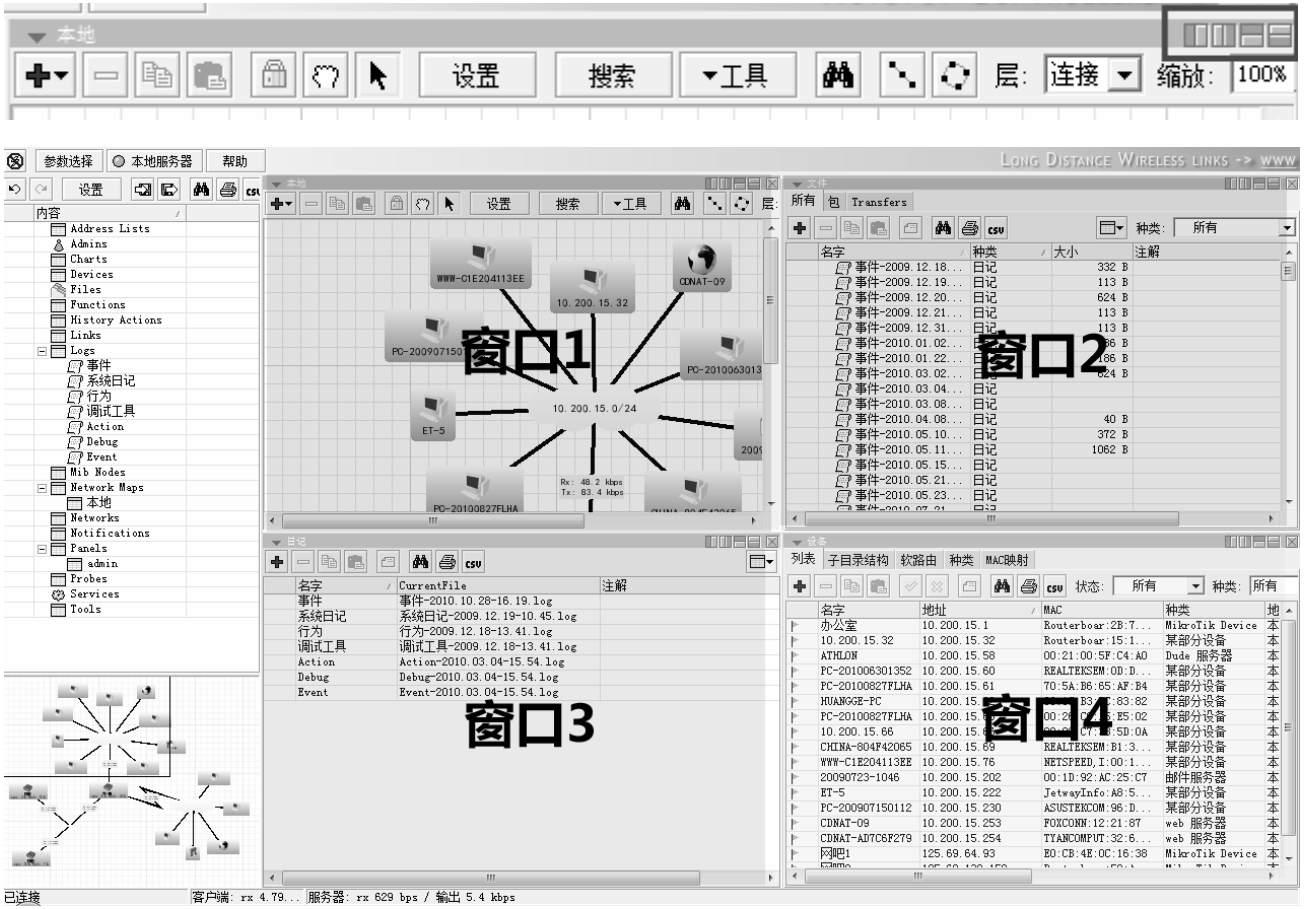


The Dude搜索

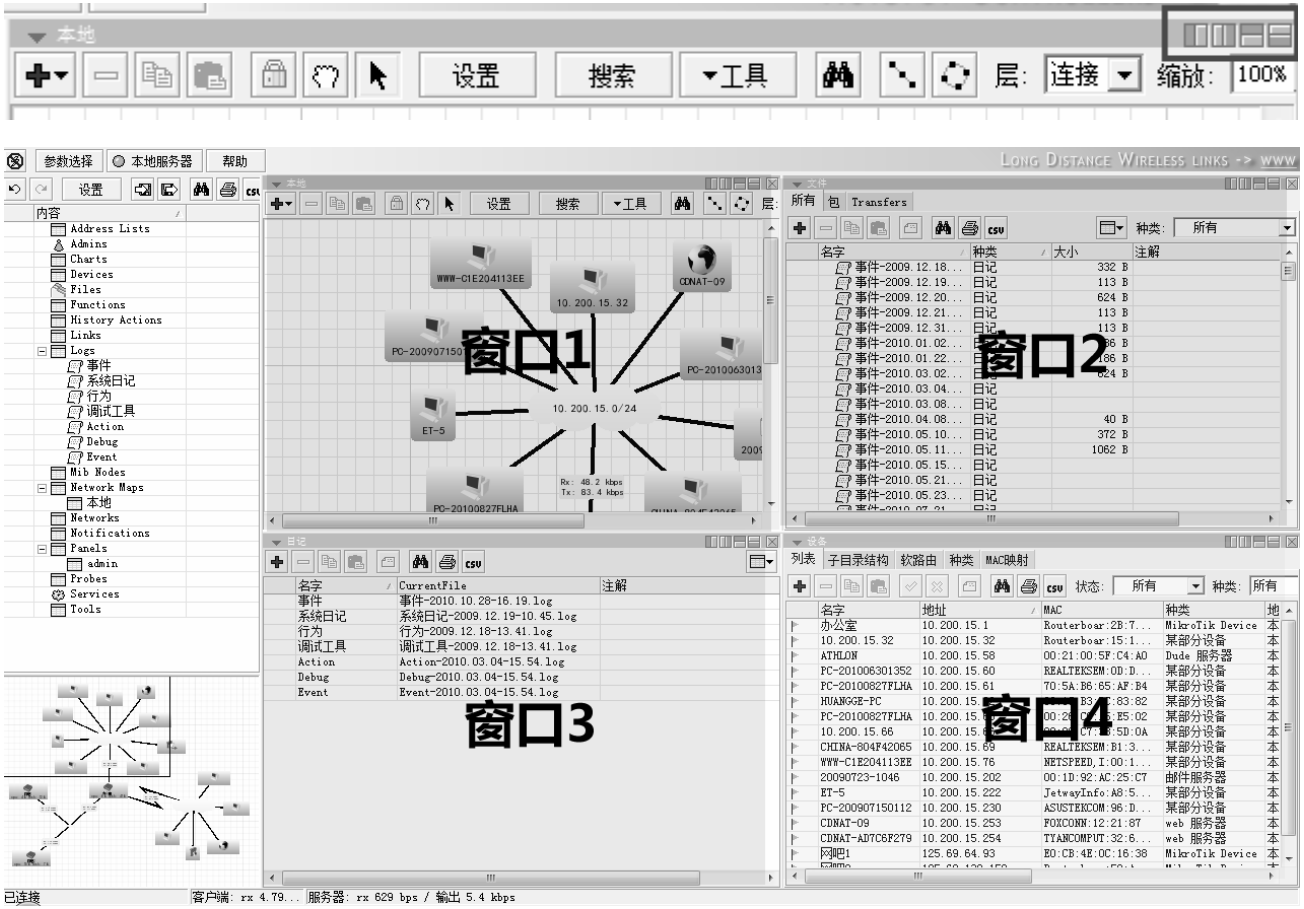
- 启动The Dude后，我们可以点搜索，并选择相应的参数查找局域网内的主机或者设备



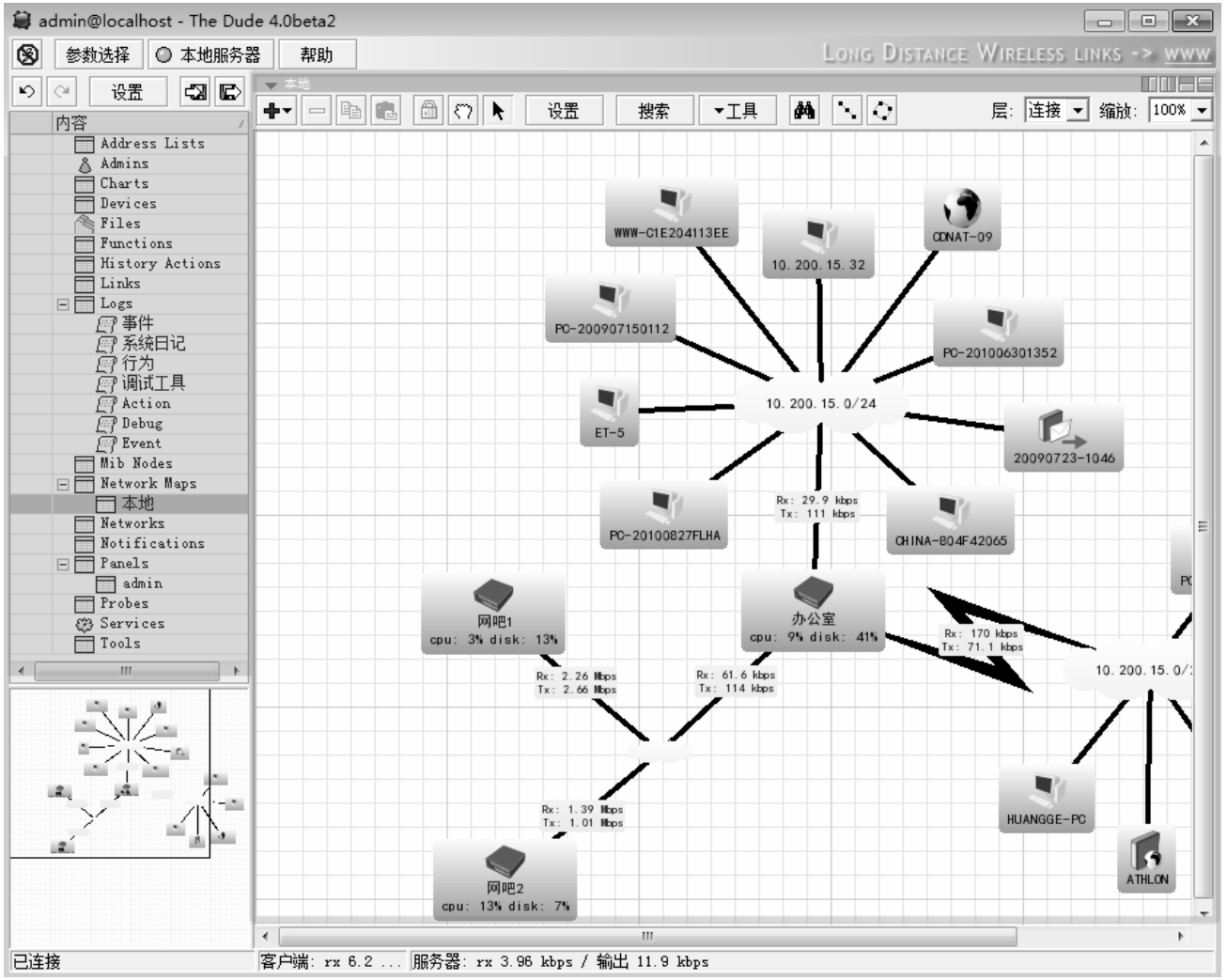
The Dude窗口


本地
设置
搜索
工具
层:
连接
缩放:
100%
参数选择
本地服务器
帮助
窗口1
窗口2
窗口3
窗口4
已连接
客户端: rx 4.79...
服务器: rx 629 bps / 输出 5.4 kbps

The Dude窗口


本地
设置
搜索
工具
层:
连接
缩放:
100%
参数选择
本地服务器
帮助
内容
Address Lists
Admins
Charts
Devices
Files
Functions
History Actions
Links
Logs
事件
系统日记
行为
调试工具
Action
Debug
Event
Mib Nodes
Network Maps
本地
Networks
Notifications
Panels
admin
Probes
Services
Tools
10.200.15.0/24
窗口1
窗口2
窗口3
窗口4
所有
包
Transfers
名字
种类
大小
注解
CurrentFile
列表
子目录结构
软路由
MAC映射
MAC
地址
已连接

The Dude菜单


admin@localhost - The Dude 4.0beta2
参数选择
本地服务器
帮助
LONG DISTANCE WIRELESS LINKS -> WWW
设置
搜索
工具
层: 连接
缩放: 100%
内容
Address Lists
Admins
Charts
Devices
Files
Functions
History Actions
Links
Logs
事件
系统日记
行为
调试工具
Action
Debug
Event
Mib Nodes
Network Maps
本地
Networks
Notifications
Panels
admin
Probes
Services
Tools
WWW-C1E204113EE
10.200.15.32
CDNAT-09
PC-200907150112
PC-201006301352
10.200.15.0/24
ET-5
20090723-1046
PC-20100827FLHA
Rx: 29.9 kbps
Tx: 111 kbps
CHINA-804F42065
网吧1
cpu: 3% disk: 13%
办公室
cpu: 9% disk: 41%
Rx: 170 kbps
Tx: 71.1 kbps
Rx: 2.26 Mbps
Tx: 2.66 Mbps
Rx: 61.6 kbps
Tx: 114 kbps
HUANGGE-PC
Rx: 1.39 Mbps
Tx: 1.01 Mbps
网吧2
cpu: 13% disk: 7%
ATHLON
已连接
客户端: rx 6.2 ...
服务器: rx 3.96 kbps / 输出 11.9 kbps

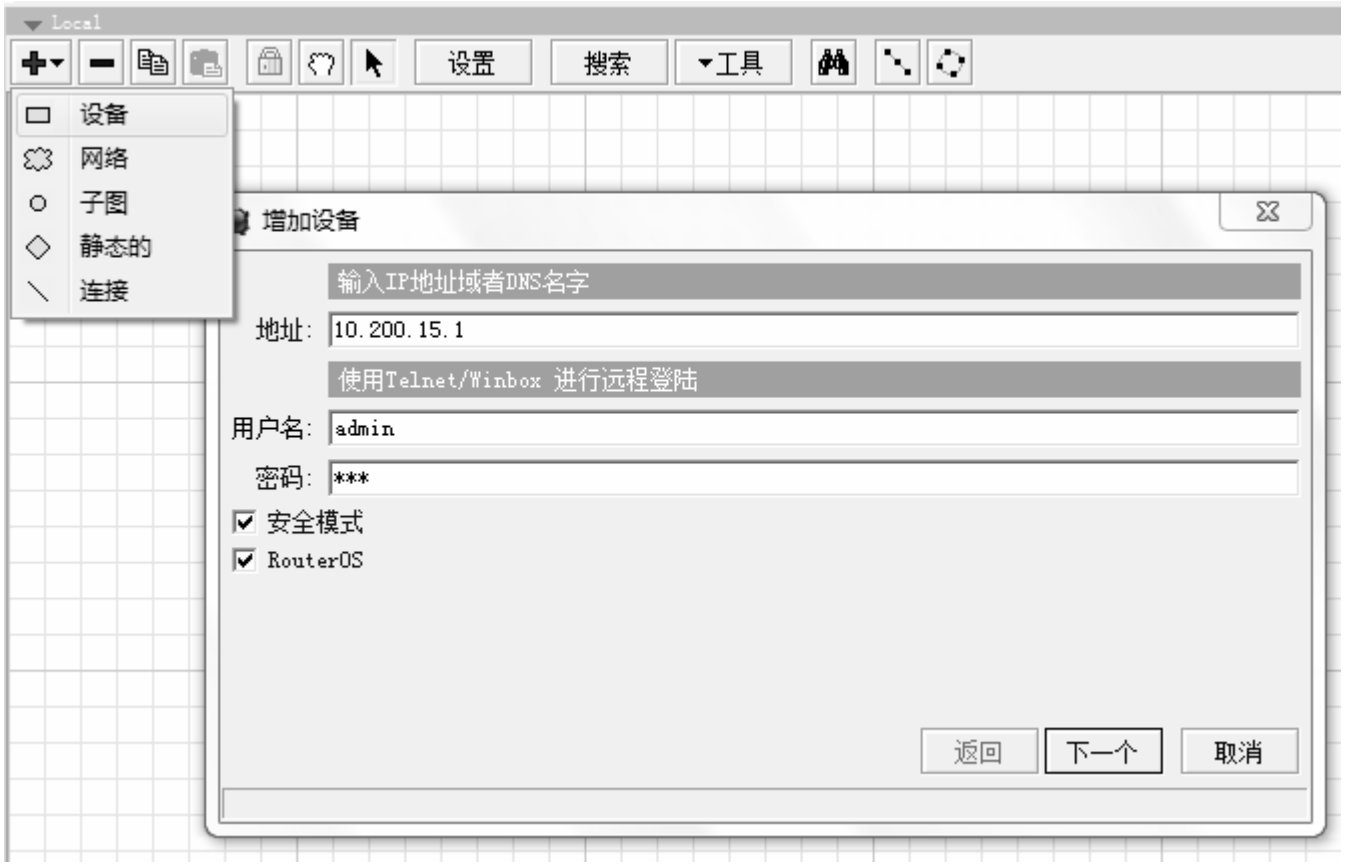
启用web服务器

- 通过IE访问The Dude服务器

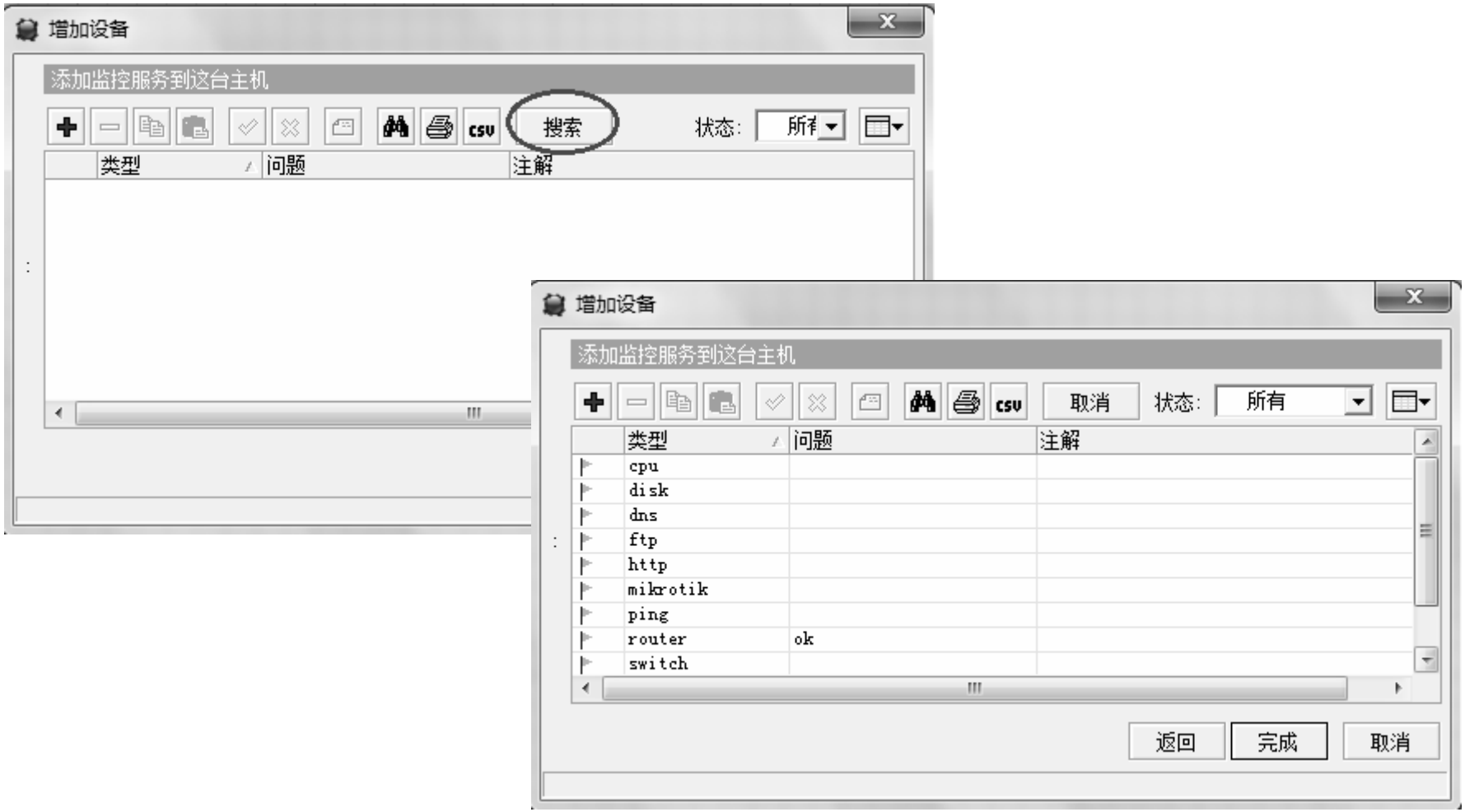
我们可以在IE浏览器里输入 http:\\IP地址:端口

添加设备

- 添加网络设备，例如**RouterOS**

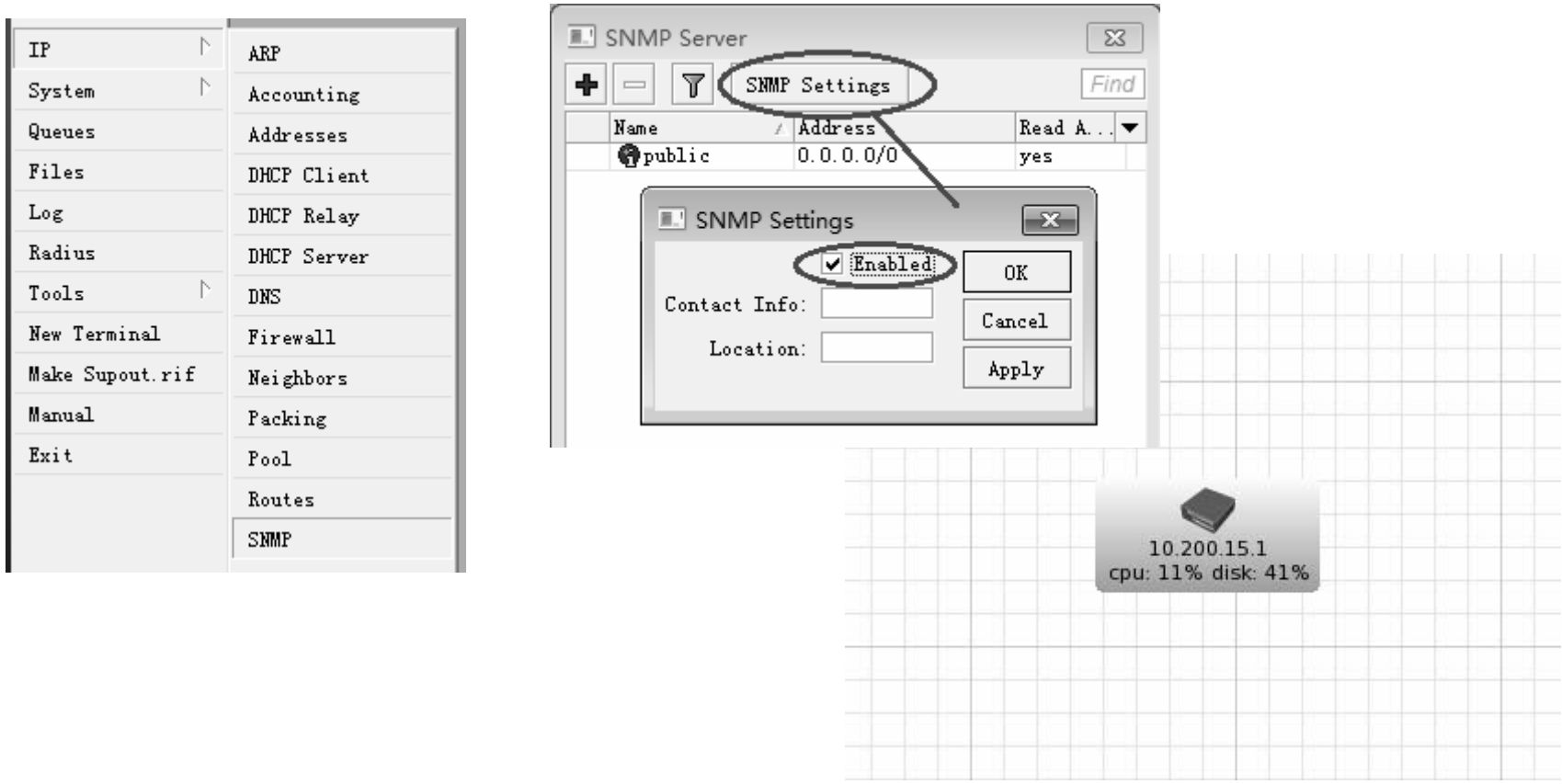


添加设备

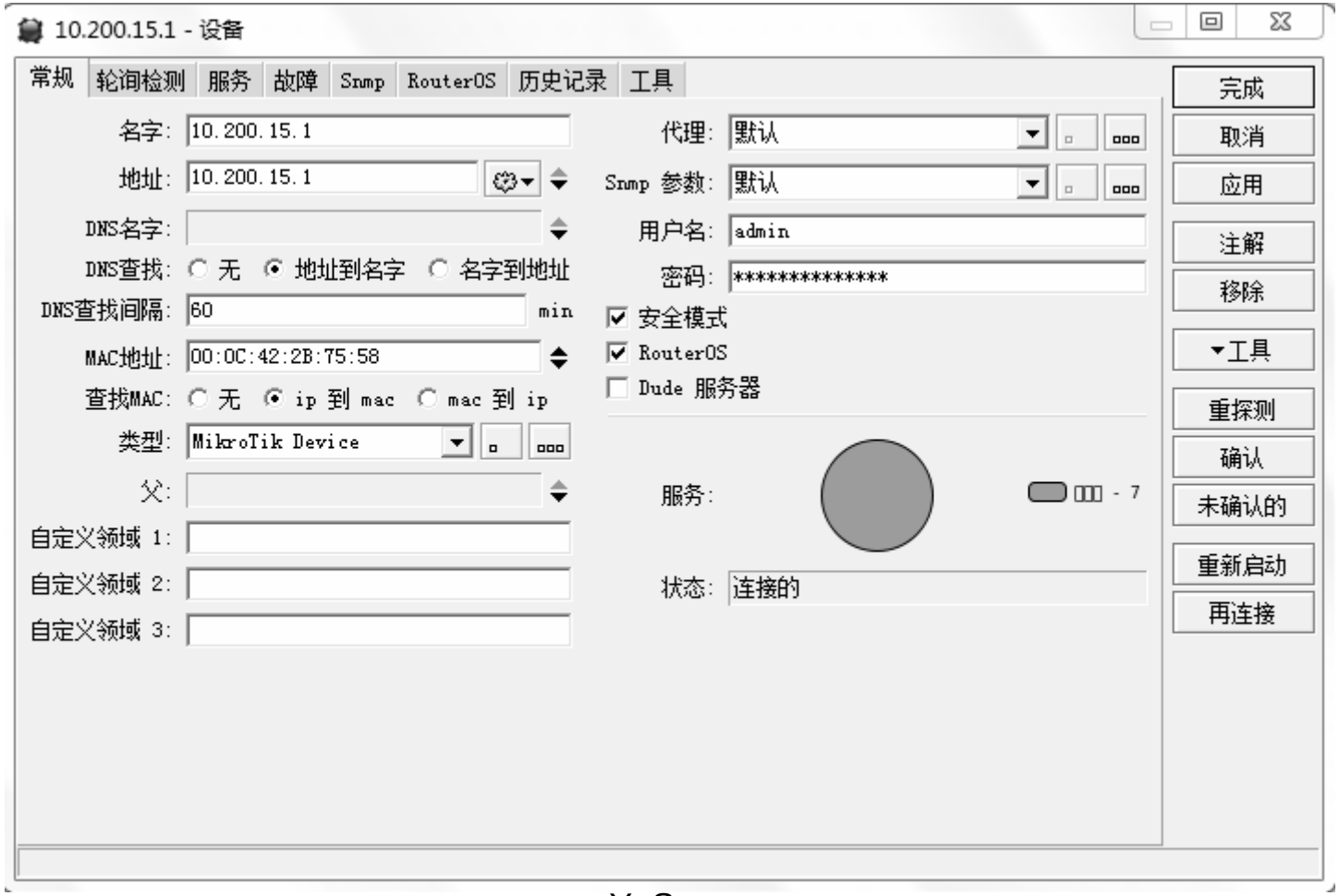

增加设备
添加监控服务到这台主机
搜索
状态:
所有
类型
问题
注解
增加设备
添加监控服务到这台主机
取消
状态:
所有
类型
问题
注解
cpu
disk
dns
ftp
http
mikrotik
ping
router
ok
switch
返回
完成
取消

打开SNMP

- 打开RouterOS的SNMP协议，监测CPU和硬盘情况



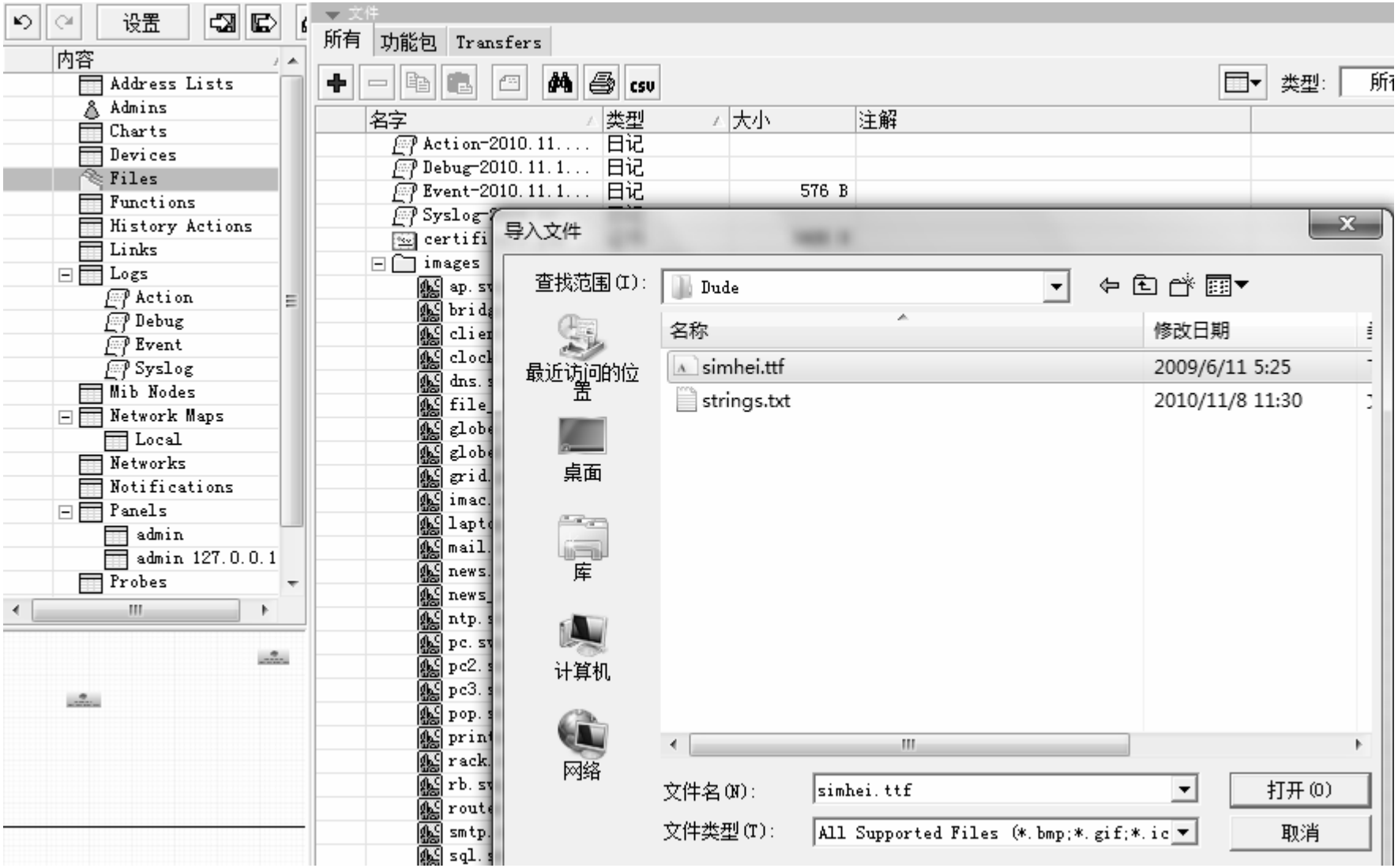
设备配置情况

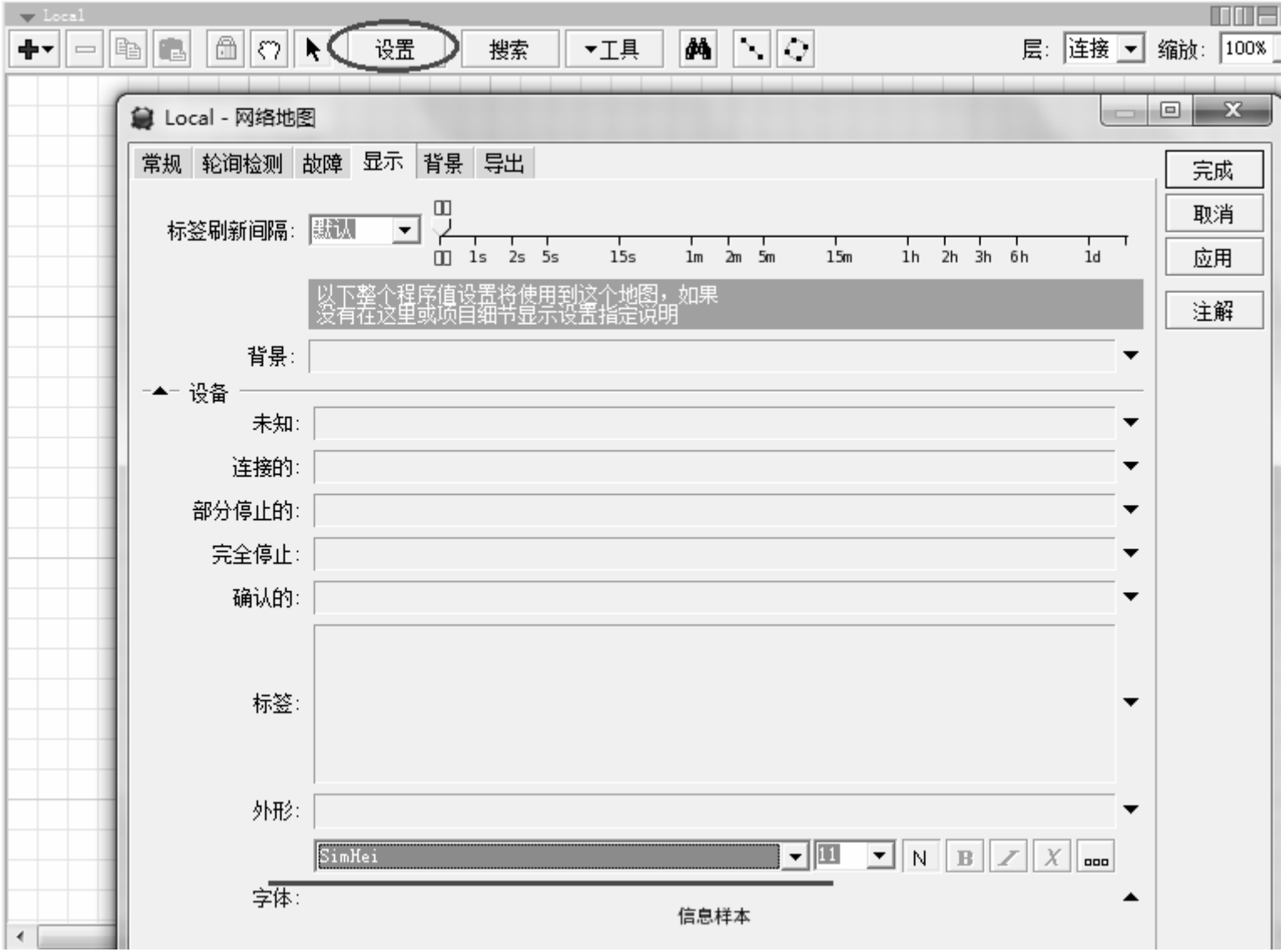

10.200.15.1 - 设备
常规
轮询检测
服务
故障
Snmp
RouterOS
历史记录
工具
名字: 10.200.15.1
地址: 10.200.15.1
DNS名字:
DNS查找: 无 地址到名字 名字到地址
DNS查找间隔: 60 min
MAC地址: 00:0C:42:2B:75:58
查找MAC: 无 ip 到 mac mac 到 ip
类型: MikroTik Device
父:
自定义领域 1:
自定义领域 2:
自定义领域 3:
代理: 默认
Snmp 参数: 默认
用户名: admin
密码: **********
安全模式
RouterOS
Dude 服务器
服务:
- 7
状态: 连接的
完成
取消
应用
注解
移除
工具
重探测
确认
未确认的
重新启动
再连接

字体问题

- The Dude对简体中文支持不完善，我们需要导入一个字体文件simhei.ttf，这个文件在任何windows电脑的系统文件下都可以找到

- 将这个文件导入The Dude的files菜单下

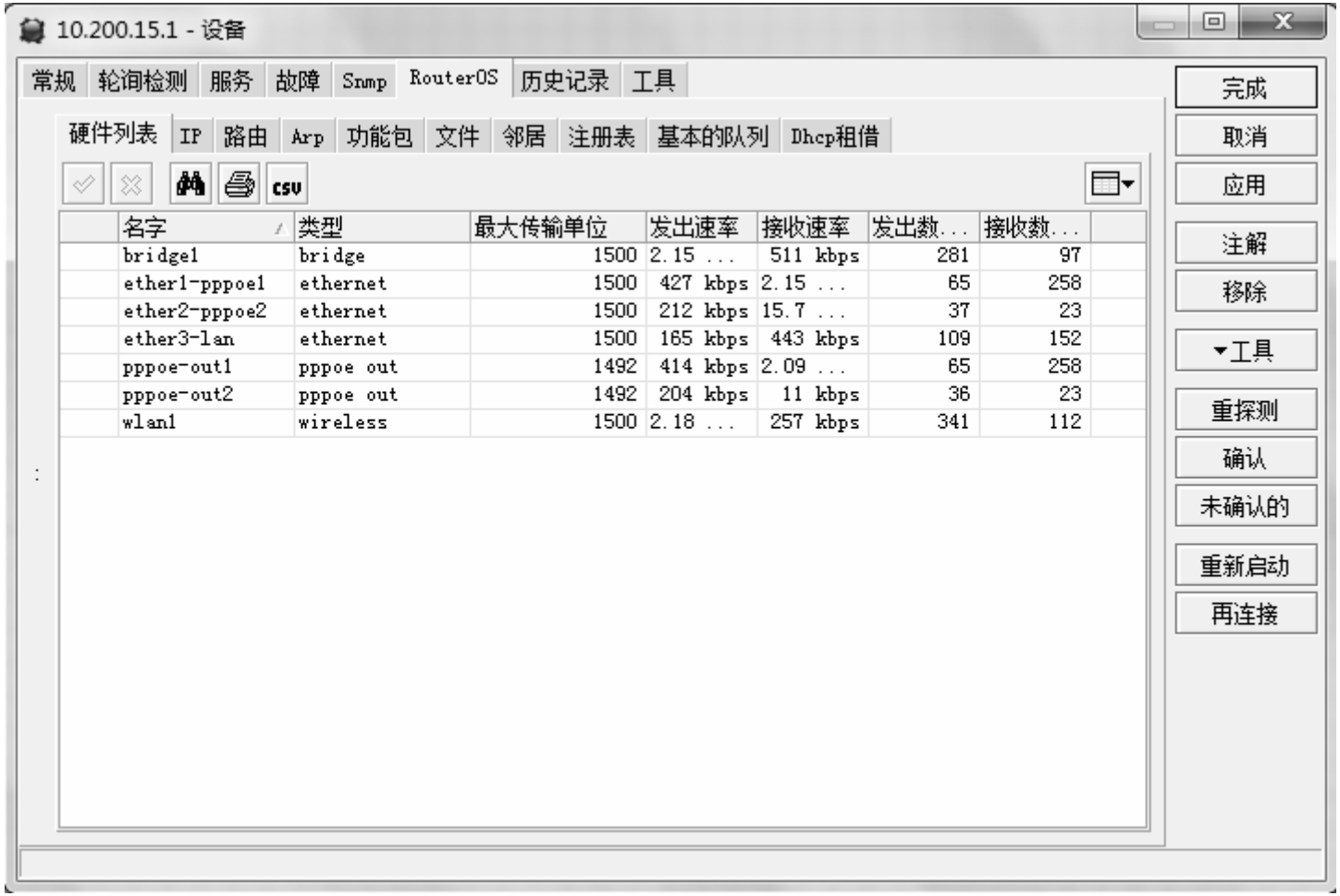

设置
内容
Address Lists
Admins
Charts
Devices
Files
Functions
History Actions
Links
Logs
Action
Debug
Event
Syslog
Mib Nodes
Network Maps
Local
Networks
Notifications
Panels
admin
admin 127.0.0.1
Probes
文件
所有
功能包
Transfers
csv
类型:
名字
类型
大小
注解
Action-2010.11....
Debug-2010.11.1...
Event-2010.11.1...
576 B
日记
导入文件
查找范围(I):
Dude
名称
修改日期
simhei.ttf
2009/6/11 5:25
strings.txt
2010/11/8 11:30
最近访问的位置
桌面
库
计算机
网络
文件名(N):
simhei.ttf
文件类型(T):
All Supported Files (*.bmp;*.gif;*.ic
打开(O)
取消


Local
设置
搜索
工具
层:
连接
缩放:
100%
Local - 网络地图
常规
轮询检测
故障
显示
背景
导出
完成
取消
应用
注解
标签刷新间隔:
默认
1s
2s
5s
15s
1m
2m
5m
15m
1h
2h
3h
6h
1d
以下整个程序值设置将使用到这个地图，如果
没有在这里或项目细节显示设置指定说明
背景:
设备
未知:
连接的:
部分停止的:
完全停止:
确认的:
标签:
外形:
SimHei
11
N
B
I
X
字体:
信息样本

字体修改

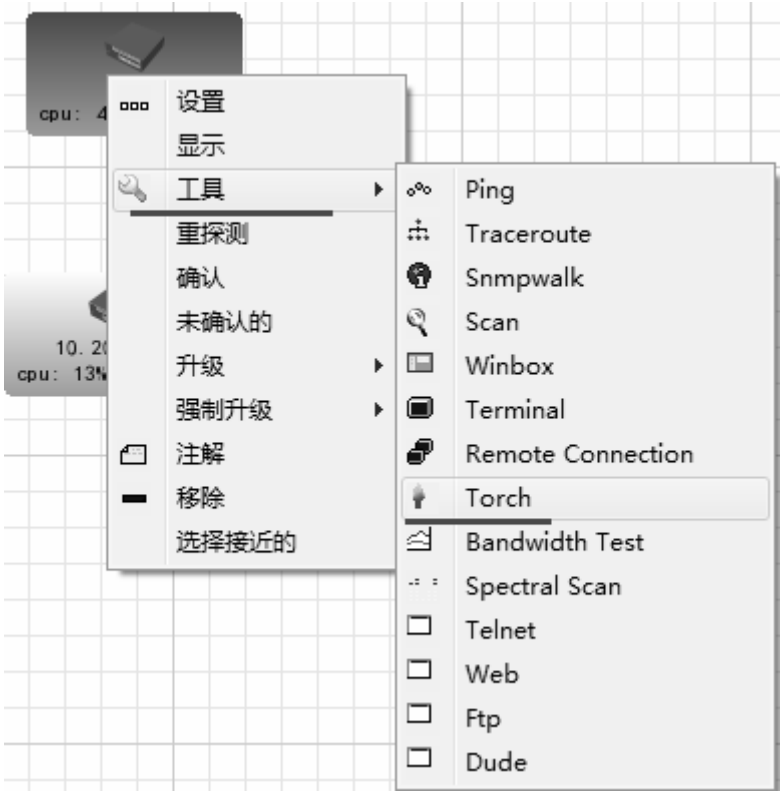
- 在The Dude中有多处字体需要修改，需要一一对应设置，才可以完全支持简体中文的现实
- 如下图划线部分的所有字体内容都需要修改

设备的RouterOS情况

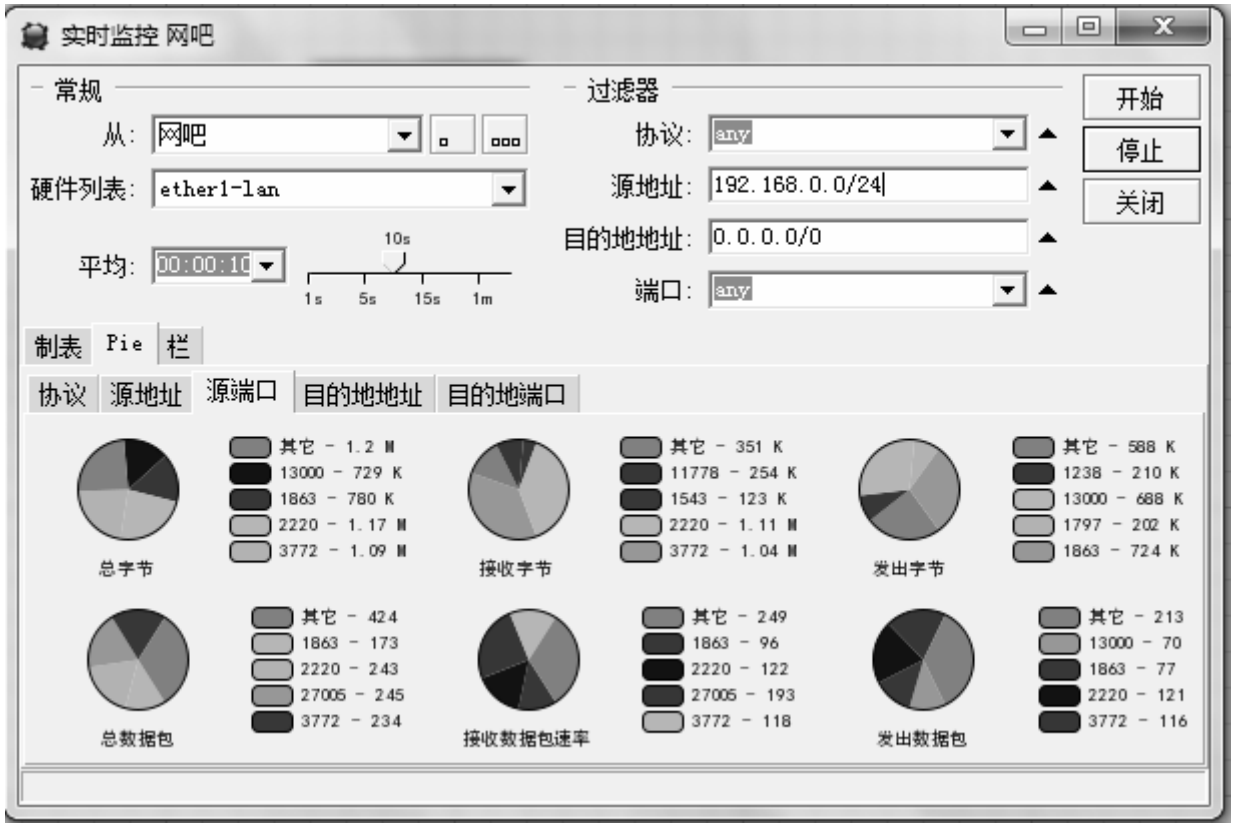


The Dude 实时监控

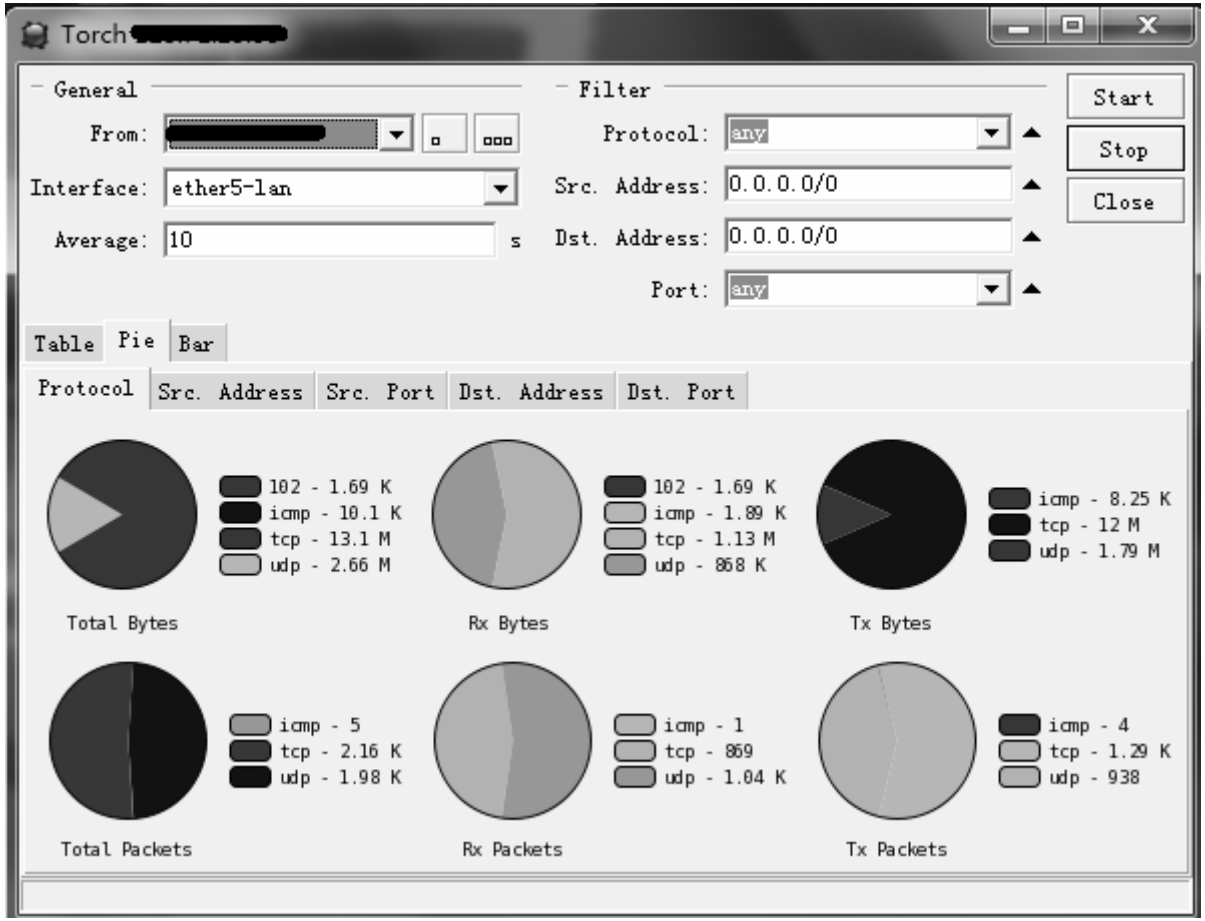
- 即RouterOS Torch工具，通过The Dude能做有效的分析



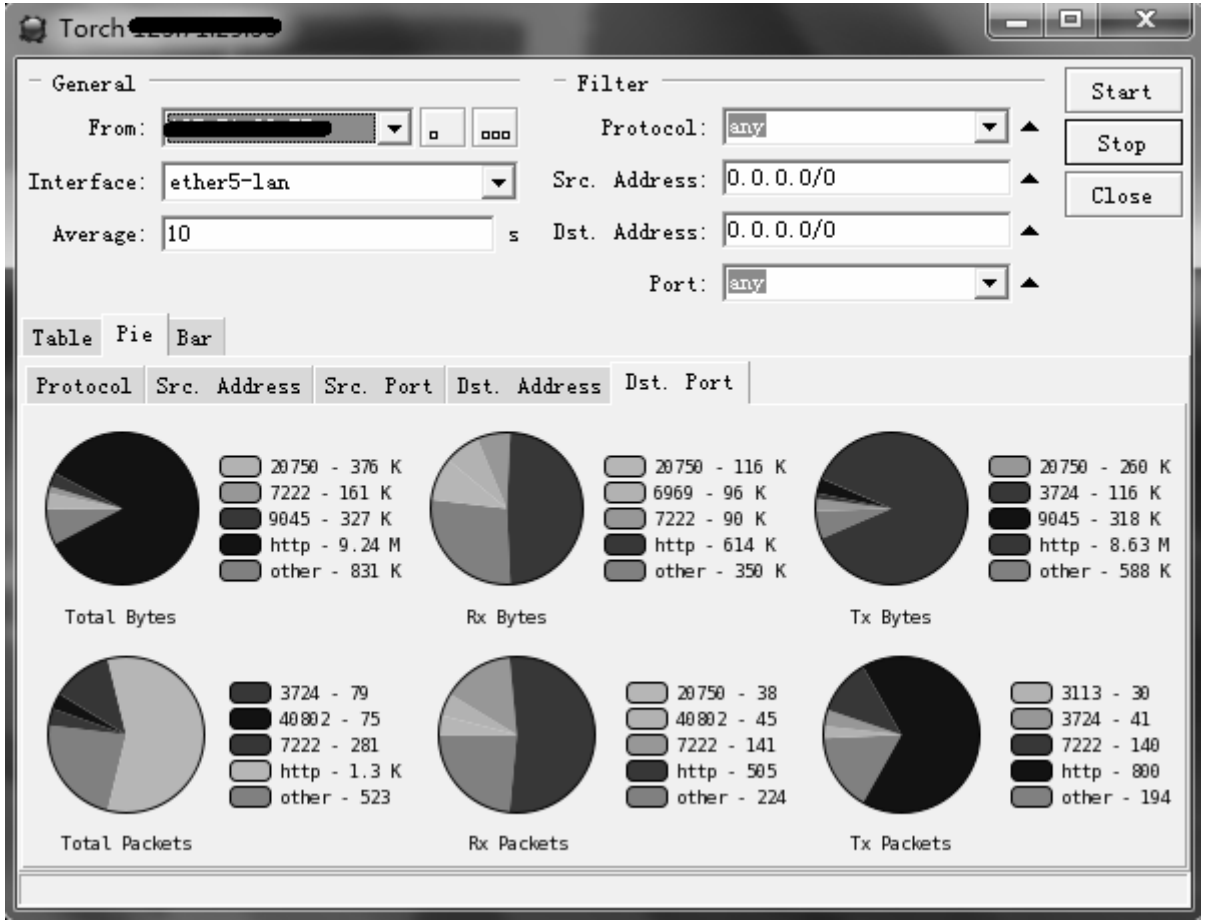
The Dude 实时监控



协议的流量分析

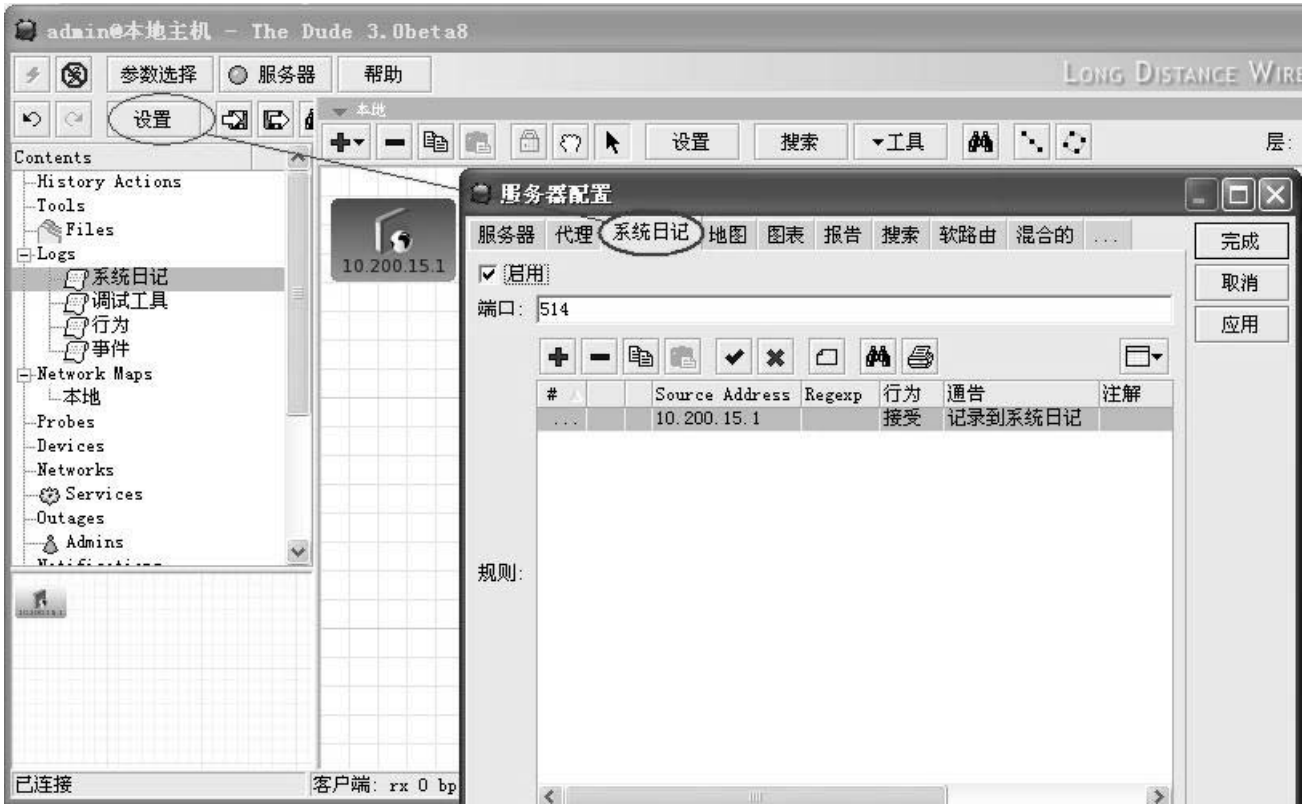

Torch
General
From:
Interface:
ether5-lan
Average:
10
s
Filter
Protocol:
any
Src. Address:
0.0.0.0/0
Dst. Address:
0.0.0.0/0
Port:
any
Start
Stop
Close
Table
Pie
Bar
Protocol
Src. Address
Src. Port
Dst. Address
Dst. Port
102 - 1.69 K
icmp - 10.1 K
tcp - 13.1 M
udp - 2.66 M
Total Bytes
102 - 1.69 K
icmp - 1.89 K
tcp - 1.13 M
udp - 868 K
Rx Bytes
icmp - 8.25 K
tcp - 12 M
udp - 1.79 M
Tx Bytes
icmp - 5
tcp - 2.16 K
udp - 1.98 K
Total Packets
icmp - 1
tcp - 869
udp - 1.04 K
Rx Packets
icmp - 4
tcp - 1.29 K
udp - 938
Tx Packets

目标端口的流量分析


Torch
General
From:
Interface: ether5-lan
Average: 10 s
Filter
Protocol: any
Src. Address: 0.0.0.0/0
Dst. Address: 0.0.0.0/0
Port: any
Start
Stop
Close
Table
Pie
Bar
Protocol
Src. Address
Src. Port
Dst. Address
Dst. Port
20750 - 376 K
7222 - 161 K
9045 - 327 K
http - 9.24 M
other - 831 K
Total Bytes
20750 - 116 K
6969 - 96 K
7222 - 90 K
http - 614 K
other - 350 K
Rx Bytes
20750 - 260 K
3724 - 116 K
9045 - 318 K
http - 8.63 M
other - 588 K
Tx Bytes
3724 - 79
40802 - 75
7222 - 281
http - 1.3 K
other - 523
Total Packets
20750 - 38
40802 - 45
7222 - 141
http - 505
other - 224
Rx Packets
3113 - 30
3724 - 41
7222 - 140
http - 800
other - 194
Tx Packets

The Dude日志记录

- 我们进入The Dude选择“设置”，并打开里面的系统日记

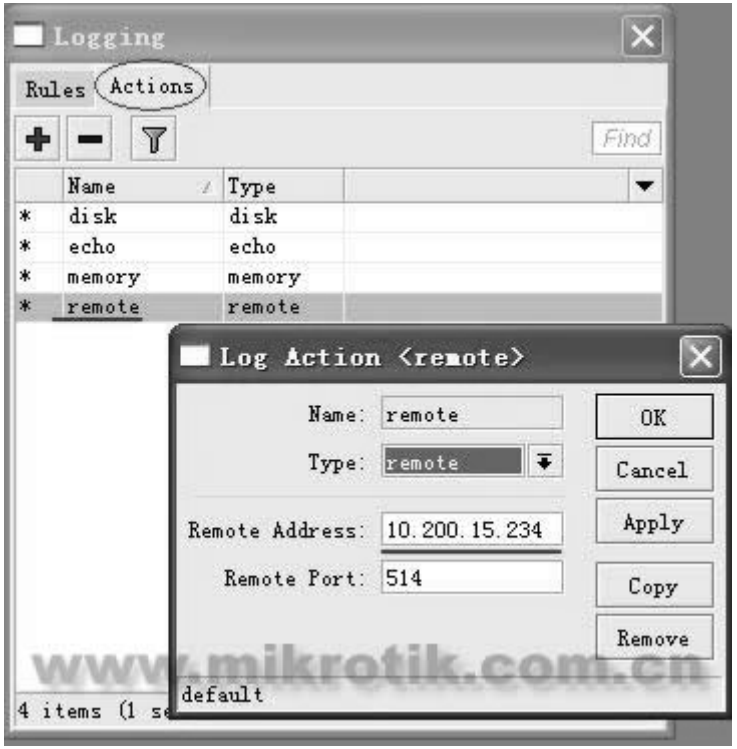


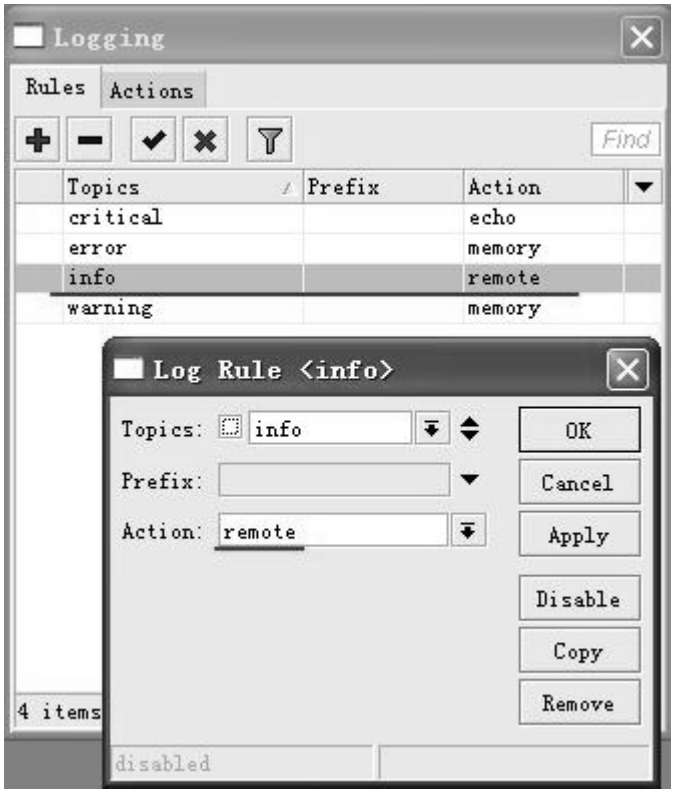
The Dude日志记录

- 进入设置选项，选择系统日记，设置10.200.15.1的RouterOS路由器的IP地址

RouterOS配置

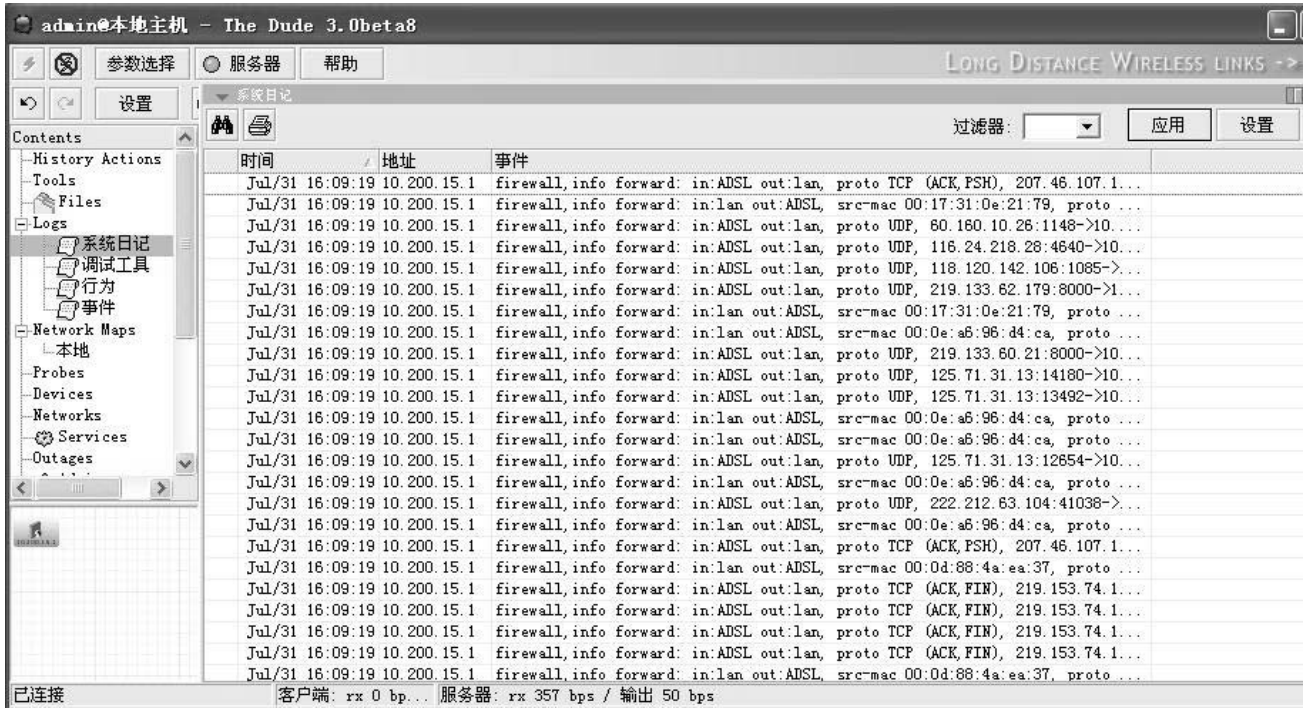
- 进入system logging设置在action标签里设置remote属性，执行我们的The Dude服务器IP地址，默认端口是514





The Dude记录

- 在The Dude的log中显示正在接收，并不断刷新的日志信息



日志保存设置

- 选择保存日志的记录时间和相关参数，在log里点加号可以设置相应的参数
- 文件到保存：保存多少个文件后，将就文件删除
- 缓冲列表：多少日志在写入硬盘时在内存缓冲