Netkit4TIC lab

Name	PoPToP
Version	1.0
Author	Sandro Doro
E-mail	sandro.doro@istruzione.it
Web	http://www.tic.fdns.net/tic/html/lab.html
Description	configuring PoPToP firewall with Linux & Windows road warriors

copyright notice

VPNs

- from Wikipedia http://en.wikipedia.org/wiki/Virtual_private_network :
  - VPN is a private communications network often used to communicate confidentially over a publicly accessible network.
- There are many types of VPN. In this slide we choose PPTP (Point-to-point tunneling protocol)
  - All releases of Microsoft Windows since Windows 95 are bundled with a PPTP client.
  - PPTP connections are authenticated with Microsoft MSCHAP-v2 or EAP-TLS. MSCHAP-v2 can be compromised if users choose weak passwords.

in this lab

- ...and usually in the real world...
  - PPTP allows remote users to securely and inexpensively access their corporate network from anywhere on the Internet.
  - with PoPTop http://www.poptop.org/ Linux servers can now function seamlessly in a PPTP VPN environment

Configuring client

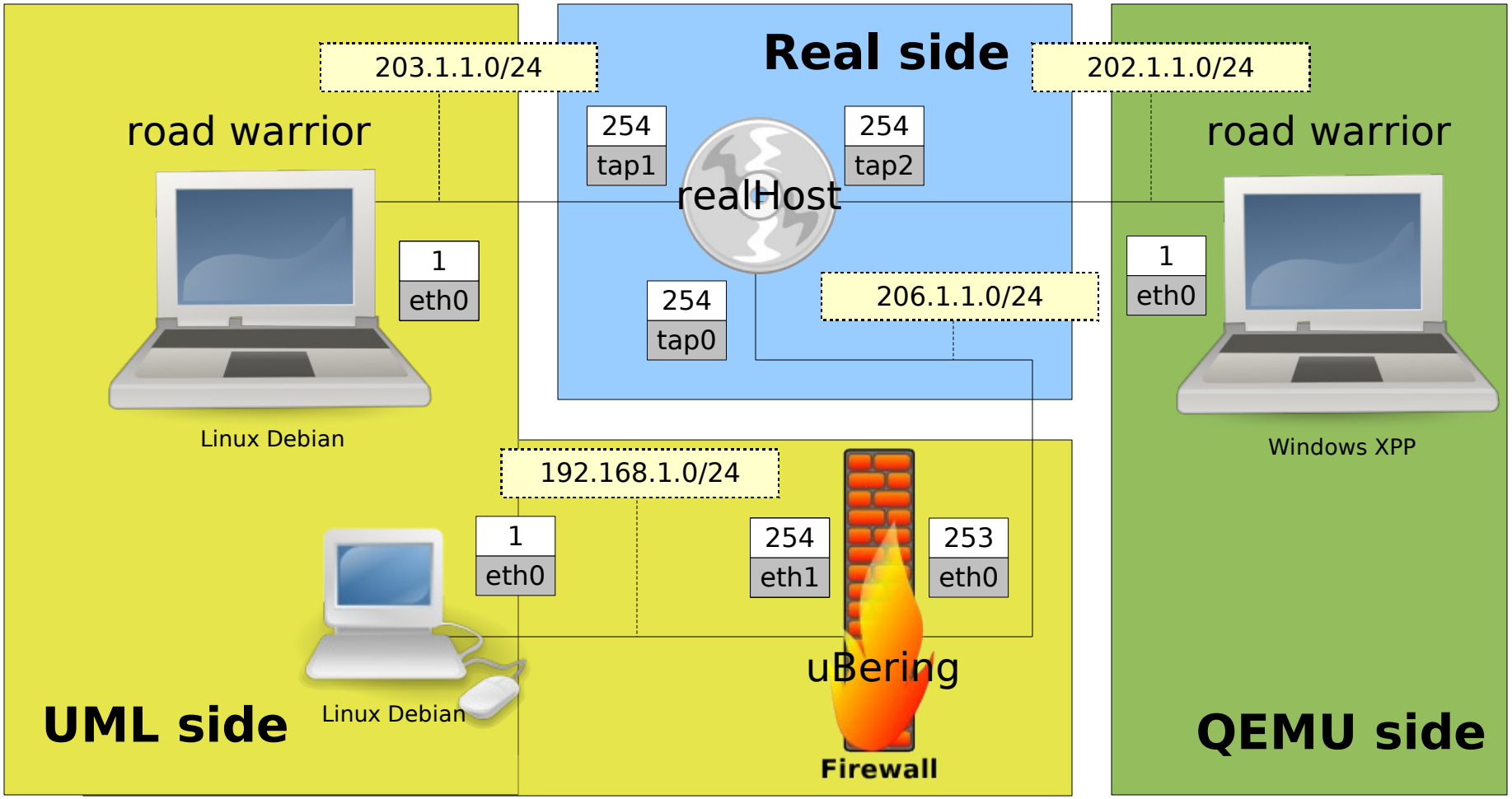
- Microsoft:
  - Google (keywords: XP PPTP Configure)
  - http://cto.secs.oakland.edu/VPN

- Linux http://pptpclient.sourceforge.net/howto-debian.phtml:
  - Using PPTP client GUI: pptpconfig
  - by hand

Configuring Server (uClibc)

- LEAF/Bering uClibc:
  - kernel just patched for mppe
  - packages: pptpd.lrp (depends on ppp.lrp)
  - shorewall:
    - /etc/shorewall/tunnels
    - /etc/shorewall/rules
  - samba as wins server

step 1 - topology



step2 – configuration (server)

fw

```
fw # cat /etc/ppp/pptpd.conf | grep -v \# | strings
option /etc/ppp/options.pptpd
mtu 1400
mru 1400
bcrelay eth1
localip 192.168.1.230-238
remoteip 192.168.1.130-138
```

step2 – configuration (server)

fw

```
fw# cat /etc/ppp/options.pptpd | grep -v \# | strings
name nk4tic
auth
refuse-pap
refuse-chap
refuse-mschap
require-mschap-v2
require-mppe-128
proxyarp
nodefaultroute
debug
dump
lock
nobsdcomp
nodeflate
```

step2 – configuration (server)

fw

```
fw# cat /etc/shorewall/interfaces
#ZONE    INTERFACE       BROADCAST       OPTIONS
net      eth0            detect          dhcp,routefilter,norfc1918,tcpflags
loc      eth1            192.168.1.255
loc      ppp+
```

fw

```
fw# cat /etc/shorewall/tunnels
# TYPE          ZONE        GATEWAY         GATEWAY ZONE
pptpserver      net         0.0.0.0/0
```

step2 - configuration client

net

```
net# cat net.startup
modprobe ppp_async
modprobe ppp_mppe
```

net

```
net# cat /etc/ppp/peers/netkit
pty "pptp 206.1.1.253 --nolaunchpppd"s$ cat netkit
name nk4tic
require-mppe-128
mtu 1400
file /etc/ppp/options.pptp
```

step2 - configuration client

net

```
net# cat /etc/ppp/chap-secrets
nk4tic * not24get *
```

net

```
net# cat /etc/ppp/options.pptp
lock auth nobsdcomp nodeflate noproxyarp
```

step3 – starting the lab

host machine

```
realHost$ tar zxf PoPTop.tgz
realHost$ cd PoPToP
realHost:~/PoPToP$ cp netkit4tic.ds ~/.devilspie
realHost:~/PoPToP$ devilspie &
```

- upon launching the lab
  - 3 virtual machines are started
  - node fw is ready to accept pptp connection from road warriors
  - road warrior with Linux OS can activare VPN with “pon netkit”
  - road warrior with Windows XP OS can activate VPN

step3 – starting (UML side)

realHost

```
realHost$ ifname1=`sudo tunctl -b -u knoppix`; \
          ifname2=`sudo tunctl -b -u knoppix`
realHost$ sudo ifconfig $ifname1 206.1.1.254; \
          sudo ifconfig $ifname2 203.1.1.254
realHost# echo "1" > /proc/sys/net/ipv4/ip_forward
realHost$ lstart
```

step3 – starting (QEMU side)

realHost

```
realHost$ ifname3=`sudo tunctl -b -u knoppix`
realHost$ su
realHost# umount /dev/shm; \
          mount -t tmpfs -o size=160m none /dev/shm; \
          modprobe kqemu; \
          cp qemu-ifup /etc; \
          exit
realHost$ export VM-Repository="path locale/remote to VM images"
realHost$ qemu-img create -b $VM-Repository/XPPSP2.img \
                          -f qcow delta-XPPSP2.img
realHost$ export QEMU_SW="-usb -usbdevice tablet -kernel-kqemu"; \
          export QEMU_NET="-net nic -net tap,$ifname3"; \
          qemu -m 140 -hda delta-XPPSP2.img $QEMU_SW $QEMU_NET
```

```
From 203.1.1.254 icmp_seq=1 Destina

--- 192.168.1.1 ping statistics ---
1 packets transmitted, 0 received,

net:~# ifconfig ppp0
ppp0      Link encap:Point-to-Point
          inet addr:192.168.1.131
          UP POINTOPOINT RUNNING NO
          RX packets:28 errors:0 dr
          TX packets:6 errors:0 dro
          collisions:0 txqueuelen:3
          RX bytes:2723 (2.6 KiB)

net:~#
```

QEMU

Cestino

Risorse del computer

Risorse di rete

C:\WINDOWS\system32\cmd.exe

```
C:\>ping 192.168.1.1

Esecuzione di Ping 192.168.1.1 con 32 byte di dati:

Risposta da 192.168.1.1: byte=32 durata=719ms TTL=63
Risposta da 192.168.1.1: byte=32 durata=2ms TTL=63
Risposta da 192.168.1.1: byte=32 durata=3ms TTL=63
Risposta da 192.168.1.1: byte=32 durata=2ms TTL=63

Statistiche Ping per 192.168.1.1:
    Pacchetti: Trasmessi = 4, Ricevuti = 4, Persi = 0 (0% persi),
Tempo approssimativo percorsi andata/ritorno in millisecondi:
    Massimo =  719ms, Medio =  181ms
```

fw

```
fw# ip addr show ppp0
16: ppp0: <POINTOPOINT,MULTICAST,NOARP,UP> mtu 1396 qdisc pfifo_fast qlen 3
    link/ppp
    inet 192.168.1.230 peer 192.168.1.130/32 scope global ppp0
fw# ip addr show ppp1
14: ppp1: <POINTOPOINT,MULTICAST,NOARP,UP> mtu 1496 qdisc pfifo_fast qlen 3
    link/ppp
    inet 192.168.1.231 peer 192.168.1.131/32 scope global ppp1
fw#
```

```
loc:~# ifconfig eth0
eth0      Link encap:Ethernet  HWad
          inet addr:192.168.1.1  Bc
          inet6 addr: fe80::fcfd:c0
          UP BROADCAST RUNNING MULT
          RX packets:35 errors:0 dr
          TX packets:17 errors:0 dr
          collisions:0 txqueuelen:1
          RX bytes:3321 (3.2 KiB)
          Interrupt:5

loc:~# []
```

fortic

start | C:\WINDOWS\syste... | IT | 15.02

net | loc | fw | QEMU

3 4

16:02 27/12/2006