

## OpenWRT - Workshop

Ondřej Caletka

6. března 2011

### 1 Cíl workshopu

Cílem workshopu je osahat si router s OpenWRT. K tomu použijeme x86 port OpenWRT, který spustíme ve virtualizéru Qemu.

### 2 Příprava počítače

\$ sudo apt-get install uml-utilities qemu bridge-utils

### 3 Stažení a práce s obrazem disku

Port OpenWRT pro x86 existuje přímo v podobě obrazu HDD, který stáhneme:

<http://shell.sh.cvut.cz/~oskar/wrt/openwrt-x86-generic-combined-ext2.img.gz>

Rozbalíme:

\$ gunzip openwrt-x86-generic-combined-ext2.img.gz

A podíváme se, co je uvnitř:

\$ file openwrt-x86-generic-combined-ext2.img  
\$ /sbin/fdisk -lu openwrt-x86-generic-combined-ext2.img

První oddíl (/boot) obsahuje zaváděč GRUB a obraz jádra Linuxu, druhý oddíl je kořenový oddíl (/) distribuce. Poškodí-li se nám v budoucnu filesystem (například nekorektním vypnutím Qemu), bude se nám hodit připojení root oddílu přes loopback:

```
$ sudo losetup -fs -o $(( 9135 * 512 )) \
--sizelimit $(( (107855 - 9135)*512 )) \
openwrt-x86-generic-combined-ext2.img
$ sudo fsck -f /vrrc0 /dev/loop0
$ sudo losetup -d /dev/loop0
```

### 4 Spuštění OpenWRT v QEMU

První pokus:

```
$ qemu -hda openwrt-x86-generic-combined-ext2.img
```

Vypadá to, že to funguje, ale, nestíže, což je u směrovače zásadní problém :)

Stiskem Enter vyvoláme shell a vypneme virtuální zadání příkazu poweroff.

Nevyplňané zavřením okna (poškodi se souborový systém).

Vytvoříme tedy síťová síťová rozhraní LAN a WAN:

```
$ sudo tunctl -u if_t wan
$ sudo tunctl -u if_t lan
```

A spusťme se dvěma síťovými kartami, propojenými na nově vytvořená TAP rozhraní v hostitelském systému:

```
$ qemu -hda openwrt-x86-generic-combined-ext2.img \
-net nic,vlan=0,model=e1000 -net nic,vlan=1,model=e1000 \
-net tap,vlan=0,ifname=lan,script=no,dnsscript=no \
-net tap,vlan=1,ifname=wan,script=no,dnsscript=no -nographic
```

Volba --nographic způsobí, že se tentokrát nespustí grafické okno, místo toho se na konzoli, kde ponustíme qemu objeví data, která přichází po virtualizované sériové lince. Protože OpenWRT má standardní konzoli i na sériové lince, bude to fungovat.

V nastartovaném OpenWRT můžeme pomocí ifconfig -a zkontrolovat, že máme dvě síťová rozhraní. První síťová karta je dokonce i aktivní a funguje jako LAN rozhraní s DHCP serverem a adresou 192.168.1.1.

Rozhraní WAN není na x86 portu OpenWRT definováno, musíme si jej dodefinovat na eth1. To můžeme udělat buď editací konfiguračního souboru /etc/config/network, nebo pomocí UCI:

```
root@openwrt: # uci batch <-EOF
set network.wan=interface
set network.wan.proto=dhcp
set network.wan.ifname=eth1
commit network
EOF
```

Máme připraveno, zatím virtuální router vypneme (pomocí poweroff).

### 5 Připojujeme do SH sítě

Nyní již máme virtuální směrovač, který má dvě síťová rozhraní, viditelná v hostitelském počítači. Chceme ho předfudit hostitelskému počítači, tak aby primo komunikoval se sítí SH rozhraním WAN a hostitelský počítač přistupoval k sítii rozhraním LAN. To ještě zhomoplíkáváme tím, že na fyzickém rozhraní počítače se nesmí objevit jiná, než jediná správná MAC adresa. (Toto je omezení sítě SH; na jiných sítích obvykle není třeba mac adresy takto složitě měnit).

Začneme tím, že vypneme sítí (V network manageru – odklonout „Povolit sítě“). Dále vytvoříme most mezi reálným rozhraním *eth0* a rozhraním *wan*. Poznamenáme si aktuální MAC adresu:

## 7 \$ První start

```
$ /sbin/iptables dev eth0  
A změněme jí na neglobální tak, že invertujeme bit 1 (s váhou 02):  
$ sudo -i  
# ip link set dev eth0 address 02:xx:xx:xx:xx:xx
```

Ted' už vytvoříme vlastní most:

```
# brctl addbr br0  
# brctl stp br0 off  
# brctl addif br0 eth0  
# brctl addif br0 wan
```

A předtím, než most nahodíme, vypneme na všech těchto rozhraních IPv6:

```
# for name in eth0 wan br0; do \  
echo 1 > /proc/sys/net/ipv6/conf/${name}/disable_ipv6 ; \  
ip link set dev $name up;  
done
```

# <Ctrl+D>

Ted' už můžeme sněrovač spustit, je však potřeba nastavit WAN síťové kartě správnou MAC adresu (takovou, která dřívější byla na skutečném rozhraní eth0):  
\$ qemu -hda openvz-x86-generic-combined-ext2.img \  
-net nic,vlan=0,model=ef1000 \  
-net nic,vlan=1,model=ef1000,macaddr=00:xx:xx:xx:xx:xx \  
-net tap,vlan=0,ifname=lan,script=no,downscript=no \  
-net tap,vlan=1,ifname=wan,script=no,downscript=no -nographic

## 6 Testujeme

Pokud všechno šlo, jak mělo, měl by nám fungovat Internet ve virtualizovaném routeru. Ověřme to:

```
root@OpenWrt:~# ifconfig  
root@OpenWrt:~# ping google.com
```

Ted' se zkuste v hostitelském počtači připojit na LAN rozhraní routera. Nejdříve budeme dráždit Network Managera, DHCP klienta spustíme ručně:

```
$ sudo dhclient lan
```

Měli bychom dostat IP adresu z rozsahu 192.168.1.0/24, nejspíše 192.168.1.128. Vyzkoušme, zda funguje NAT:

```
$ ping google.com
```

## 8 DNSSEC validátor Unbound

Pokud vše prošlo, máme funkční virtuální OpenWRT. První, co bychom měli udělat, je nastavit heslo administrátora. Sériovou konzoli obvykle nemáme k dispozici, přihlásme se tedy telnetem a nastavíme heslo:

```
$ telnet 192.168.1.1  
root@OpenWrt:~# passwd  
<Ctrl+D>  
# brctl stop br0 off  
# brctl addif br0 eth0  
# brctl addif br0 wan  
http://192.168.1.1
```

A předtím, že zavřeme telnet nefunguje, můžeme se ale přihlásit přes SSH. Také můžeme použít webové rozhraní LuCI, stačí napasat do prohlížeče adresu

## 8 DNSSEC validátor Unbound

Zkusme si nainstalovat a zaprovázit validující DNSSEC resolver unbound. Začneme instalací:

```
root@OpenWrt:~# opkg update  
root@OpenWrt:~# opkg install unbound  
Spusťte ho ještě nemůžeme, na portu udp/53 nám totiž poslouchá DNS forwarder dnsmasq. Nemůžeme ho ale vypnout, funguje i jako DHCP server, který chce používat. Užiláme to tedy tak, že přes LuCI nastavíme port pro DNS část programu dnsmasq na jiný, třeba 5353.  
Pak už můžeme unbound spustit, není napojen na UCI, ale jeho výchozí konfigurace je přímo použitelná. A rovnou ho i přidáme do runlevelu, aby se spustil automaticky.
```

```
root@OpenWrt:~# /etc/init.d/unbound start  
root@OpenWrt:~# /etc/init.d/unbound enable
```

A na hostitelském počtači vyzkoušme, zda dostáváme validní odpovědi (flag *ad*) a že se nedostaneme na podvržené domény:

```
$ dig +dnssec dnssec.cz  
$ dig +dnssec rhybar.cz
```

*Poznámka: Při workshopu se ukázalo, že unbound z nežádoucího důvodu nefunguje.*  
*Na příčinu jsem nepřišel.*

## 9 Kam dál

Další návody, co dělat v OpenWRT najdeme zde:

<http://wiki.openwrt.org/doc/start>

Na konci uvědeme počtač do původního stavu nejlépe restartem :)