Hotspot with OpenWrt

+

Private VPN access

Jan Beba, Bjoern Biesenbach

20. May 2005

Contents

1 Intro

Today many people have a broadband Internet connection and surely don't use the whole bandwidth all the time. So why don't give others the opportunity to use your connection? With this document we want to describe how to set up a hotspot using an accesspoint running with OpenWrt. A very important aspect when you decide to open your wireless network for everyone often is, that you still want to use it for your own purpose. This might be accessing a local file- or printserver or anything else not everybody in front of your house should be able to see and to use. Also your own connection should be encrypted. WEP-encryption is not only quite insecure but would also conflict with the idea of an open hotspot. So we decided to create a VPN using OpenVPN.

2 Network

The structure of our network is quite easy. We will use three separated networks; the first will be our own private network (DMZ[1]), the second the public wireless lan and the last our VPN.

LAN: 192.168.1.0/24
WLAN: 192.168.2.0/24
VPN: 192.168.3.0/24

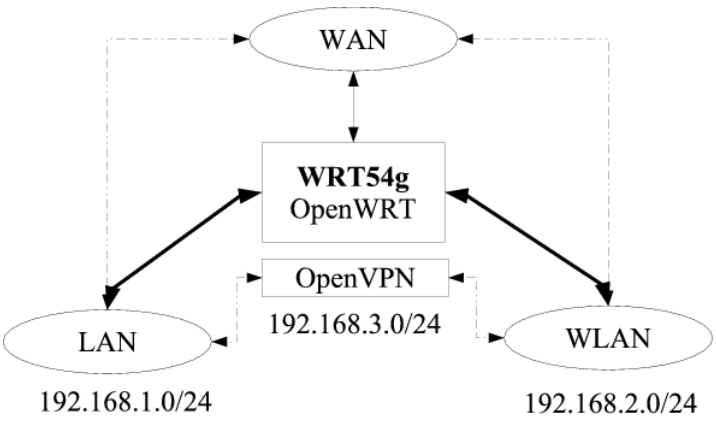

Figure 1: Network

[1] Demilitarized Zone

3 What you need

To use this howto you need the openwrt-experimental firmware with the follwoing extensions:

- openssl
- lzo
- kmod-tun
- openvpn

You also can build your own OpenWrt experimental firmware, then you need to download the sources of OpenWrt instead of the binary file. In this case you directly build the extensions mentioned above into your firmware. An instruction how to do this is in the documentation of the source file.

4 OpenWrt

Our configuration has been tested with the Linksys WRT54g versions 2.0 and 2.2. If you use other hardware please mind that the interface names may be changed. Assuming your OpenWrt installation is untouched your box is reachable via telnet on 192.168.1.1. The first thing to do is to set a password. Log into your box, type "passwd" and set your new root password. After doing so disconnect and reconnect via ssh.

Now you should get a screen like this:

```
BusyBox v1.00 (2005.04.23-22:18+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.


  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M

root@OpenWrt:~#
```

4.1 Network devices

The default config is a little tricky. The LAN-device (vlan0) and the WLAN-device (eth1) are bridged together to "br0". But as we want to have separated nets for those devices, we have to split them. Also the Internet (WAN) device has to be configured.

```
nvram set lan_ifname=vlan0
nvram set lan_proto=static
nvram set lan_ipaddr=192.168.1.1
nvram set lan_netmask=255.255.255.0

nvram set wifi_ifname=eth1
nvram set wifi_proto=static
nvram set wifi_ipaddr=192.168.2.1
nvram set wifi_netmask=255.255.255.0

nvram set wan_ifname=ppp0
nvram set wan_proto=pppoe
nvram set wan_mtu=1492

nvram set pppoe_ifname=vlan1
nvram set pppoe_username=user@provider.name
nvram set pppoe_passwd=yourpassword
nvram commit

nvram set wl0_ssid=Hotspot

reboot
```

The box will restart and *hopefully* come up again.

4.2 ipkg setup

The special thing about OpenWrt is that it comes with its own package management system, called "ipkg". We think it's comparable to the Debian "apt" system. To get this running and ready for later software install, **/etc/ipkg.conf** has to be changed. Do this with your favorite editor or by using scp. We recommend using vim ;-)

/etc/ipkg.conf

```
src experimental http://openwrt.org/downloads/experimental/bin/packages
src openwrt http://openwrt.org/ipkg
dest root /
dest ram /tmp
```

Maybe you know it from Debian, the package list has to be updated now.

```
ipkg update
```

4.3 DHCP-Server

The dnsmasq package in OpenWrt is responsible for the dhcpd functions. As we have a local LAN and a public WLAN we want to serve both with dynamically IP-address allocation. IP-addresses in the range between 192.168.1.200-192.168.1.250 and 192.168.2.200-192.168.2.250 are being offered.

/etc/dnsmasq.conf

```
domain-needed
bogus-priv
filterwin2k
local=/lan/
domain=lan

except-interface=vlan1

dhcp-range=vlan0,192.168.1.200,192.168.1.250,255.255.255.0,3h
dhcp-range=eth1,192.168.2.200,192.168.2.250,255.255.255.0,3h

dhcp-leasefile=/tmp/dhcp.leases

dhcp-option=vlan0,3,192.168.1.1
dhcp-option=vlan0,6,192.168.1.1
dhcp-option=eth1,3,192.168.2.1
dhcp-option=eth1,6,192.168.2.1
```

4.4 OpenVPN

First we should install the required software.

ipkg install openvpn

Let's create the directory and a private key for our VPN.

mkdir /etc/openvpn openvpn –genkey –secret /etc/openvpn/wlan_home.key

Load the tunneling module and add it to the autoloader.

insmod tun

echo "tun" » /etc/modules

/etc/openvpn/wlan_home.conf

```
dev tun0
ifconfig 192.168.3.1 192.168.3.2
secret /etc/openvpn/wlan_home.key
port 1194
ping 15
ping-restart 45
ping-timer-rem
persist-key
persist-tun
verb 3
```

/etc/init.d/S60openvpn

```
#!/bin/sh
openvpn --daemon --config /etc/openvpn/wlan_home.conf
```

Don't forget to assign executable rights to this file.

chmod a+x /etc/init.d/S60openvpn

4.5 Iptables setup

/etc/init.d/S45firewall

```
[...]
iptables -A FORWARD -i eth1 -o ppp0 -j ACCEPT
iptables -A FORWARD -i tun0 -j ACCEPT
iptables -A FORWARD -i vlan0 -o tun0 -j ACCEPT
```

This has to be appended! The whole file is much longer.
Finally you can do a last reboot.

5 Clientside

Now if you want to access the Internet from either your local network or via wifi you just have to select dhcp for your network device. To access your local network from out the wifi, the OpenVPN client has to be installed.

5.1 OpenVPN

Install the fitting OpenVPN client for your operating system. Copy the /etc/openvpn/wlan_home.key file from the Wrt to your client. We prefer using scp.

scp 192.168.1.1:/etc/openvpn/wlan_home.key /etc/openvpn/

If you're using M$ Windows copy the file to "C:\Program Files\OpenVPN\config". Now create the config file.

/etc/openvpn/wlan_home.conf
C:\Program Files\OpenVPN\config\wlan_home.conf

```
dev tun
remote 192.168.2.1
ifconfig 192.168.3.2 192.168.3.1
secret wlan_home.key
port 1194
route-gateway 192.168.3.1
route 0.0.0.0 0.0.0.0
```

```
redirect-gateway

ping 15
ping-restart 45
ping-timer-rem
persist-tun
persist-key

verb 3
```

Using Linux you have to load the tunnel module.

modprobe tun

Now you can start the tunnel using

openvpn –daemon –config /etc/openvpn/wlan_home.conf

For Windows just right-click onto your config and choose the second point to execute the config.