Setting an OpenWrt Based Router as OpenVPN Client

braian87b edited this page on 2 Jan · 1 revision

These instructions are for getting an OpenWrt Based Router working as OpenVPN Client (should work for LEDE, Gargoyle and another distributions). Computers connected to Lan Ports of the OpenWrt Router will navigate through the Internet connection of the OpenVPN Server (in this case the Streisand one previously set up) you need a working Router with OpenWrt based firmware flashed on it (LEDE or eko.one.pl could also work) steps works well on Chaos Calmer 15.05 or 15.05.1.

1) You need to `telnet 192.168.1.1` (OpenWrt Router) and set up a password using `passwd` You can skip this if you already have a password and can connect using ssh.

```
root@OpenWrt:~# passwd
Changing password for root
New password:
Retype password:
Password for root changed by root
```

Now you should be able to ssh the OpenWrt Router `ssh root@192.168.1.1` using the previously typed password.

2) Important: Ensure that you have at least 1MB of free space en `rootfs` on your OpenWrt Device, depending on the OpenWrt version flashed you may need more or less space to set up everything, if you dont have space but you have a USB port on the Router you could use ExtRoot, or try to build a custom image, or even try to write a script to download openvpn to ram on every start.

```
root@OpenWrt:~# df -h
Filesystem                Size      Used Available Use% Mounted on
rootfs                  239.9M     15.8M    207.3M   7% /
```

3) Install OpenVPN (you need internet connection on the OpenWrt Router)

```
opkg update
opkg install openvpn-openssl # brings openvpn kmod-tun liblzo zlib libopenssl (~1M)
```

If you want to autostart OpenVPN at router startup (in some OpenWrt releases comes enabled by default):

```
/etc/init.d/openvpn enable
```

4) Run UCI commands to configure as VPN Client:

```
# a new OpenVPN instance:
uci set openvpn.streisand=openvpn
uci set openvpn.streisand.enabled='1'
uci set openvpn.streisand.config='/etc/openvpn/streisand.conf'

# a new network interface for tun:
uci set network.streisandvpn=interface
uci set network.streisandvpn.proto='none' #dhcp #none
uci set network.streisandvpn.ifname='tun0'
```

Pages 4

Home

Installing Streisand on Cloud Provider

Installing Streisand on Local Server

Setting an OpenWrt Based Router as OpenVPN Client

Clone this wiki locally

https://github.com/jlund/st

Clone in Desktop

```
# a new firewall zone (for VPN):
uci add firewall zone
uci set firewall.@zone[-1].name='vpn'
uci set firewall.@zone[-1].input='REJECT'
uci set firewall.@zone[-1].output='ACCEPT'
uci set firewall.@zone[-1].forward='REJECT'
uci set firewall.@zone[-1].masq='1'
uci set firewall.@zone[-1].mtu_fix='1'
uci add_list firewall.@zone[-1].network='streisandvpn'

# enable forwarding from LAN to VPN:
uci add firewall forwarding
uci set firewall.@forwarding[-1].src='lan'
uci set firewall.@forwarding[-1].dest='vpn'

# Finally, you should commit UCI changes:
uci commit
```

5) DNS: There is a tricky part with this, you have to choose one of these:

- Use your Wan port default DNS (the one that uses the OpenWrt to resolve domains currently), you could leave as is then, but be aware that your DNS queries will be done through VPN and some ISP DNS Servers are configured to blocks connections attemps from outside their network.
- Set up fixed DNS on Lan Interface (only for Lan and Wifi Clients)
- Set up fixed DNS on Wan Interface (will replace the default DNS provided to the Router on Wan Port)
- Set up two script that use the DNS provided through the VPN Tunnel on the Streisand host (recommended).

I recommend the last option, you will use same DNS Server as the Streisand host, you probably should also check if on the Streisand host are configured Fixed DNS like ones from OpenDNS or Google, you could change this to use Defaults DNS on the Streisand host.

Fixed DNS on Lan interface, Using OpenDNS:

```
uci add_list dhcp.lan.dhcp_option='6,208.67.222.222,208.67.220.220'
```

Fixed DNS on Lan interface, Using Google DNS:

```
uci add_list dhcp.lan.dhcp_option='6,8.8.8.8,8.8.4.4'
```

Fixed DNS on Wan interface, using OpenDNS:

```
uci set network.wan.peerdns='0' # this disable the DNS provided by DHCP
uci del network.wan.dns # Deletes the previous list of DNS if exist.
# now add the DNS, These are from OpenDNS:
uci add_list network.wan.dns='208.67.222.222'
uci add_list network.wan.dns='208.67.220.220'
```

Fixed DNS on Wan interface, using Google DNS:

```
uci set network.wan.peerdns='0' # this disable the DNS provided by DHCP
uci del network.wan.dns # Deletes the previous list of DNS if exist.
# now add the DNS, These are from OpenDNS:
uci add_list network.wan.dns='8.8.8.8'
uci add_list network.wan.dns='8.8.4.4'
```

Finally, you should commit UCI changes:

```
uci commit
```

6) You will need to download the OpenVPN Client file from the Streisand host `[ip]-direct.ovpn` or `[ip]-sslh.ovpn` . (first one will use port 636 (ldaps), and later 443 (standard

https port), I think that exist two because some people may have restrictions in their country on some port or disallow use of ssl on another)

7) Open the `.ovpn` file on a PLAINTEXT text editor, as we need to perform some editings:

- Add this line at top `cat<<'EOF' > /etc/openvpn/streisand.conf`
- Add this line at bottom `EOF`

(these lines will enable us later to copy entire text content of the file and paste it on the terminal/putty window)

- You can comment or remove a line at the beggining of file that is something like: `router [ip] 255.255.255.255 net_gateway` , simply add `#` at the start of that line. This setting is already pushed from the OpenVPN Server side. If you don't do this, you will get an error on the OpenVPN logs, but should work fine too.

- To Enable OpenVPN log and status file:

  ```
  log-append /var/log/openvpn.log # To append to log file
  ```

  ```
  status /var/log/openvpn-status.log # To mantain a status file
  ```

- If you want to use the OpenVPN Server side DNS's from Streisand host:

  ```
  script-security 2 system # needed to be able to use 'up' and 'down' scripts
  ```

  ```
  up "/etc/openvpn/updns" # FIX DNS, we will create it later
  ```

  ```
  down "/etc/openvpn/downdns" # FIX DNS, we will create it later
  ```

8) Now copy the entire content of the `.ovpn` file and paste on Terminal, you should have now a new file (check for it):

```
ls -l /etc/openvpn/streisand.conf
```

9) If you choose to use the DNS provided by OpenVPN you need to create these two files, (just copy and paste and the code and files will be created):

```
# FIX to use DNS provided by OpenVPN server:
cat<<'EOF' > /etc/openvpn/updns
#!/bin/sh
mv /tmp/resolv.conf.auto /tmp/resolv.conf.auto.hold
echo $foreign_option_1 | sed -e 's/dhcp-option DOMAIN/domain/g' -e 's/dhcp-option DNS/namese
echo $foreign_option_2 | sed -e 's/dhcp-option DOMAIN/domain/g' -e 's/dhcp-option DNS/namese
echo $foreign_option_3 | sed -e 's/dhcp-option DOMAIN/domain/g' -e 's/dhcp-option DNS/namese
EOF
cat<<'EOF' > /etc/openvpn/downdns
#!/bin/sh
mv /tmp/resolv.conf.auto.hold /tmp/resolv.conf.auto
EOF
# add execution permission to both files:
chmod 755 /etc/openvpn/updns
chmod 755 /etc/openvpn/downdns
```

You should have now two new files (check for it):

```
ls -l /etc/openvpn/*dns
```

10) All Ready!

Since we modified firewall we need to run

```
/etc/init.d/firewall reload
```

Since we added a new interface we need to restart network daemon (you will lost connectivity for a moment)

```
/etc/init.d/network restart
```

Start OpenVPN and see what and see what happens:

```
/etc/init.d/openvpn stop      # stop daemon in case that is currently running
rm /var/log/openvpn.log       # delete previous OpenVPN log
/etc/init.d/openvpn start     # start OpenVPN
sleep 1                       # wait a second.
tail -f /var/log/openvpn.log  # monitor log.
```

When you successfully see `Initialization Sequence Completed` you can press `CTRL+C` to exit. You can do `traceroute 8.8.8.8` or some other IP to see if you pass through the VPN or check online your Public IP.

Important Remarks about testing if it works properly:

- Please always test VPN using `ping`, `traceroute`, `wget` or even browsing to an IP and not browsing to a domain, since you may have a working VPN but not working DNS.
- If you reboot your router allow a 30-60sec to properly boot and bring up internet (important if you have extroot or a slow router), and additional 30-60sec to bring up VPN.

11) Bonus! Enable WiFi:

If you started from scratch and you want to enable WiFi (if your router have dual-band replace `[-1]` with `[0]`:

```
uci set wireless.@wifi-iface[-1].ssid='OpenWrt With OpenVPN'
uci set wireless.@wifi-iface[-1].encryption='psk2+aes'
uci set wireless.@wifi-iface[-1].key='put here a password' # this should be more than 8 char
uci commit wireless
wifi reload
```

12) Related info: Just in case the OpenVPN client file change in future: The content config at the beggining of a working .ovpn as is (doesn't include any needed modifications):

```
client
remote 123.456.789.012 636
dev tun
proto tcp
cipher AES-256-CBC
auth SHA256
resolv-retry infinite
nobind
persist-key
persist-tun
ns-cert-type server
comp-lzo
key-direction 1
verb 3
route 123.456.789.012 255.255.255.255 net_gateway
```

Note: 123.456.789.012 represents the Streisand host IP

Configuration Pushed by the OpenVPN Server on Streisand Host (taken from `/var/log/openvpn.log`)

```
PUSH: Received control message: 'PUSH_REPLY,redirect-gateway def1,dhcp-option DNS 10.8.0.1,r
```

TODO: Add necessary code to have one WiFi Network with VPN and other without (in a few days)...

braian87