How to configure OpenVPN to resolve local DNS & hostnames

From steamWiki

OpenVPN (https://openvpn.net/) is a great tool to allow remote access of your network and I've been using it on and off for years. For longer than I care to admit I've been trying to solve a particularly pesky problem by which I've been unable to access local resources (other computers on my LAN) by their hostname. I could always access these resources by their IP addresses but using this system was cumbersome as I'd have to remember their IP addresses (instead of their easier to recall hostnames) and services that were setup to access resources by hostname (such as mapped network drives) wouldn't work. In this article I'll describe the parts of my network setup that are relevant and the procedure I implemented to allow an OpenVPN client to be able to access machines on the LAN by their hostnames.

This solution applies to a Linux based OpenVPN server and Linux based client.

Contents

Straight to the Solution

Here's the solution up front. Check out the rest of the article for more details on my setup.

The problem boils down to the fact that, by default, the client's *resolv.conf* file doesn't contain a line to point the client to the VPN's DNS server nor does it contain a line telling the client what your local domain name is. Getting these two lines added (and removed) from *resolv.conf* automatically is the goal.

Server Mod

1. Ensure the following two lines are in your *server.conf* (typically at */etc/openvpn/server.conf*). This tells the client that they should use *192.168.1.1* as the DNS server (typically your router's IP) and *mylocaldomain.lan* as a domain to sort of "automatically" append to hostnames that are requested.

```
push "dhcp-option DNS 192.168.1.1"
push "dhcp-option DOMAIN mylocaldomain.lan"
```

Client Mod

1. Install the *resolvconf* package to give your OpenVPN client the ability to rebuild the *resolv.conf* file when you start and stop your VPN connection; backup/remove your existing *resolv.conf* file; and create a symlink (https://en.wikipedia.org/wiki/Symbolic_link) to *resolvconf*'s *resolv.conf* file. This can all be done by running the following

   ```
   sudo apt install resolvconf
   sudo mv /etc/resolv.conf /etc/resolv.conf.orig
   sudo ln -s /run/resolvconf/resolv.conf /etc/resolv.conf
   ```

2. Add the following 2 lines to your *client.ovpn* file to run *update-resolv-conf* every time you connect to or disconnect from your VPN server

   ```
   up /etc/openvpn/update-resolv-conf
   down /etc/openvpn/update-resolv-conf
   ```

3. Run the *openvpn* command with *--script-security 2* flag to allow *update-resolv-conf* to be run as in the following example

   ```
   sudo openvpn --script-security 2 --config /path/to/client.ovpn
   ```

More Detailed Explanation

Router Setup

My TP-Link TL-WDR3600 (https://wikidevi.com/wiki/TP-LINK_TL-WDR3600) router is setup as a DNS (https://www.startpage.com/do/search) Server and DHCP (https://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol) Server.

- My router is loaded with DD-WRT (https://dd-wrt.com/) Firmware v24-sp2 (03/25/13) std
- **Setup** -> **Network Address Server Settings (DHCP)** -> **Use DNSMasq for DNS** is *checked*
- **Services** -> **Services** -> **LAN Domain** is set to *mylocaldomain.lan*
- Static IP addresses for LAN resources (computers) are assigned at **Services** -> **Services** -> **DHCP Server** -> **Static Leases**

OpenVPN Server Setup

- My OpenVPN is running on an LXC Container hosted on a Proxmox (https://www.proxmox.com/en/) server.
- I setup my OpenVPN server using a script from https://raw.githubusercontent.com/davejm/OpenVPN-install/tcp/openvpn-install.sh

- My server configuration file (*/etc/openvpn/server.conf*) looks like this

```
port 1194
proto tcp
dev tun
ca ca.crt
cert server.crt
key server.key
dh dh.pem
user nobody
group nogroup
topology subnet
server 192.168.2.0 255.255.255.0
ifconfig-pool-persist ipp.txt
cipher AES-256-CBC
auth SHA512
tls-version-min 1.2
tls-cipher TLS-DHE-RSA-WITH-AES-256-GCM-SHA384
push "dhcp-option DNS 192.168.1.1"
push "dhcp-option DOMAIN mylocaldomain.lan"
push "route 192.168.0.0 255.255.0.0"
keepalive 10 120
persist-key
persist-tun
crl-verify crl.pem
tls-server
tls-auth tls-auth.key 0
log /var/log/openvpn/openvpn.log
status /var/log/openvpn/status.log 20
```

  - Make sure that your OpenVPN IP pool (the *server 192.168.2.0 255.255.255.0* line does not conflict with the addresses assigned by your router / DHCP server. In this example all local resources are at 192.168.1.XXX and all OpenVPN clients are at 192.168.2.XXX.
  - The line *push dhcp-option DNS 192.168.1.1* tells the server to send the address of the local networks DNS server (in this case your router) to the client
  - The line *push dhcp-option DOMAIN mylocaldomain.lan* tells the server to send your local domain to the client as a place for it to search for hostnames that are used by not fully qualified (https://en.wikipedia.org/wiki/Fully_qualified_domain_name).

OpenVPN Client Setup

As mentioned above, the crux of the problem is that the client's *resolve.conf* files doesn't contain everything it needs. We've already modified the *server.conf* file to tell the server to send the necessary options to the client, but we have to make some changes on the client to ensure these options actually get put where they need to go (in the *resolve.conf* file)

- *resolve.conf* is automatically generated, and updated, by the OS. Therefore we can't really modify it directly or our changes will simply be lost. a package called *resolveconf* comes to the rescue. Once installed we replace the *resolv.conf* file (typically located at */etc/resolv.conf* with a symlink to *resolvconf's* version of the file. This file gets modified by *resolvconf* which we can take advantage of with OpenVPN. The procedure is to install *resolveconf*; then move/backup the original *resolv.conf* file; and create a symlink to *resolvconf* instance of the *resolv.conf* file.

```
sudo apt install resolvconf
sudo mv /etc/resolv.conf /etc/resolv.conf.orig
sudo ln -s /run/resolvconf/resolv.conf /etc/resolv.conf
```

- Now that we have setup *resolvconf* we can use it in the OpenVPN client configuration to take the DNS and DOMAIN information setup in the server's *server.conf* and insert them into the client's *resolv.conf*. Add the following 2 lines to your *client.ovpn* file which will run *update-resolv-conf* each time you start, and stop, OpenVPN.

```
up /etc/openvpn/update-resolv-conf
down /etc/openvpn/update-resolv-conf
```

- Lastly, in order to allow the OpenVPN to actually call *update-resolv-conf* we have to tell *openvpn* that we want to ease off its default security posture a bit. The follow command will allow you to connect to your OpenVPN server in a way that allows *update-resolv-conf* to run at start & stop.

```
sudo openvpn --script-security 2 --config /path/to/client.ovpn
```

References

1. http://www.softwarepassion.com/solving-dns-problems-with-openvpn-on-ubuntu-box/
2. https://serverfault.com/questions/318563/how-to-push-my-own-dns-server-to-openvpn

Retrieved from "https://steamforge.net/wiki/index.php?title=How_to_configure_OpenVPN_to_resolve_local_DNS_%26_hostnames&oldid=1563"

Category: Linux

This page was last modified on 21 June 2021, at 13:34.