Part 6 – Final troubleshooting

Enable Wireguard on startup

If we are going to move SSH into the Wireguard tunnel only, we need to make sure that the interface is active even (or especially) after the server restarts. For that, we need to enable Wireguard with systemd (or other init service). Fortunately it's pretty easy.

```
$ sudo systemctl enable wg-quick@wg0.service
$ sudo systemctl daemon-reload
$ sudo wg-quick down wg0
$ sudo systemctl start wg-quick@wg0
```

It's time to test all of our efforts – reboot the server and hope all configuration stays correct afterwards. I like to do this, even though it's not neccessary, to save myself from unnecessary headaches when the server unexpectedly restarts and it turns out that the configuration wouldn't survive a schedules restart anyway.

Troubleshooting post-reboot

First thing I noticed is that Nginx was unable to start. Wireguard interface started up fine, but the webserver didn't, even after manual service restart. Turns out Apache2 was starting as a service and occupying port 80. I am too scared to completely purge Apache2 from the system, so I will just stop and disable the service.

```
$ sudo systemctl stop apache2
$ sudo systemctl disable apache2
```

Time for another reboot. This time, Nginx failed to start again, but manual restart of the serviced fixed it. This leads me to the idea, that Nginx is trying to start sooner than Wireguard and is unable to bind to the Wireguard interface, because it does not exist yet.

Edit the `nginx.service` systemd file to make sure it starts after Wireguard had already brought up the interface. You can either directly edit `/lib/systemd/system/nginx.service` file, but that is a bad practice, because the file is usually overwritten with updates. The correct way to do this should be use `sudo systemctl edit --full nginx.service`. Add `wg-quick@wg0.service` to the line end of the line with `After=` and add a new line `Requires=wg-quick@wg0.service` before the previous one.

After another reboot, Nginx started on its own successfully. List systemd services to check if there are any other issues.

```
$ systemctl list-units --type=service

  UNIT                               LOAD   ACTIVE SUB     DESCRIPTION
  console-getty.service              loaded active running Console Getty
  dbus.service                       loaded active running D-Bus System Message Bus
● dnsmasq.service                    loaded failed failed  dnsmasq - A lightweight DHCP and caching DNS server
  filtron.service                    loaded active running filtron
  ifupdown-pre.service               loaded active exited  Helper to synchronize boot up for ifupdown
  networking.service                 loaded active exited  Raise network interfaces
  nginx.service                      loaded active running nginx - high performance web server
  ssh.service                        loaded active running OpenBSD Secure Shell server
  systemd-journal-flush.service      loaded active exited  Flush Journal to Persistent Storage
  systemd-journald.service           loaded active running Journal Service
  systemd-logind.service             loaded active running User Login Management
  systemd-modules-load.service       loaded active exited  Load Kernel Modules
  systemd-networkd.service           loaded active running Network Service
  systemd-remount-fs.service         loaded active exited  Remount Root and Kernel File Systems
  systemd-resolved.service           loaded active running Network Name Resolution
  systemd-sysctl.service             loaded active exited  Apply Kernel Variables
  systemd-sysusers.service           loaded active exited  Create System Users
  systemd-tmpfiles-setup-dev.service loaded active exited  Create Static Device Nodes in /dev
  systemd-tmpfiles-setup.service     loaded active exited  Create Volatile Files and Directories
● systemd-udev-trigger.service       loaded failed failed  Coldplug All udev Devices
  systemd-udevd.service              loaded active running Rule-based Manager for Device Events and Files
  systemd-update-utmp.service        loaded active exited  Update UTMP about System Boot/Shutdown
  systemd-user-sessions.service      loaded active exited  Permit User Sessions
  user-runtime-dir@1000.service      loaded active exited  User Runtime Directory /run/user/1000
  user@1000.service                  loaded active running User Manager for UID 1000
  uwsgi.service                      loaded active running LSB: Start/stop uWSGI server instance(s)
  wg-quick@wg0.service               loaded active exited  WireGuard via wg-quick(8) for wg0
  whoogle.service                    loaded active running Whoogle

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.
28 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.
```

It seems that dnsmasq also failed to start and I assume it was due to the same issue. Edit dnsmasq with systemd again:

```
$ sudo systemctl edit --full dnsmasq.service
```

Both lines `Requires` and `After` already exist, so just add wg-quick@wg0.service on each of these line:

```
[Unit]
Description=dnsmasq - A lightweight DHCP and caching DNS server
Requires=network.target wg-quick@wg0.service
Wants=nss-lookup.target
Before=nss-lookup.target
After=network.target wg-quick@wg0.service
```

This howerver, creates a paradox. DNSmasq **requires** `wg-quick` and **starts after** `wg-quick`, but also **before** `nss-lookup.target`

Now examine `wg-quick` (`systemctl cat wg-quick@wg0`)

```
...
After=network-online.target nss-lookup.target
Wants=network-online.target nss-lookup.target
...
```

According to this configuraion – `wg-quick` starts **after** `nss-lookup.target` when `DNSmasq` has to start **before** `nss-lookup.target`, while also starting **after** `wg-quick`, which has to start **after** `nss-lookup.target`......and we got a loop. My solution to this is to simply comment out the `DNSmasq` dependency of starting before `nss-lookup.target`.

```
$ sudo systemctl edit --full dnsmasq.service
```

```
[Unit]
Description=dnsmasq - A lightweight DHCP and caching DNS server
Requires=network.target wg-quick@wg0.service
Wants=nss-lookup.target
# Before=nss-lookup.target
After=network.target wg-quick@wg0.service
```

This shouldn't break anything. Here's a sidenote about what `nss-lookup.target` even is:

A target that should be used as synchronization point for all host/network name service lookups. Note that this is independent of UNIX user/group name lookups for which nss-user-lookup.target should be used. All services for which the availability of full host/network name resolution is essential should be ordered after this target, but not pull it in. systemd automatically adds dependencies of type After= for this target unit to all SysV init script service units with an LSB header referring to the "$named" facility.

The only thing that's left is to restore iptables at boot according to the config.

Setup iptables-persistent

There are multiple ways to make iptables rules persist accross reboots, but this seems to be the prefered way.

Install `iptables-persistent`. It will ask you to save the current configuration to a file. We already have a config file present and this would only overwrite our file, so say no.

```
$ sudo apt install iptables-persistent
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  netfilter-persistent
The following NEW packages will be installed:
  iptables-persistent netfilter-persistent
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 23.4 kB of archives.
After this operation, 91.1 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Try rebooting and checking with `sudo iptables -L -nv` if the rules have been applied.

Revision #3
Created 22 September 2021 01:53:07 by Marek
Updated 22 September 2021 01:59:35 by Marek