Ubuntu Networking Configuration Lab

Purpose

The purpose of this lab is to get hands on practice utilizing the three methods of network configuration available on Ubuntu Linux systems. Knowing how to properly and efficiently configure networking on Linux systems is vital since a vast majority of the work done on Linux requires networking. Even simple tasks like browsing the internet or streamlining a video will require the network to be configured properly. Luckily, on many Linux distributions there are multiple methods that can be used to configure networking. In this lab I will cover how to configure network settings temporarily via various command line tools, using configuration files to configure networking, and utilizing the GUI network configuration tool that comes with Ubuntu.

Skills Applied

- Command Line Network Configuration
- Netplan Configuration
- GUI Network Settings
- Troubleshooting Network Issues
- Using Network Diagnostic Tools

Sections

1. Command Line Network Configuration
2. Netplan Network Configuration
3. GUI Network Configuration

Section #1 Command Line Network Configuration

Note: Most command line based network configuration is only temporary and will be gone after a system reboot. The main methods for creating persistent network changes are via editing configuration files or using the GUI.

1. Display the current IP configuration.

You can view your current IP configuration using the ip a, or the ip addr show command.

I used the ip a command this time. It tells important configuration information for the various network interfaces used by your system. Information like the MTU (the largest amount of data that can be sent over the interface), the state of the interface (is it up/down) and whether the interface is a default route for traffic to leave the device from. Information at the network layer such as the ip address, broadcast address, and whether they are static or determined by DHCP is included as well.

```
spy2@Laptop:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b5:c7:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.30/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
       valid_lft 86313sec preferred_lft 86313sec
    inet6 fe80::f09b:6c1a:b386:a292/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
spy2@Laptop:~$
```

2. Bring a network interface up and down.

To bring a network interface up or down you can use the ip link set command.

```
spy2@Laptop:~$ sudo ip link set  enp0s3 down
[sudo] password for spy2:
spy2@Laptop:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
    link/ether 08:00:27:b5:c7:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.30/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
       valid_lft 85980sec preferred_lft 85980sec
    inet6 fe80::bf8:ab74:9573:bfbd/64 scope link tentative noprefixroute
       valid_lft forever preferred_lft forever
spy2@Laptop:~$
```

Bringing the interface down means that it is turned off and won't accept network traffic or send traffic.

```
spy2@Laptop:~$ sudo ip link set  enp0s3 up
spy2@Laptop:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b5:c7:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.30/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
       valid_lft 86399sec preferred_lft 86399sec
    inet6 fe80::f09b:6c1a:b386:a292/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

3. Assign a static IP address to an interface (until next boot):

```
spy2@Laptop:~$ sudo ip addr add 192.168.1.100/24 dev enp0s3
spy2@Laptop:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b5:c7:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.30/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
       valid_lft 86326sec preferred_lft 86326sec
    inet 192.168.1.100/24 scope global secondary enp0s3
       valid_lft forever preferred_lft forever
    inet6 fe80::f09b:6c1a:b386:a292/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
spy2@Laptop:~$
```

4. Add a default gateway.

A default gateway (aka the router ip address) can be configured via the command line with the ip route add command. The key thing to consider is what the router ip address is.

```
spy2@Laptop:~$ sudo ip route add default via 192.168.1.1
spy2@Laptop:~$ ip route
default via 192.168.1.1 dev enp0s3
default via 192.168.1.1 dev enp0s3 proto dhcp metric 100
169.254.0.0/16 dev enp0s3 scope link metric 1000
192.168.1.0/24 dev enp0s3 proto kernel scope link src 192.168.1.30 metric 100
spy2@Laptop:~$
```

5. Configure DNS settings.

You can add a DNS server to use temporarily by editing the /etc/resolv.conf

```
  GNU nano 6.2                                        /etc/resolv.conf *
#
# This is a dynamic resolv.conf file for connecting local clients to
# internal DNS stub resolver of systemd-resolved. This file lists al
# configured search domains.
#
# Run "resolvectl status" to see details about the uplink DNS server
# currently in use.
#
# Third party programs should typically not access this file directl
# through the symlink at /etc/resolv.conf. To manage man:resolv.conf
# different way, replace this symlink by a static file or a differen
#
# See man:systemd-resolved.service(8) for details about the supporte
# operation for /etc/resolv.conf.

nameserver 127.0.0.53
options edns0 trust-ad
search .

nameserver 1.1.1.1
```

This is a temporary solution since the /etc/resolv.conf files contents will be cleared upon system reboot. A more stable way of configuring the DNS servers for your system is to use Netplan on the GUI.

6. Test connectivity to an external host.

```
spy2@Laptop:~$ ping x.com
PING x.com (172.66.0.227) 56(84) bytes of data.
64 bytes from 172.66.0.227 (172.66.0.227): icmp_seq=1 ttl=56 time=24.6 ms
64 bytes from 172.66.0.227 (172.66.0.227): icmp_seq=2 ttl=56 time=19.3 ms
64 bytes from 172.66.0.227 (172.66.0.227): icmp_seq=3 ttl=56 time=15.5 ms
64 bytes from 172.66.0.227 (172.66.0.227): icmp_seq=4 ttl=56 time=15.2 ms
^C
--- x.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3011ms
rtt min/avg/max/mdev = 15.198/18.630/24.587/3.795 ms
spy2@Laptop:~$ nslookup x.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   x.com
Address: 172.66.0.227
```

As always it is a good idea to test that your network configuration is working. I pinged x.com and received a response. That means that my gateway address has been configured correctly since I am able to make requests to remote devices. Also, I used the host name for X rather than its ip address so DNS works as well. I further verified that DNS works by using nslookup too.

Section #2 Netplan Network Configuration

1. Locate the Netplan configuration files.

Netplan configuration files are found under the /etc/netplan/ directory. Typically the Ubuntu system will come with one configuration file by default which is a .yaml file. You are able to create custom .yaml files for various use cases. For example you can have one file for wired connections and another for WIFI connections. ***It should be noted that Netplan will apply all configuration files linked to an interface so best practice is to use one file per interface***. Having more than one file for an interface will cause confusion for Netplan.

```
spy2@Laptop:~$ cd /etc/netplan/
spy2@Laptop:/etc/netplan$ ls -l
total 4
-rw-r--r-- 1 root root 104 Feb 20  2024 01-network-manager-all.yaml
spy2@Laptop:/etc/netplan$
```

As you can see I only have the one default Netplan file.

2. Identify the current configuration file used by the system.

Now time to check what the current Netplan configuration is:

```
spy2@Laptop:/etc/netplan$ ls -l
total 4
-rw-r--r-- 1 root root 104 Feb 20  2024 01-network-manager-all.yaml
spy2@Laptop:/etc/netplan$ sudo cat 01-network-manager-all.yaml
[sudo] password for spy2:
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
spy2@Laptop:/etc/netplan$
```

The current configuration just says that the renderer is NetworkManager. There are no other settings being applied by Netplan since it is not being used.

3. Create a new YAML configuration file for static IP assignment.

I will create a netplan_static.yaml file to use as a custom Netplan file.

```
spy2@Laptop:/etc/netplan$ sudo touch netplan_static.yaml
spy2@Laptop:/etc/netplan$
```

Now I will configure the file to assign a static ip address to the system.

```
  GNU nano 6.2                    netplan_static.yaml *
network:
  version: 2
  renderer: NetworkManager
  ethernets:
      enp0s3:
          dhcp4: no
          addresses:
              - 192.168.1.30/24
```

I assigned 192.168.1.30 to the system. Note that I used the CIDR block for the subnet instead of the subnet mask address. Also I put no for dhcp4 meaning IPv4 addresses will not be given to the system via DHCP.

4. Add DNS servers and the gateway to the configuration file.

```
  GNU nano 6.2                    netplan_static.yaml *
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enp0s3:
      dhcp4: no
      addresses:
        - 192.168.1.30/24
      gateway4: 192.168.1.1
      nameservers:
        addresses: [1.1.1.1, 1.1.1.2]
```

Underneath the addresses line I added the gateway4 and nameservers lines. Notice how all the lines under enp0s3 are aligned with each other. This is vital for yaml files since yaml uses indentation to determine if

lines are related or not. Typically sub-settings are indented 2 spaces underneath the main setting. For instance, notice how under the nameservers line how the addresses line is indented using two spaces. This signifies that the addresses line is a sub-setting of the nameservers line.

5. Apply the configuration.

Use sudo netplan try first:

Netplan try will attempt to run the Netplan configuration changes you created. If there is anything wrong with the syntax or settings it will inform you so:

```
** (process:47235): WARNING **: 13:53:53.667: Permissions for /etc/netplan/netpl
an_static.yaml are too open. Netplan configuration should NOT be accessible by o
thers.

** (process:47235): WARNING **: 13:53:53.667: `gateway4` has been deprecated, us
e default routes instead.
See the 'Default routes' section of the documentation for more details.
Do you want to keep these settings?


Press ENTER before the timeout to accept the new configuration


Changes will revert in 100 seconds
```

Apparently I do have some errors to deal with. I need to change the permissions on the file and the routing settings.

```
spy2@Laptop:/etc/netplan$ ls -l
total 8
-rw-r--r-- 1 root root 104 Feb 20  2024 01-network-manager-all.yaml
-rw-r----- 1 root root 253 Mar 12 14:05 netplan_static.yaml
spy2@Laptop:/etc/netplan$ sudo chmod 600 netplan_static.yaml
spy2@Laptop:/etc/netplan$ ls -l
total 8
-rw-r--r-- 1 root root 104 Feb 20  2024 01-network-manager-all.yaml
-rw------- 1 root root 253 Mar 12 14:05 netplan_static.yaml
spy2@Laptop:/etc/netplan$
```

I changed the permissions for netplan_static.yaml.

```
  GNU nano 6.2                                    netplan_static.yaml *
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enp0s3:
      dhcp4: no
      addresses:
        - 192.168.1.30/24
      nameservers:
        addresses: [1.1.1.1, 1.1.1.2]
      routes:
        - to: default
          via: 192.168.1.1
```

Then I fixed the routing settings. Time to test the file again.

```
spy2@Laptop:/etc/netplan$ sudo netplan try /etc/netplan/netplan_static.yaml

** (process:111660): WARNING **: 14:07:46.231: Permissions for /etc/netplan/01-network-manager-all.yaml
en. Netplan configuration should NOT be accessible by others.

** (generate:111664): WARNING **: 14:07:46.240: Permissions for /etc/netplan/01-network-manager-all.yaml
pen. Netplan configuration should NOT be accessible by others.

** (process:111660): WARNING **: 14:07:47.473: Permissions for /etc/netplan/01-network-manager-all.yaml
en. Netplan configuration should NOT be accessible by others.

** (process:111660): WARNING **: 14:07:47.983: Permissions for /etc/netplan/01-network-manager-all.yaml
en. Netplan configuration should NOT be accessible by others.

** (process:111660): WARNING **: 14:07:47.984: Permissions for /etc/netplan/01-network-manager-all.yaml
en. Netplan configuration should NOT be accessible by others.
Do you want to keep these settings?
```

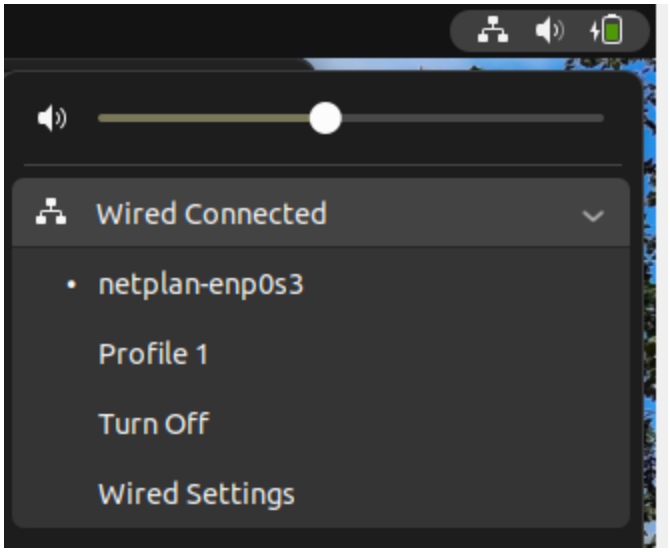
The new file is okay! I will now apply the new configuration. The errors appearing on the screen are for the other Netplan configuration file that is not in use (it is the default file).

```
spy2@Laptop:/etc/netplan$ sudo netplan apply
spy2@Laptop:/etc/netplan$
```

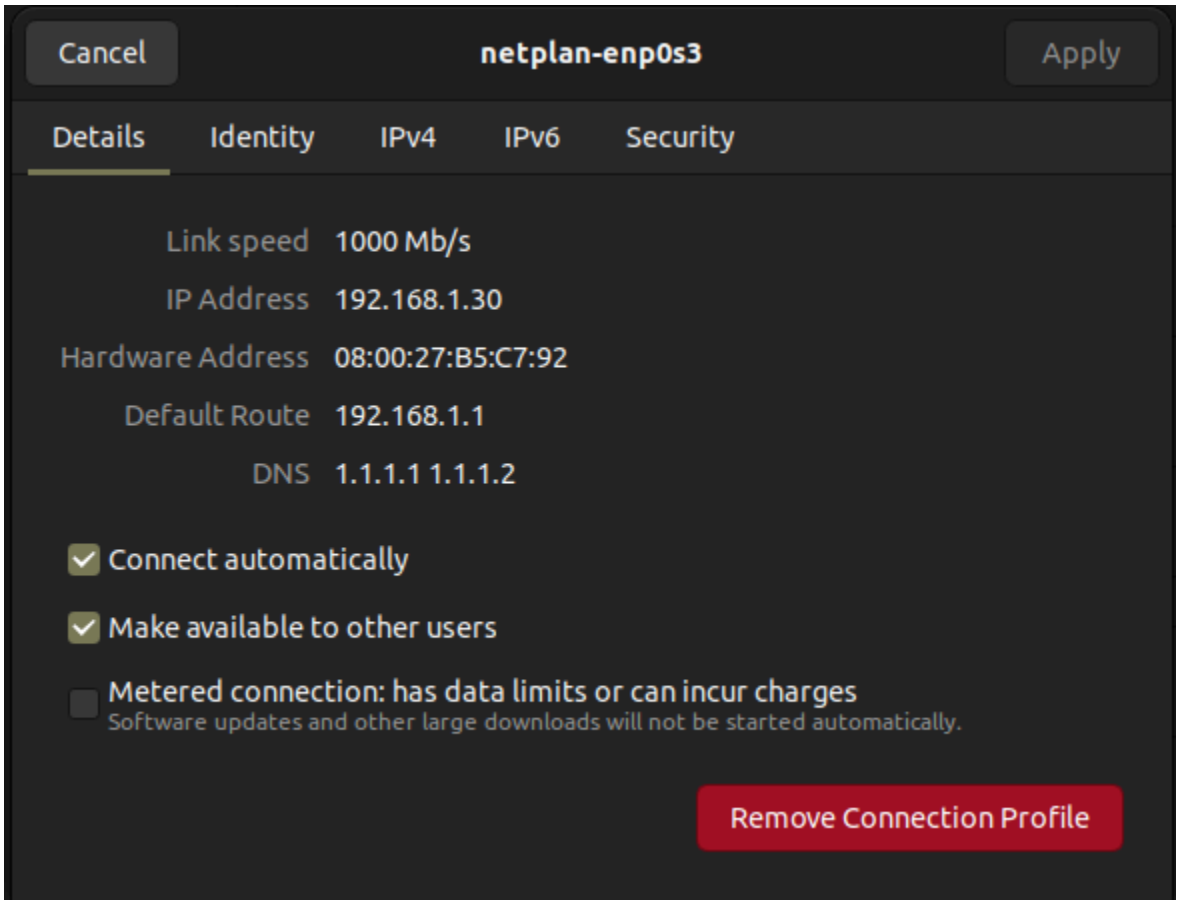
I was able to apply it with no errors!

6. Verify the configuration changes.

You can tell if the netplan configuration is working by looking at the connection profiles:



I chose netplan-enp0s3 to enable its use. You need to enable the Netplan configuration after the settings are applied in order for them to take effect. You can even double check the settings via the GUI.



Time to test the connection via the command line.

```
spy2@Laptop:/etc/netplan$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b5:c7:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.30/24 brd 192.168.1.255 scope global noprefixroute enp0s3
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:feb5:c792/64 scope link
       valid_lft forever preferred_lft forever
spy2@Laptop:/etc/netplan$ ip route
default via 192.168.1.1 dev enp0s3 proto static metric 100
169.254.0.0/16 dev enp0s3 scope link metric 1000
192.168.1.0/24 dev enp0s3 proto kernel scope link src 192.168.1.30 metric 100
spy2@Laptop:/etc/netplan$
```

Notice how there is no more mention of dynamic for the enp0s3 interface anymore. That means that the static ip configuration worked and DHCPis not being used.

```
spy2@Laptop:/etc/netplan$ ping google.com
PING google.com (142.250.176.14) 56(84) bytes of data.
64 bytes from lax17s51-in-f14.1e100.net (142.250.176.14): icmp_seq=1 ttl=116 time=39
.6 ms
64 bytes from lax17s51-in-f14.1e100.net (142.250.176.14): icmp_seq=2 ttl=116 time=15
.4 ms
64 bytes from lax17s51-in-f14.1e100.net (142.250.176.14): icmp_seq=3 ttl=116 time=23
.0 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 15.366/26.004/39.631/10.129 ms
spy2@Laptop:/etc/netplan$ nslookup x.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   x.com
Address: 162.159.140.229

spy2@Laptop:/etc/netplan$
```

6. Do a DHCP configuration.

I configured Netplan with a static ip address and will now demonstrate how DHCP can be configured. In order to do this I will disconnect the old profile and delete the netplan_static.yaml file, creating a netplan_dhcp.yaml file instead.

```
spy2@Laptop:/etc/netplan$ sudo touch netplan_dhcp.yaml
spy2@Laptop:/etc/netplan$
```

```
spy2@Laptop: /etc/netplan
  GNU nano 6.2                 netplan_dhcp.yaml *
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    enp0s3:
      dhcp4: true
```

All you need to do is set dhcp4 to true. Netplan will handle the rest of the process and will tell the interface to ask for network configuration from a DHCP server. In my case that will be my home router.

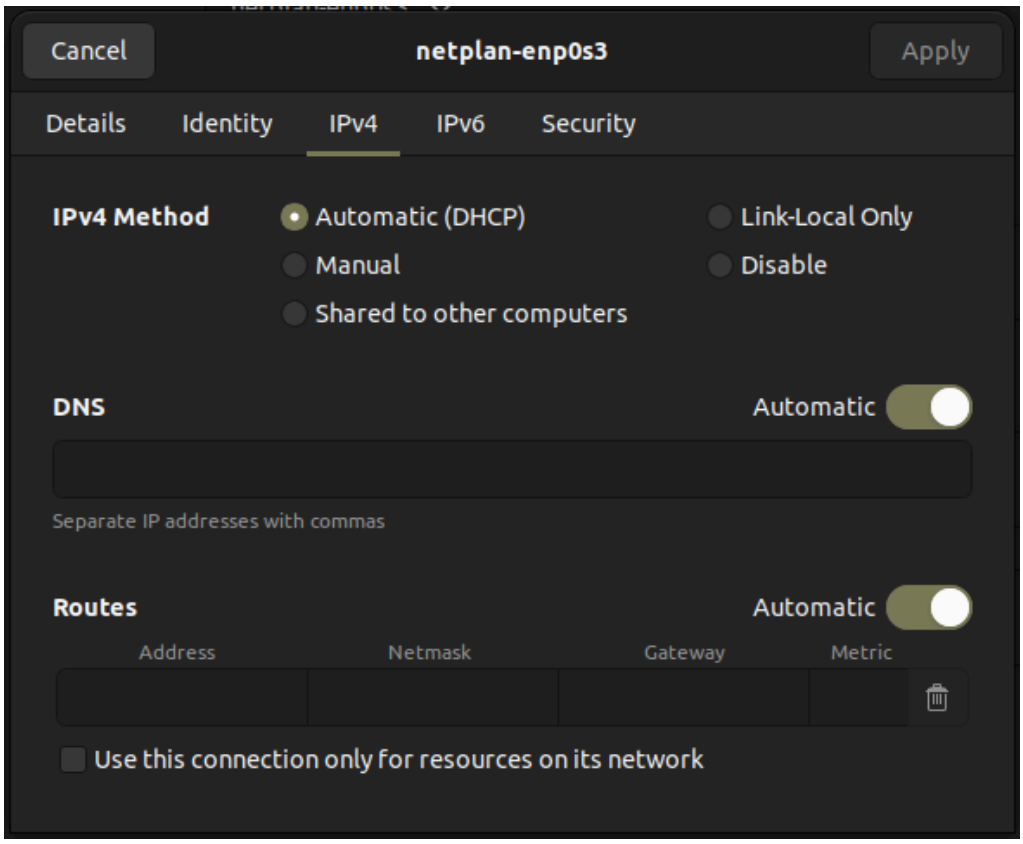
```
spy2@Laptop:/etc/netplan$ ls -l
total 8
-rw------- 1 root root 104 Feb 20  2024 01-network-manager-all.
yaml
-rw-r--r-- 1 root root  92 Mar 12 14:51 netplan_dhcp.yaml
spy2@Laptop:/etc/netplan$ sudo chmod 600 netplan_dhcp.yaml
spy2@Laptop:/etc/netplan$
```

You have to change the permissions for the configuration file to prevent other users other than root from reading and writing the file. Netplan is really picky about file permissions.

7. Safely test the new DHCP configuration.

```
spy2@Laptop:/etc/netplan$ sudo netplan try
Do you want to keep these settings?


Press ENTER before the timeout to accept the new configuration


Changes will revert in 111 seconds
```

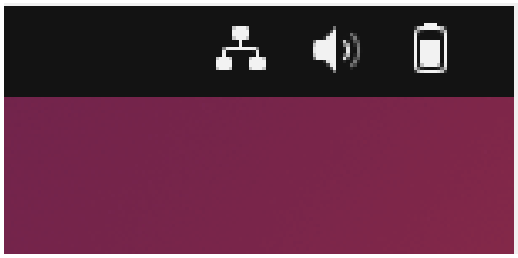
Looking at the GUI you can see that the DHCP configuration worked since the netplan profile is using DHCP.



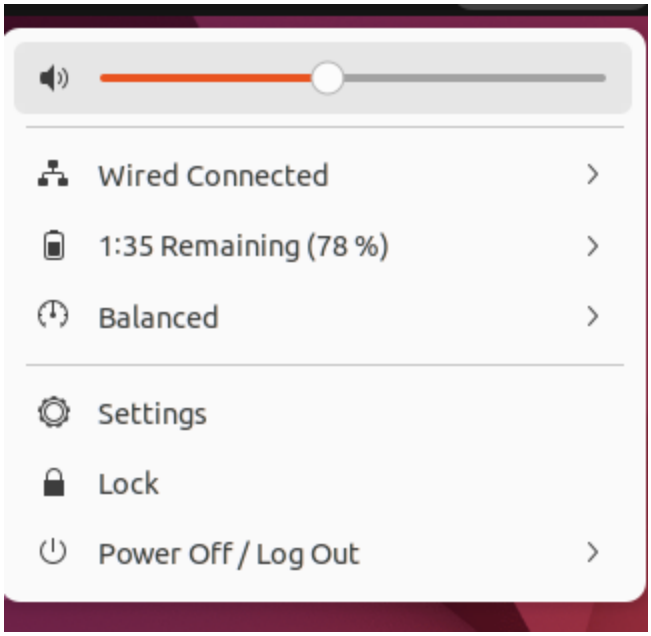
Section #3 GUI Network Configuration

1. Open the Network Settings from the system menu.

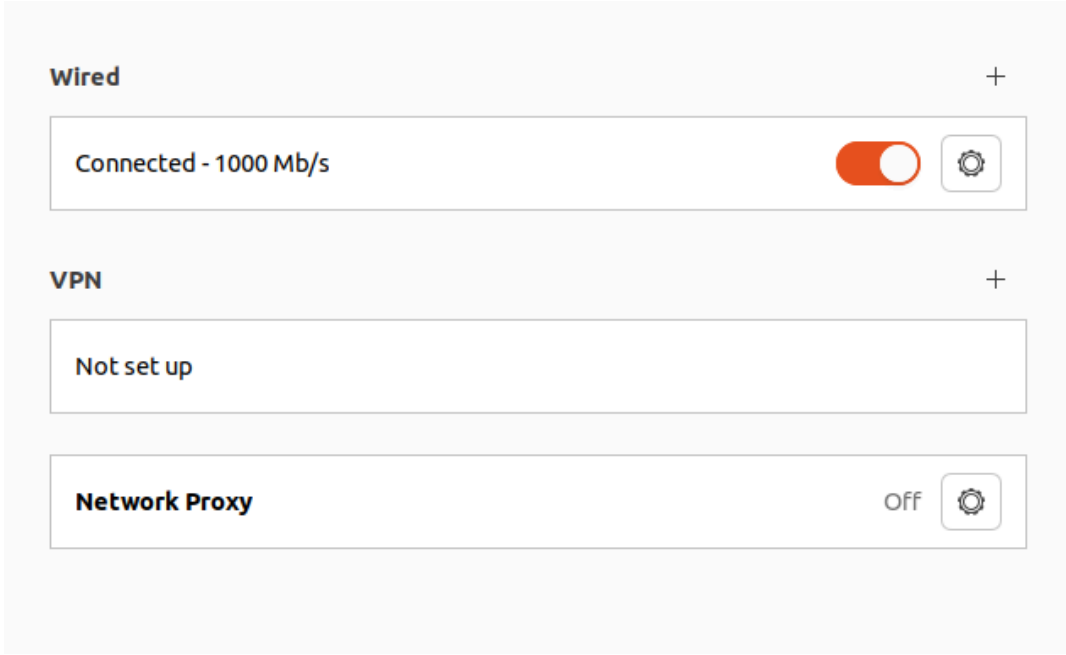
The network settings can be accessed from the GUI on the top right corner. It will look like this:



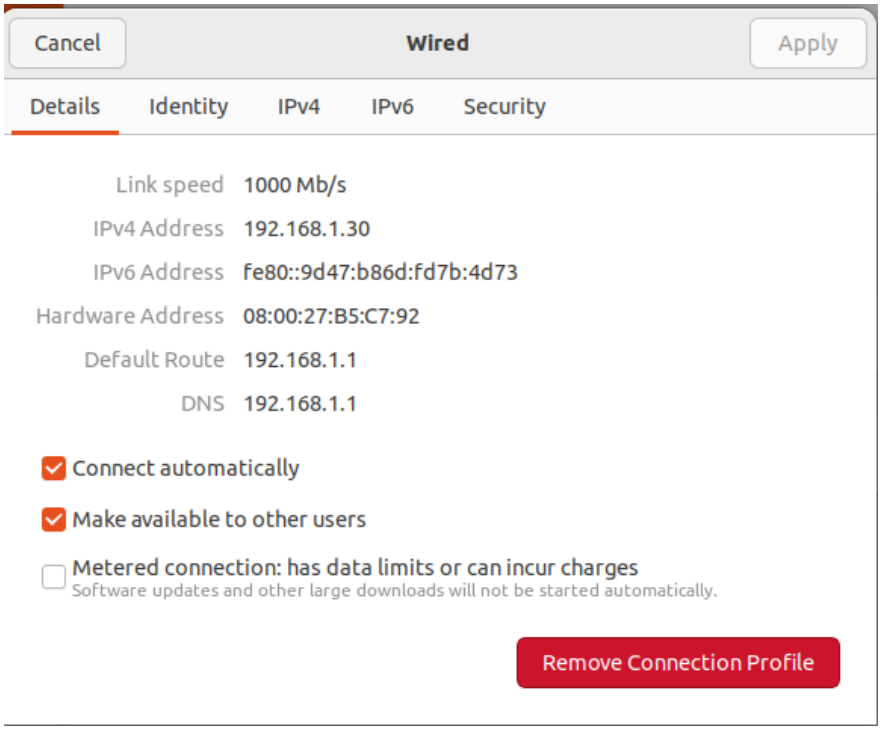
You will click on the three squares which will open to the following:



Since I have a wired (ethernet) connection I will click on the wired connection option and go to the wired settings.

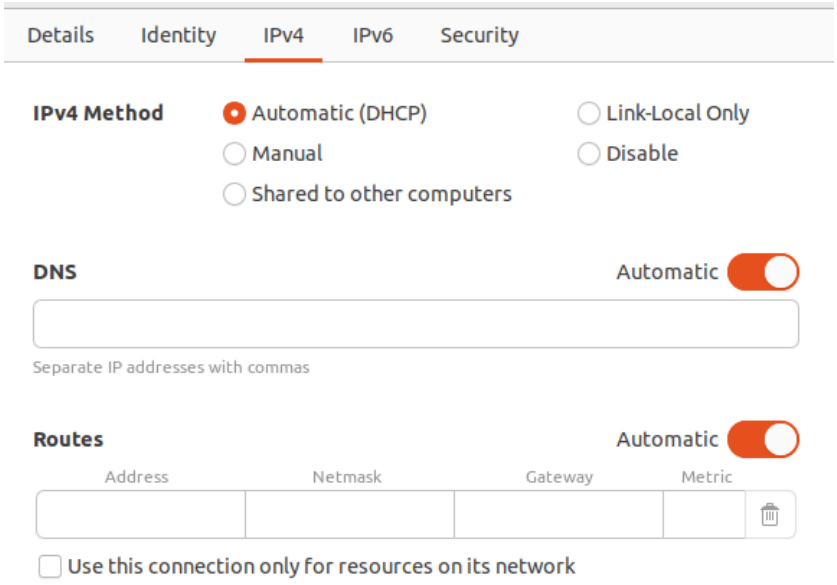


To edit the wired network settings I can click on the wheel.



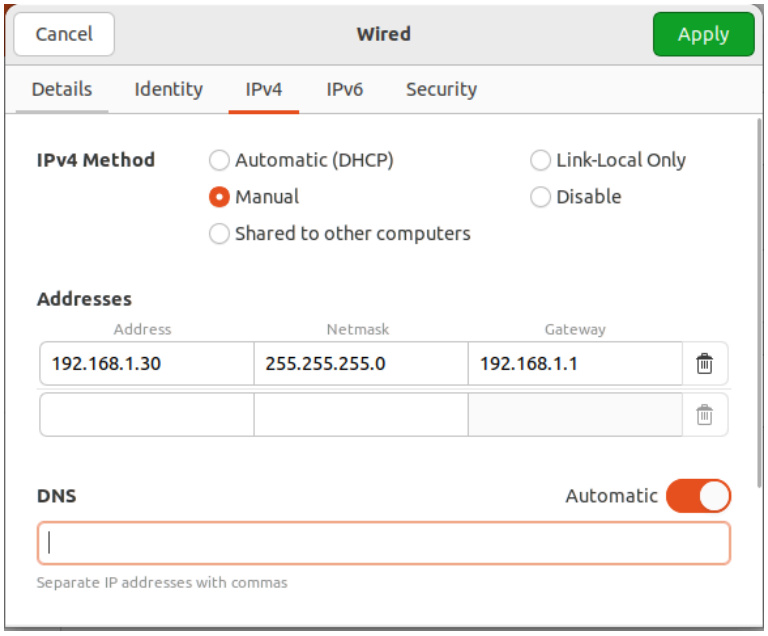
2. Set a static IP address using the GUI.

My current network configuration is the following:



This Ubuntu VM uses DHCP, receiving an IP address from my router which acts as a DHCP server. If I wanted to give this machine a static IP address via the GUI it is really simple to do so.

All I have to do is click on manual instead of automatic. I then have to put the manual IP address I want to assign, the subnet mask , and the gateway (router) address.



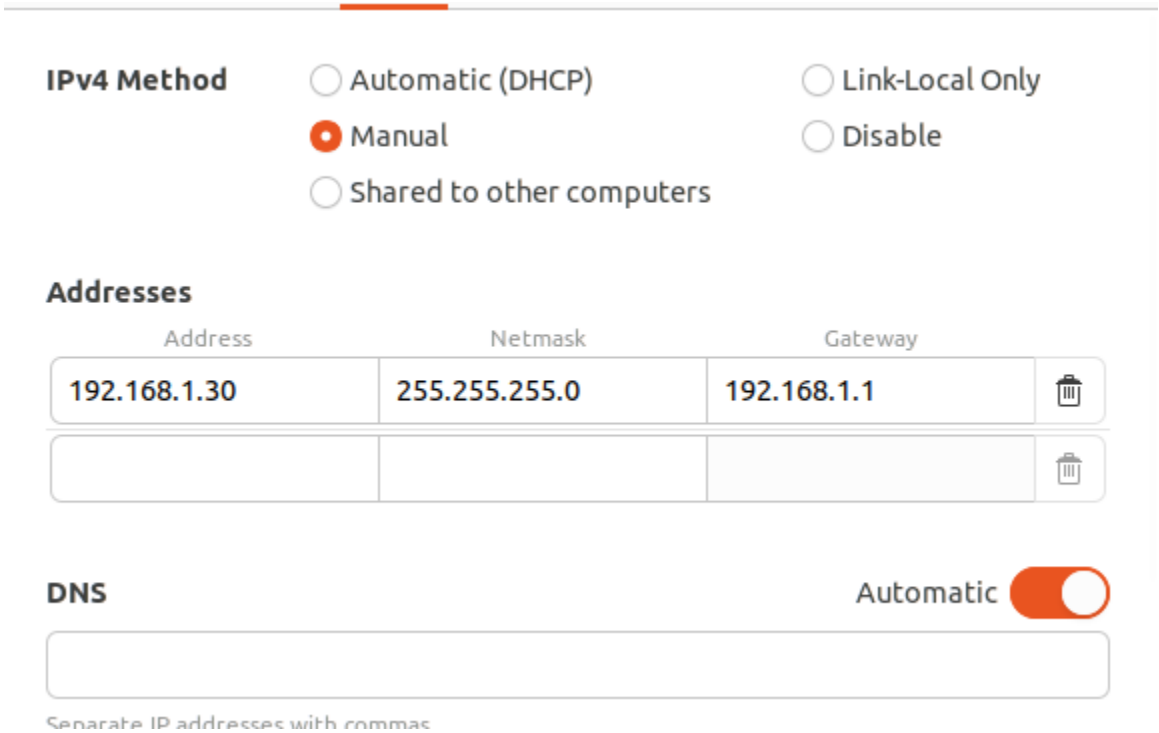
Once I am done I simply click apply and I am good.

```
spy2@Laptop:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b5:c7:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.30/24 brd 192.168.1.255 scope global dynamic noprefixroute enp0s3
       valid_lft 85267sec preferred_lft 85267sec
    inet6 fe80::9d47:b86d:fd7b:4d73/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
spy2@Laptop:~$ ping google.com -c 8
PING google.com (142.250.188.238) 56(84) bytes of data.
64 bytes from lax31s15-in-f14.1e100.net (142.250.188.238): icmp_seq=1 ttl=57 time=35.7 ms
64 bytes from lax31s15-in-f14.1e100.net (142.250.188.238): icmp_seq=2 ttl=57 time=23.7 ms
64 bytes from lax31s15-in-f14.1e100.net (142.250.188.238): icmp_seq=3 ttl=57 time=20.5 ms
64 bytes from lax31s15-in-f14.1e100.net (142.250.188.238): icmp_seq=4 ttl=57 time=125 ms
64 bytes from lax31s15-in-f14.1e100.net (142.250.188.238): icmp_seq=5 ttl=57 time=36.5 ms
64 bytes from lax31s15-in-f14.1e100.net (142.250.188.238): icmp_seq=6 ttl=57 time=28.6 ms
64 bytes from lax31s15-in-f14.1e100.net (142.250.188.238): icmp_seq=7 ttl=57 time=22.1 ms
64 bytes from lax31s15-in-f14.1e100.net (142.250.188.238): icmp_seq=8 ttl=57 time=33.1 ms

--- google.com ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7032ms
rtt min/avg/max/mdev = 20.526/40.597/124.544/32.246 ms
spy2@Laptop:~$ nslookup google.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.188.238
Name:   google.com
Address: 2607:f8b0:4007:809::200e
```

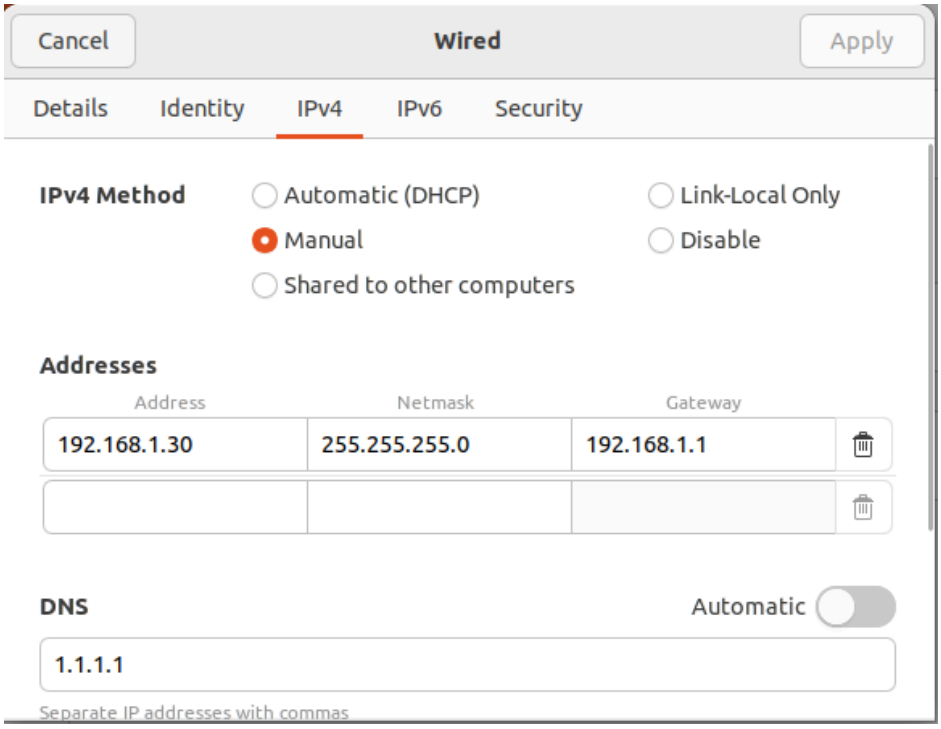
I verified that the change occurred and it did. My IP address is 192.168.1.30, I am able to reach other systems on the internet, and I am able to resolve host names with DNS.

3. Configure DNS settings through the graphical interface.

I only configured a static IP address and the gateway address. I can still assign a custom DNS server for my Ubuntu system to use. I will use Cloudflare's 1.1.1.1 DNS server for my system.



DNS is currently automatically configured. I will now manually assign 1.1.1.1 as the DNS server.



Now I will test my configuration.

```
spy2@Laptop:~$ ping google.com
PING google.com (172.217.14.110) 56(84) bytes of data.
64 bytes from lax31s01-in-f14.1e100.net (172.217.14.110): icmp_seq=1 ttl=116 time=59.2 ms
64 bytes from lax31s01-in-f14.1e100.net (172.217.14.110): icmp_seq=2 ttl=116 time=21.3 ms
64 bytes from lax31s01-in-f14.1e100.net (172.217.14.110): icmp_seq=3 ttl=116 time=16.9 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2016ms
rtt min/avg/max/mdev = 16.850/32.471/59.237/19.014 ms
spy2@Laptop:~$ nslookup google.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   google.com
Address: 172.217.14.110
Name:   google.com
Address: 2607:f8b0:4007:819::200e
```

I am able to resolve hostnames to IP addresses!