

Networking

We need to set a static IP for our server so that we can always access it from the same address.

Your server should be plugged in directly to the router with a Cat6 Ethernet cable.

Before we do this, we should verify our current internet connection settings and write down some important details for later. We need to know the computer's current local network IP address and the address for our internet gateway.

Collecting Information

First, we need to install a utility that can quickly output this information for us:

```
sudo apt install -y net-tools
```

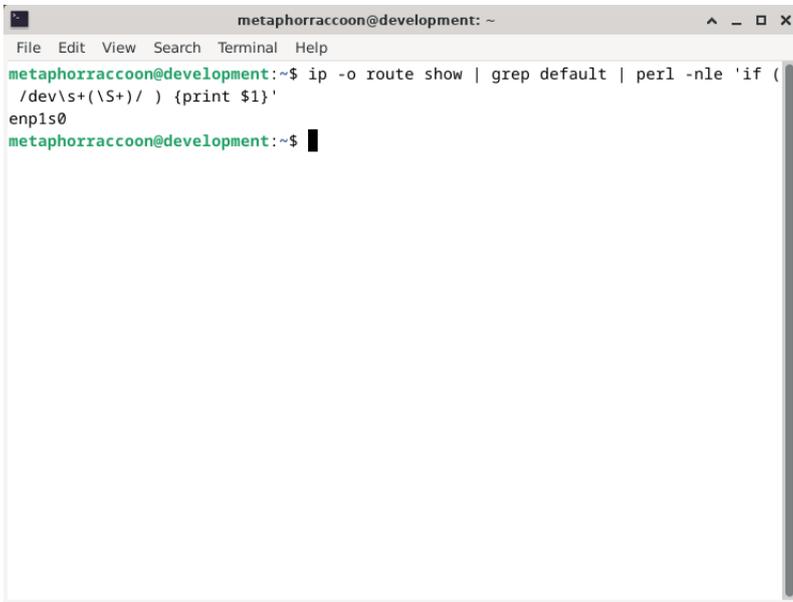
Default Network Device

Once this completes, we will need to run a command to find out what our default network device is:

```
ip -o route show | grep default | perl -nle 'if ( /dev\s+(\S+)/ ) {print $1}'
```

This will return the device name about our default network interface which should be our wired connection.

You may want a piece of paper to write down notes.



```
metaphorraccoon@development: ~  
File Edit View Search Terminal Help  
metaphorraccoon@development:~$ ip -o route show | grep default | perl -nle 'if (  
/dev\s+(\S+)/) {print $1}'  
enp1s0  
metaphorraccoon@development:~$
```

This shows that our default network interface is named **enp1s0**. Write this down for later.

Network Device

enp1s0

Gateway Address

Next, we will retrieve our network gateway address using a similar command:

```
ip -o route show | grep default | perl -nle 'if ( /via\s+(\S+)/) {print $1}'
```



```
metaphorraccoon@development: ~  
File Edit View Search Terminal Help  
metaphorraccoon@development:~$ ip -o route show | grep default | perl -nle 'if (  
/via\s+(\S+)/) {print $1}'  
192.168.68.1  
metaphorraccoon@development:~$
```

This shows that our current network gateway is '192.168.68.1'. Write this down for later.

Network Device

enp1s0

Gateway

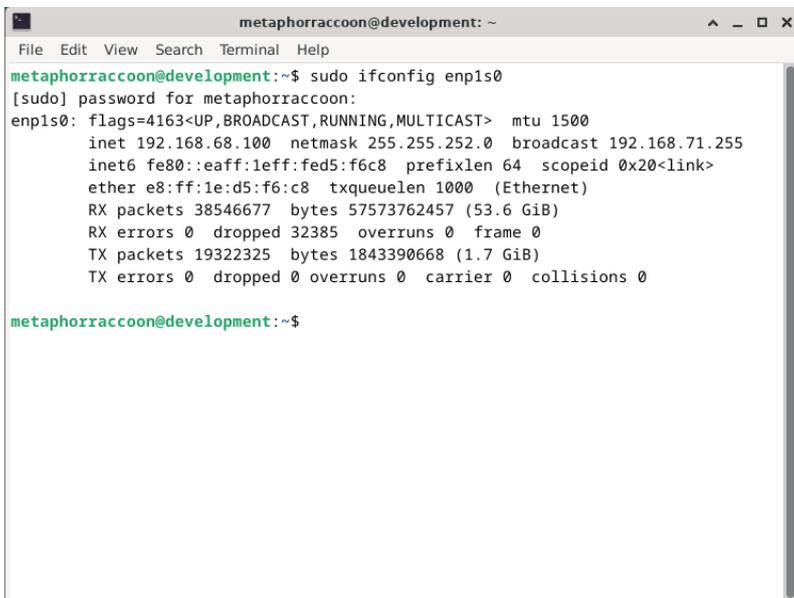
192.168.68.1

IP Address

Now, we will run a command that can tell us other required information about our current connection. Run the command using the device identifier we retrieved earlier:

```
sudo ifconfig enp0s1
```

This output provides the rest of the information we need to configure our IP address.



```
metaphorrraccoon@development: ~  
File Edit View Search Terminal Help  
metaphorrraccoon@development:~$ sudo ifconfig enp1s0  
[sudo] password for metaphorrraccoon:  
enp1s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.68.100 netmask 255.255.252.0 broadcast 192.168.71.255  
    inet6 fe80::eaff:1eff:fed5:f6c8 prefixlen 64 scopeid 0x20<link>  
    ether e8:ff:1e:d5:f6:c8 txqueuelen 1000 (Ethernet)  
    RX packets 38546677 bytes 57573762457 (53.6 GiB)  
    RX errors 0 dropped 32385 overruns 0 frame 0  
    TX packets 19322325 bytes 1843390668 (1.7 GiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
metaphorrraccoon@development:~$
```

This shows our netmask, as well as our current IP address under 'inet'. We need to write these down for later.

Network Device

enp1s0

Gateway

192.168.68.1

IP Address

192.168.68.100

Netmask

255.255.252.0

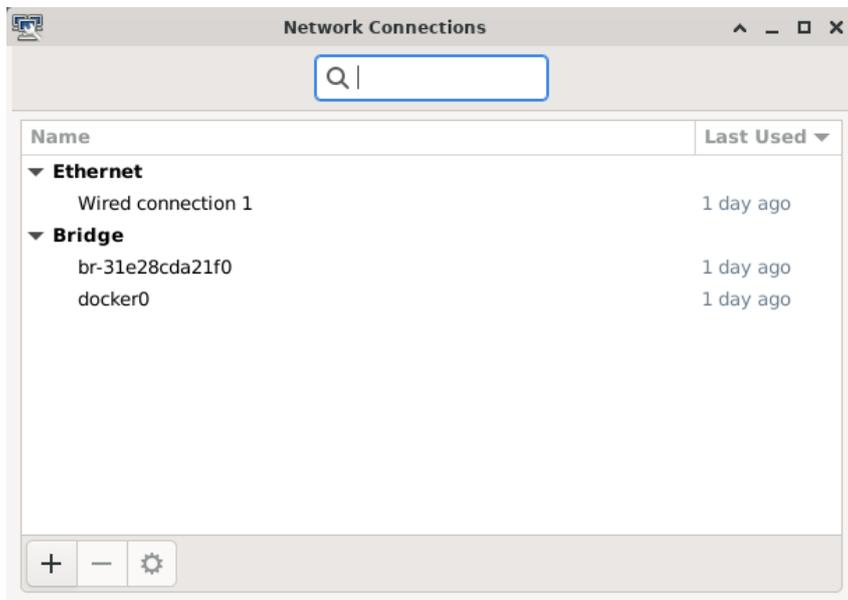
When connecting to the World Wide Web, our router acts as a gateway for all other devices on our local network. The Gateway address is the IP address of our router and can be used by other devices to request their own IP address. Most often, this happens automatically through a process called DHCP where an available IP address is automatically assigned. We will be changing this so that our computer always requests the same IP address when connecting to the internet.

Configuring Connection

Now that we have this information, we can return to the terminal. Run the command:

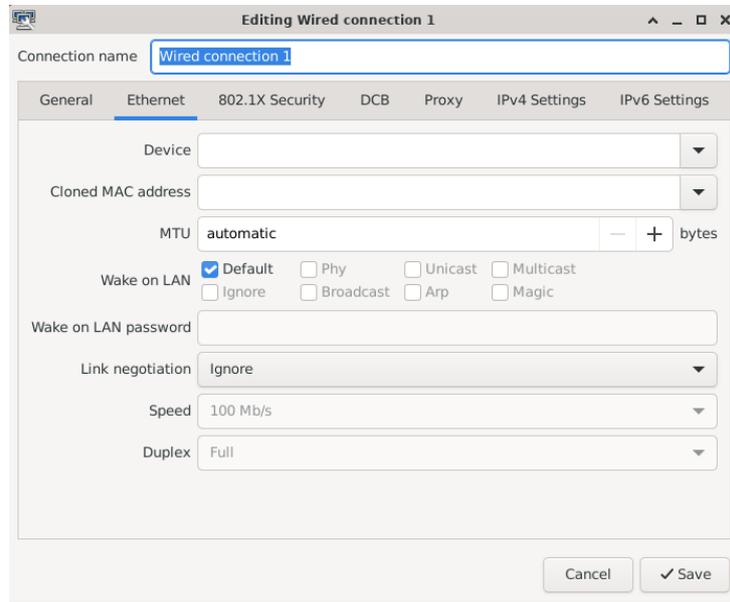
```
sudo nm-connection-editor
```

This opened the Debian Network Connections Manager as the root administrator account so that we can change the internet settings.

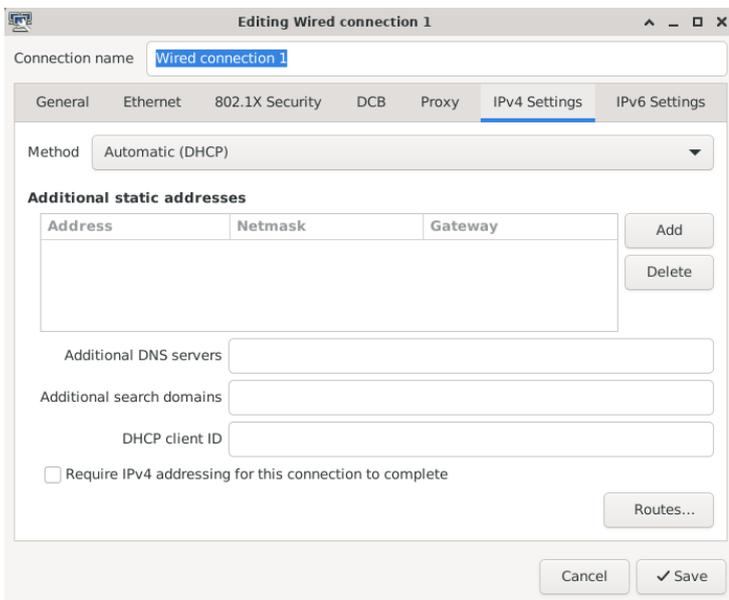


From the Network Connections manager, we can see that we have three network connections. One is an Ethernet connection, which is our direct connection to our router and the outside internet. The other two connections are Bridge connections, which are used by Docker to share internet with containers while controlling their ability to communicate directly with each other.

We will select "Wired connection 1" and the 'gear' icon to edit the settings.



This pulls up all configuration options for our wired internet connection in one place. We will be navigating to the 'IPv4 Settings' tab to manually configure our IP.



Currently, we are set to automatically get an available IP address from the router through DHCP. We will be changing the Method to 'Manual'. This will allow us to set a static IP address that will stay the same between computer and router power cycles. We will hit the 'Add' button to add a new field we can edit.

Network Device	enp1s0
Gateway	192.168.68.1
IP Address	192.168.68.100
Netmask	255.255.252.0
DNS servers	1.1.1.1, 1.0.0.1

Finally, we can enter all of the information we've been collecting. DNS servers are used to translate domain names, like adafruit.com, into their public IP address so we can access their server. We will be using the [Cloudflare DNS](https://www.cloudflare.com/dns/), but other recommendations include [Quad9](https://www.quad9.net/) or [OpenDNS](https://openDNS.com/).

Editing Wired connection 1

Connection name: Wired connection 1

General Ethernet 802.1X Security DCB Proxy IPv4 Settings IPv6 Settings

Method: Manual

Addresses

Address	Netmask	Gateway
192.168.68.100	255.255.252.0	192.168.68.1

DNS servers: 1.1.1.1, 1.0.0.1

Search domains:

DHCP client ID:

Require IPv4 addressing for this connection to complete

Buttons: Add, Delete, Routes..., Cancel, Save

Now, we can hit 'Save' to commit the settings.

Reboot the computer to ensure everything is working as intended.

Verify Connection

After restarting your computer, open up the terminal. We will now run two commands to ensure the internet is working and we have the correct local IP address.

```
ping -c 5 1.1.1.1
```

This will ensure that our computer has access to the Cloudflare DNS server. We will consider the test has passed if the command reports '0% packet loss'.

```
metaphorraccoon@development: ~  
File Edit View Search Terminal Help  
metaphorraccoon@development:~$ ping -c 5 1.1.1.1  
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.  
64 bytes from 1.1.1.1: icmp_seq=1 ttl=59 time=13.6 ms  
64 bytes from 1.1.1.1: icmp_seq=2 ttl=59 time=20.8 ms  
64 bytes from 1.1.1.1: icmp_seq=3 ttl=59 time=21.7 ms  
64 bytes from 1.1.1.1: icmp_seq=4 ttl=59 time=19.5 ms  
64 bytes from 1.1.1.1: icmp_seq=5 ttl=59 time=20.5 ms  
  
--- 1.1.1.1 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4008ms  
rtt min/avg/max/mdev = 13.583/19.214/21.724/2.903 ms  
metaphorraccoon@development:~$
```

We can now run one final command to verify our computer is using the static IP address we configured:

```
ip -o route show | grep default
```

This will show an overview our default internet connection and we can see that it is now labeled as a 'static' connection.

```
metaphorraccoon@development: ~  
File Edit View Search Terminal Help  
metaphorraccoon@development:~$ ip route show | grep 'default'  
default via 192.168.68.1 dev enp1s0 proto static metric 100  
metaphorraccoon@development:~$
```

You now have a computer with a static IP address that will always stay the same.

Revision #11

Created 14 February 2025 09:07:14 by metaphorraccoon

Updated 12 May 2025 08:51:40 by metaphorraccoon