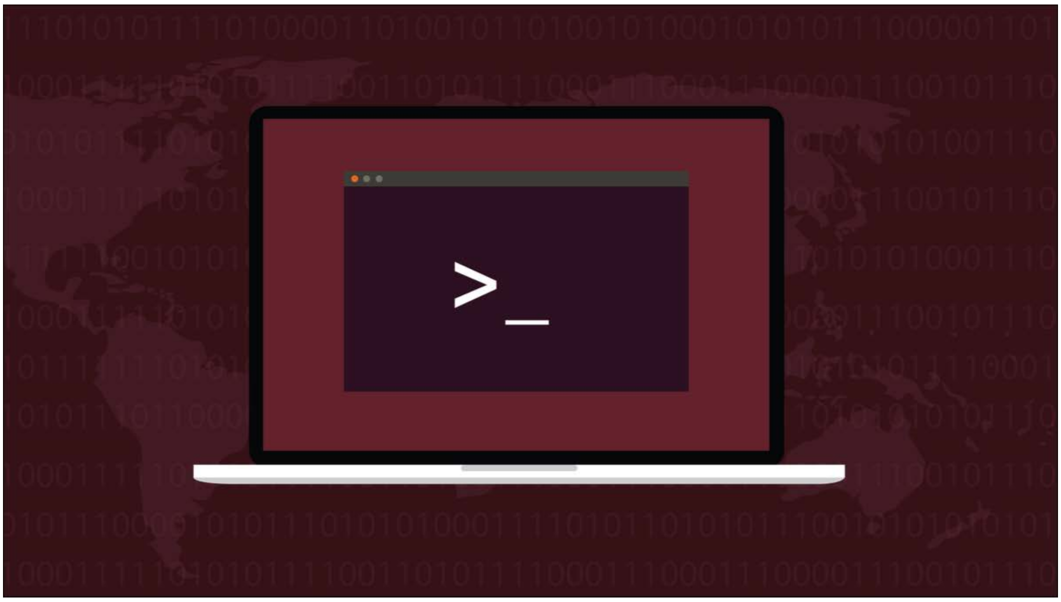
How-To Geek

How to Set a Static IP Address in Ubuntu

DAVE MCKAY @thegurkha
OCT 17, 2022, 1:00 PM EST | 6 MIN READ



fatmawati achmad zaenuri/Shutterstock.com

Key Takeaway

After gathering your connection name, subnet mask, and default gateway, you can set a static IP address in the terminal using the nmcli command. Or, in the GNOME desktop, open your connection settings and click the + icon, then enter the info for your static IP address there.

Your home network relies on IP addresses to route data between devices, and sometimes on reconnecting to the network a device's address can change. Here's how to give an Ubuntu Linux computer a permanent IP address that survives reboots.

Table of Contents

Dynamic and Static IP Addresses

Everything on your network home network, whether it’s using a wired connection or Wi-Fi, has an IP address. IP stands for Internet Protocol. An IP address is a sequence of four numbers separated by three dots. Each IP address that is unique within that network.

IP addresses act as numeric labels. Your router uses these labels to send data between the correct devices. Usually, your router assigns IP addresses. It knows which IP addresses are in use and which are free. When a new device connects to the network, it requests an IP address and the router allocates one of the unused IP addresses. This is called DHCP, or dynamic host configuration protocol.

When a device is restarted or powered off and on, it may receive its old IP address once more, or it might be allocated a new IP address. This is normal for DHCP and it doesn’t affect the normal running of your network. But if you have a server or some other computer that you need to be able to reach by its IP address, you’ll run into problems if its IP address doesn’t survive power downs or reboots.

RELATED
What Is DHCP (Dynamic Host Configuration Protocol)?

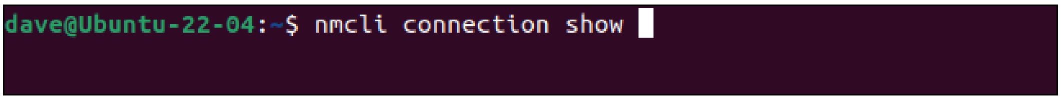
Pinning a specific IP address to a computer is called allocating a *static* IP address. A static IP address, as its name suggests, isn’t dynamic and it doesn’t change even if the computer is power-cycled.

Setting a Static IP Address in Ubuntu

We’re demonstrating this technique on Ubuntu, but it ought to work on any Linux distribution. The `nmcli` network manager tool was released in 2004, so it should be present on just about any standard distribution.

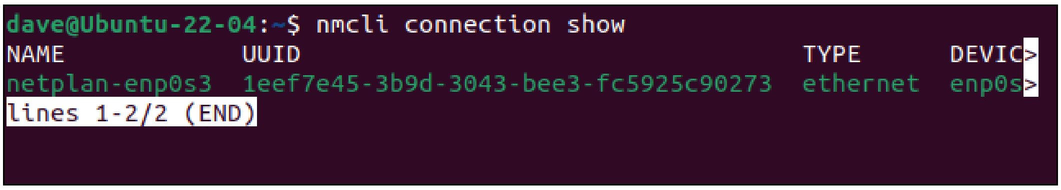
Let’s take a look at the network connections that already exist on the computer. We’re using the `connection` command with the `show` argument.

```
nmcli connection show
```



```
dave@Ubuntu-22-04:~$ nmcli connection show
```

This displays some information about each connection. We only have a single connection configured.



```
dave@Ubuntu-22-04:~$ nmcli connection show
NAME             UUID                                  TYPE      DEVIC>
netplan-enp0s3  1eef7e45-3b9d-3043-bee3-fc5925c90273  ethernet  enp0s>
lines 1-2/2 (END)
```

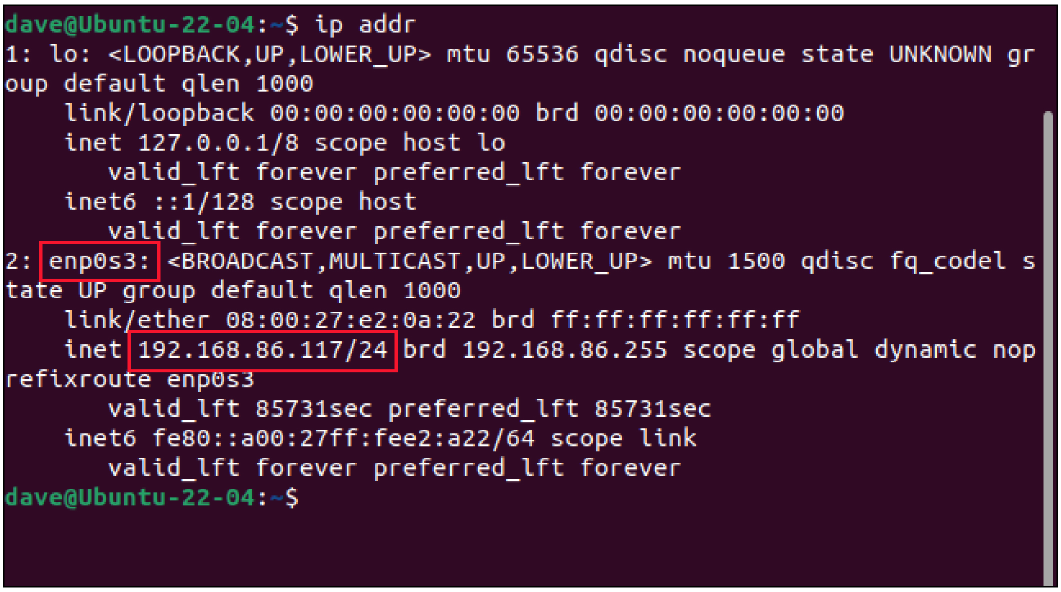
The output is wider than the terminal window. This is the information that we're shown.

```
NAME           UUID                                  TYPE     DEVICE
netplan-enp0s3 1eef7e45-3b9d-3043-bee3-fc5925c90273 ethernet enp0s3
```

- **Name**: Our network connection is called "netplan-enp0s3."
- **UUID**: The universally unique identifier Linux uses to reference this connection internally.
- **Type**: This is an ethernet connection.
- **Device**: This connection is using the "enp0s3" network interface. It's the only network card in this computer.

We can use the `ip` command to discover the IP address this computer is using.

```
ip addr
```



```
dave@Ubuntu-22-04:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN gr
oup 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 s
tate UP group default qlen 1000
    link/ether 08:00:27:e2:0a:22 brd ff:ff:ff:ff:ff:ff
    inet 192.168.86.117/24 brd 192.168.86.255 scope global dynamic nop
refixroute enp0s3
       valid_lft 85731sec preferred_lft 85731sec
    inet6 fe80::a00:27ff:fee2:a22/64 scope link
       valid_lft forever preferred_lft forever
dave@Ubuntu-22-04:~$
```

In the output we can see the "enp0s3" entry, and its current IP address, 192.168.86.117. The "/24" is a shorthand way of saying that this network uses a 255.255.255.0 subnet mask. Take a note of this number, we'll need to use it later.

We need to choose the IP address we're going to set as our static IP address. Obviously, you can't use an IP address that is already in use

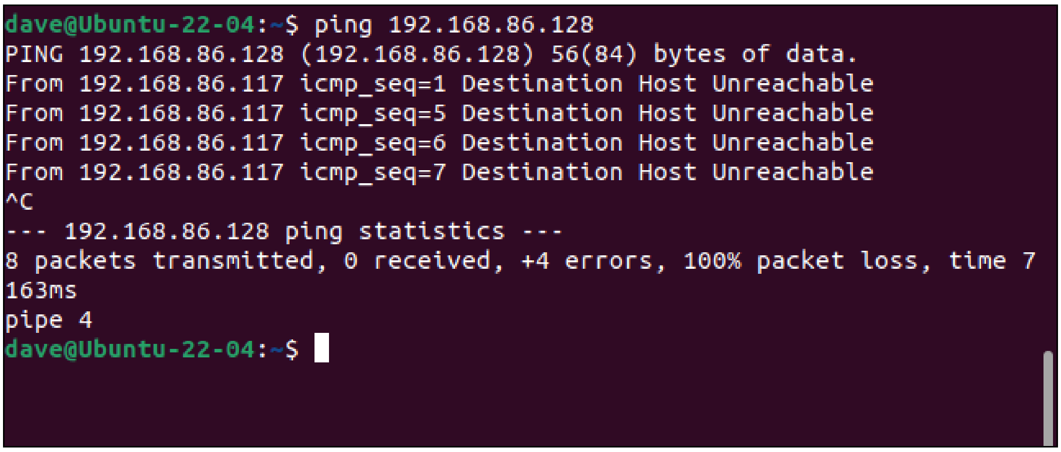
RELATED

How to Calculate Subnet Masks on Linux With ipcalc

by *another* device. One safe way to proceed is to use your current IP address. We know for certain that nothing else is using that IP address.

If we want to use a different IP address, try pinging it. We're going to test whether IP address 192.168.86.128 is in use. If everything else on your network uses DHCP and you get no response to the ping command, it should be safe to use.

```
ping 192.168.86.128
```



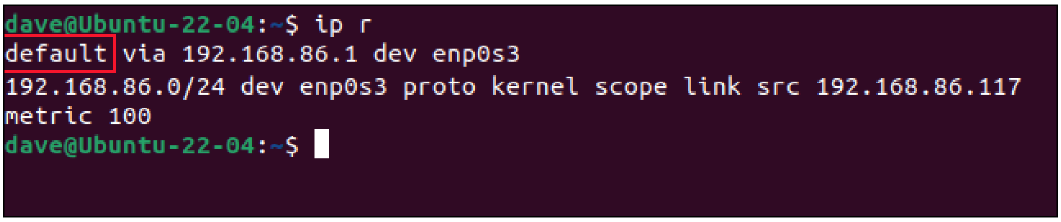
```
dave@Ubuntu-22-04:~$ ping 192.168.86.128
PING 192.168.86.128 (192.168.86.128) 56(84) bytes of data.
From 192.168.86.117 icmp_seq=1 Destination Host Unreachable
From 192.168.86.117 icmp_seq=5 Destination Host Unreachable
From 192.168.86.117 icmp_seq=6 Destination Host Unreachable
From 192.168.86.117 icmp_seq=7 Destination Host Unreachable
^C
--- 192.168.86.128 ping statistics ---
8 packets transmitted, 0 received, +4 errors, 100% packet loss, time 7
163ms
pipe 4
dave@Ubuntu-22-04:~$ 
```

Even if another device had previously used that IP address, it'll be given a new IP address when it next boots up. Nothing is responding to the ping requests. We're clear to go ahead and configure 192.168.86.128 as our new static IP.

We also need to know the IP address of your default gateway, which will usually be your broadband router. We can find this using the ip command and the route option, which we can abbreviate to "r."

RELATED
How to Set the Default Gateway in Linux

```
ip r
```

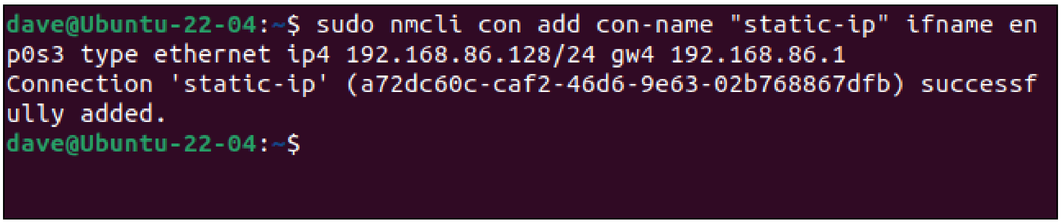


```
dave@Ubuntu-22-04:~$ ip r
default via 192.168.86.1 dev enp0s3
192.168.86.0/24 dev enp0s3 proto kernel scope link src 192.168.86.117
metric 100
dave@Ubuntu-22-04:~$ 
```

The entry that starts with "default" is the route to the default gateway. Its IP address is 192.168.86.1. Now we can start to issue commands to set up our static IP address.

The first command is a long one.

```
sudo nmcli con add con-name "static-ip" ifname enp0s3 type ethernet
```



```
dave@Ubuntu-22-04:~$ sudo nmcli con add con-name "static-ip" ifname en
p0s3 type ethernet ip4 192.168.86.128/24 gw4 192.168.86.1
Connection 'static-ip' (a72dc60c-caf2-46d6-9e63-02b768867dfb) successf
ully added.
dave@Ubuntu-22-04:~$
```

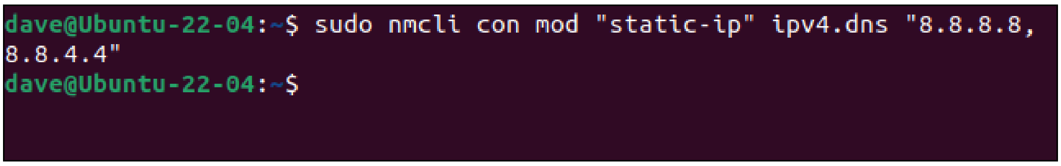
Taken in small chunks, It's not as bad as it looks. We're using `sudo`. The `nmcli` arguments are:

- **con**: Short for "connection."
- **add**: We're going to *add* a connection.
- **con-name "static-ip"**: The name of our new connection will be "static-ip."
- **ifname enp0s3**: The connection will use network interface "enp0s3."
- **type ethernet**: We're creating an ethernet connection.
- **ip4 192.168.86.128/24**: The IP address and subnet mask in classless inter-domain routing notation. This is where you need to use the number you took a note of earlier.
- **gw4 192.168.86.1**: The IP address of the gateway we want this connection to use.

To make our connection a functioning connection, we need to provide a few more details. Our connection exists now, so we're not adding anything, we're modifying settings, so we use the `mod` argument. The setting we're changing is the IPv4 DNS settings. 8.8.8.8 is the IP address of Google's primary public DNS server, and 8.8.4.4 is Google's fallback DNS server.

Note that there is a "v" in "ipv4." In the previous command the syntax was "ip4" without a "v." The "v" needs to be used when you're modifying settings, but not when adding connections.

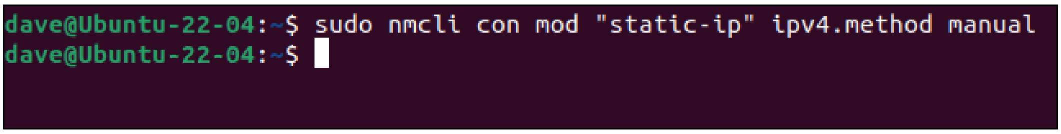
```
nmcli con mod "static-ip" ipv4.dns "8.8.8.8,8.8.4.4"
```



```
dave@Ubuntu-22-04:~$ sudo nmcli con mod "static-ip" ipv4.dns "8.8.8.8,
8.8.4.4"
dave@Ubuntu-22-04:~$
```

To make our IP address static, we need to change the *method* which the IP address obtains its value. The default is "auto" which is the setting for DHCP. We need to set it to "manual."

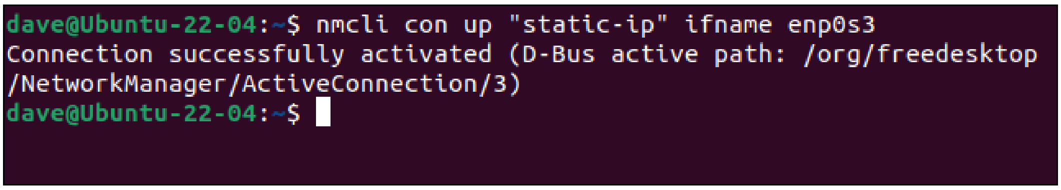
```
nmcli con mod "static-ip" ipv4.method manual
```



```
dave@Ubuntu-22-04:~$ sudo nmcli con mod "static-ip" ipv4.method manual
dave@Ubuntu-22-04:~$ 
```

And now we can start or “bring up” our new connection.

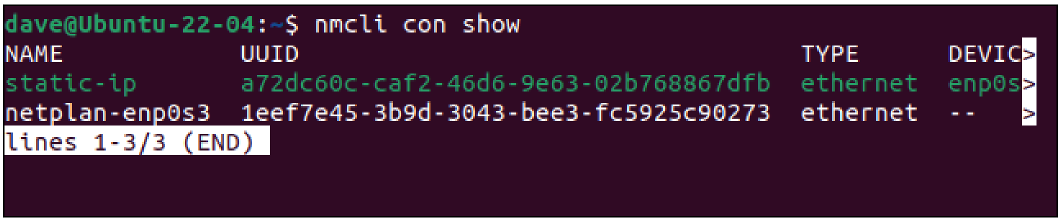
```
nmcli con up "static-ip" ifname enp0s3
```



```
dave@Ubuntu-22-04:~$ nmcli con up "static-ip" ifname enp0s3
Connection successfully activated (D-Bus active path: /org/freedesktop
/NetworkManager/ActiveConnection/3)
dave@Ubuntu-22-04:~$ 
```

We didn’t get any error messages which is great. Lets use `nmcli` to look at our connections once more.

```
nmcli con show
```



```
dave@Ubuntu-22-04:~$ nmcli con show
NAME            UUID                                  TYPE      DEVIC>
static-ip       a72dc60c-caf2-46d6-9e63-02b768867dfb  ethernet  enp0s>
netplan-enp0s3  1eef7e45-3b9d-3043-bee3-fc5925c90273  ethernet  --   >
lines 1-3/3 (END)
```

Here’s the output:

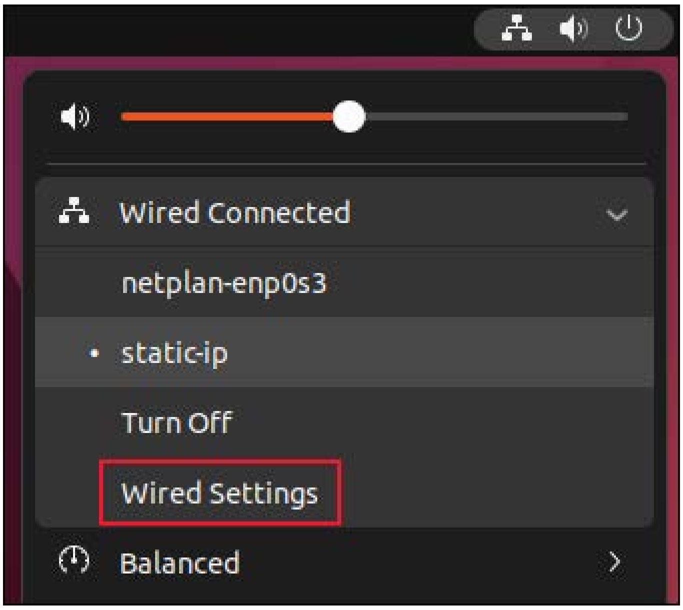
```
NAME           UUID                                 TYPE     DEVICE
static-ip      da681e18-ce9c-4456-967b-63a59c493374 ethernet enp0s3
netplan-enp0s3 1eef7e45-3b9d-3043-bee3-fc5925c90273 ethernet --
```

Our static-ip connection is active and using device “enp0s3.” The existing connection “netplan-enp0s3” is no longer associated with a physical network interface because we’ve pinched “enp0s3” from it.

Using the GNOME Desktop and Applications

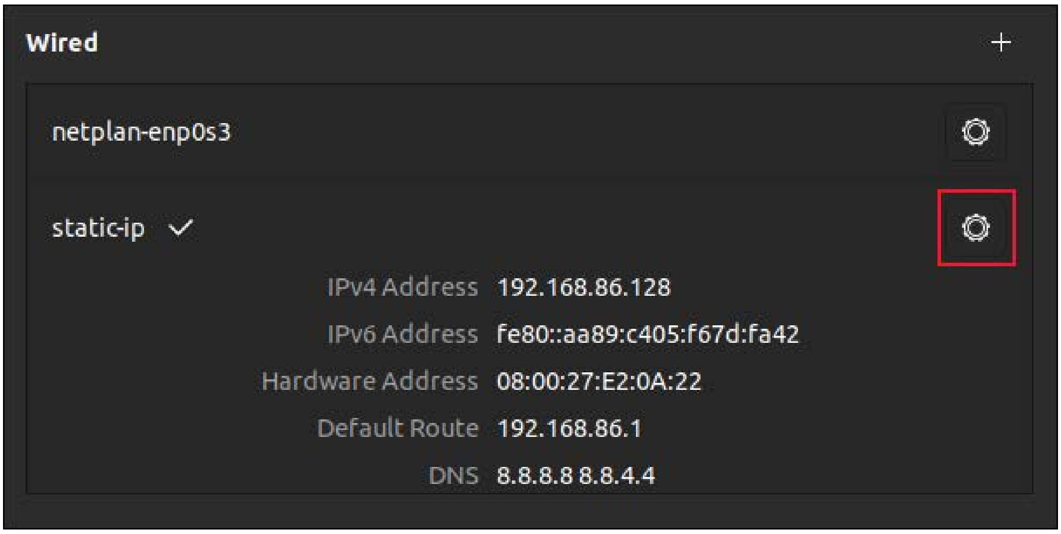
Click the icons at the far-right end of the system bar to show the system menu, then click on the “Wired Connected” menu option. If you’re using a wireless connection, instead click the name of your Wi-Fi network.

The available connections are displayed. A dot indicates which is in use. Our new connection is the active connection. Click the “Wired Settings” or “Wi-Fi Settings” menu option.



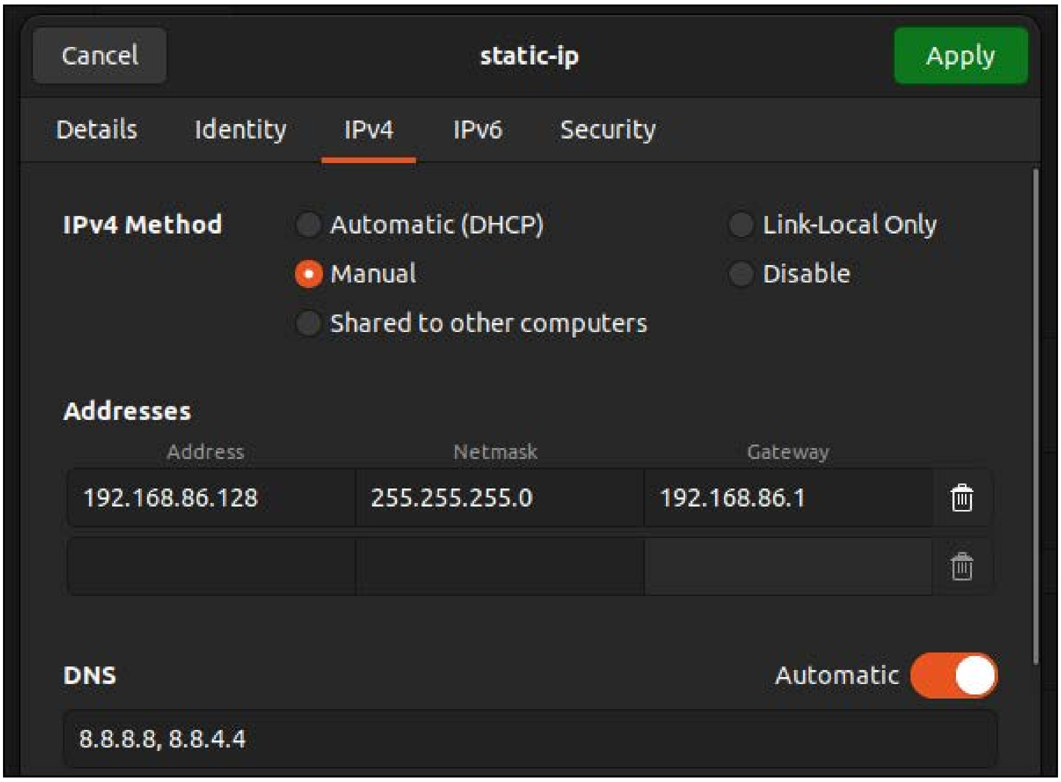
The details of the active connection are displayed. We can see our new “static-ip” connection has the IP address, default gateway, and DNS servers that we set for it. You can swap between the available connections by clicking directly on their names.

Click the cog icon to enter the settings for the “static-ip” connection.



IPv4 Address 192.168.86.128
IPv6 Address fe80::aa89:c405:f67d:fa42
Hardware Address 08:00:27:E2:0A:22
Default Route 192.168.86.1
DNS 8.8.8.8 8.8.4.4

A dialog box opens. Click on the “IPv4” tab.



IPv4 Method

○ Automatic (DHCP) ○ Link-Local Only
● Manual ○ Disable
○ Shared to other computers

Addresses

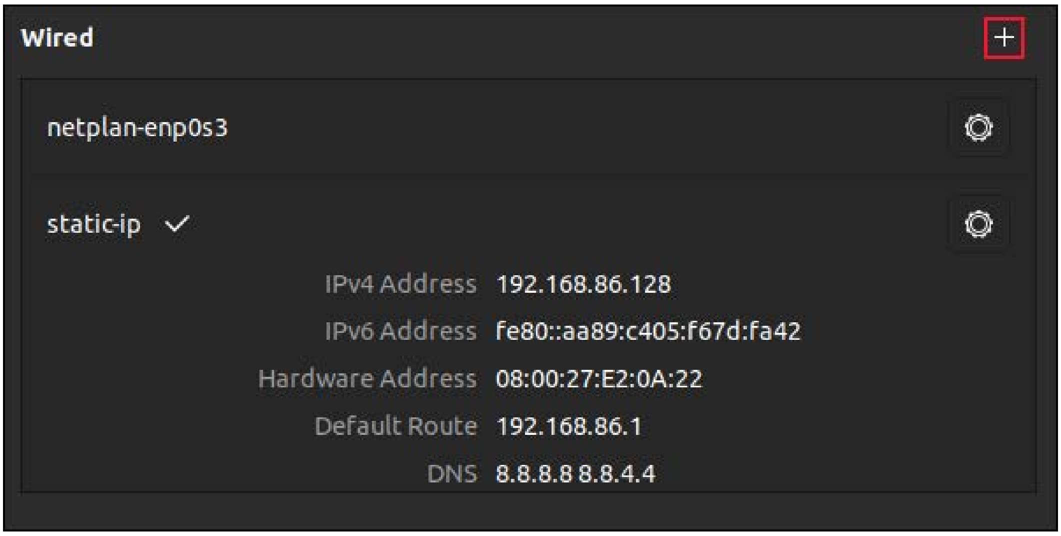
Address	Netmask	Gateway
192.168.86.128	255.255.255.0	192.168.86.1

DNS Automatic ●

8.8.8.8, 8.8.4.4

Because we set our new IP address to be static, the "Manual" radio button is selected. You could change this back to DHCP by selecting the "Automatic (DHCP)" radio button, and clicking the green "Apply" button.

To create a new connection using the "Settings" application, click the "+" icon on the "Networks" page, above the list of wired connections.

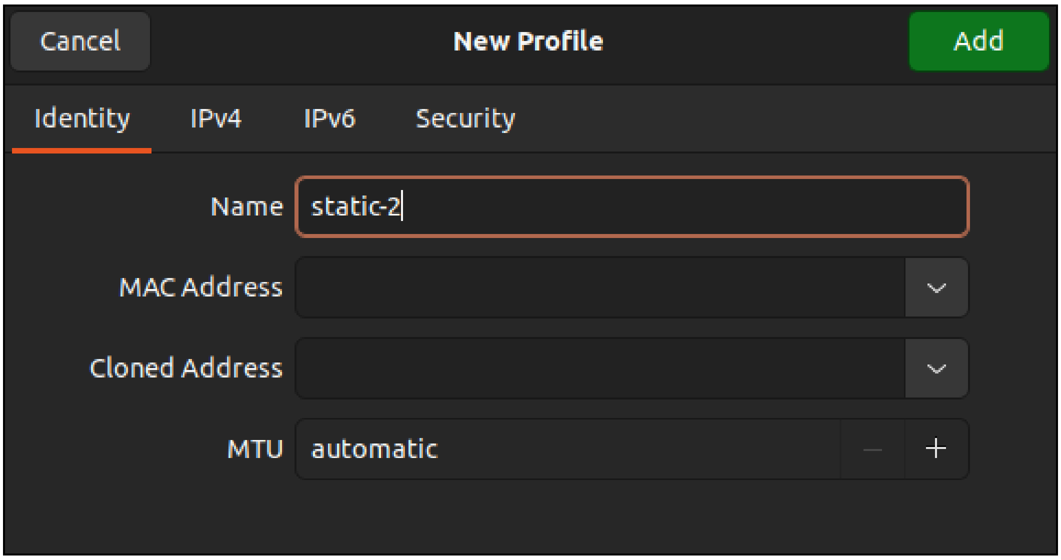


netplan-enp0s3

static-ip ✓

IPv4 Address	192.168.86.128
IPv6 Address	fe80::aa89:c405:f67d:fa42
Hardware Address	08:00:27:E2:0A:22
Default Route	192.168.86.1
DNS	8.8.8.8 8.8.4.4

A dialog appears. We need to provide a name for this connection.



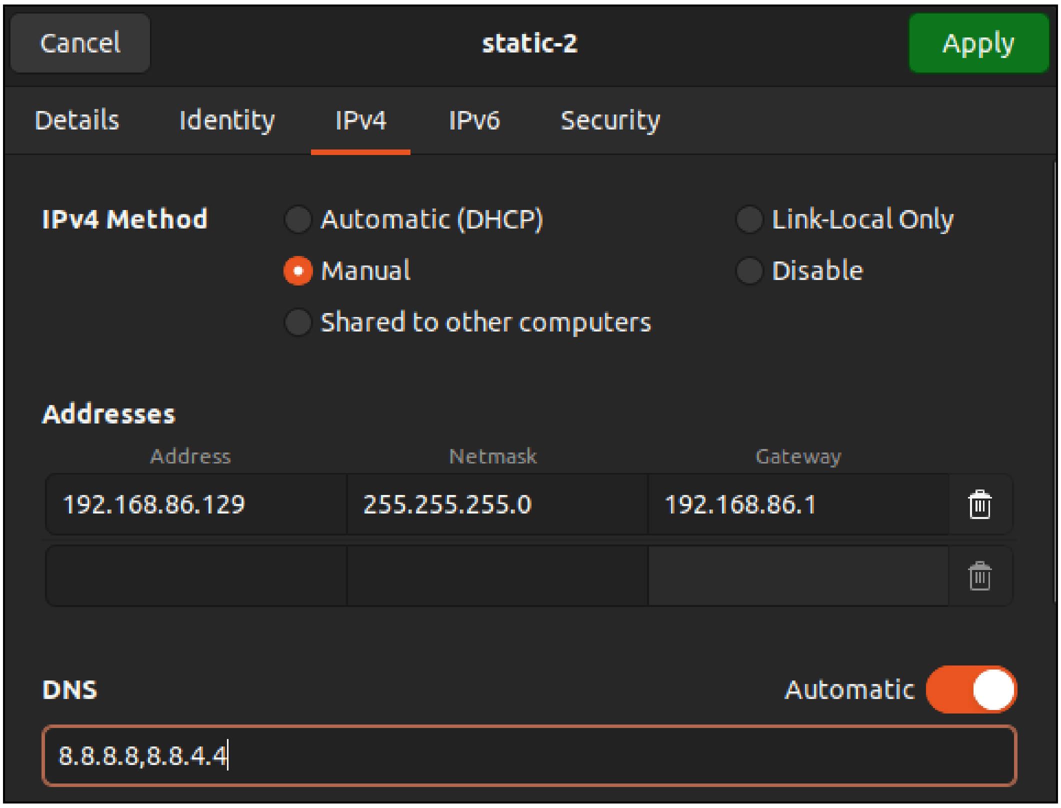
Name static-2

MAC Address

Cloned Address

MTU automatic

We're calling our new connection "static-2." Click the "IPv4" tab.



IPv4 Method
○ Automatic (DHCP)
● Manual
○ Shared to other computers
○ Link-Local Only
○ Disable

Addresses

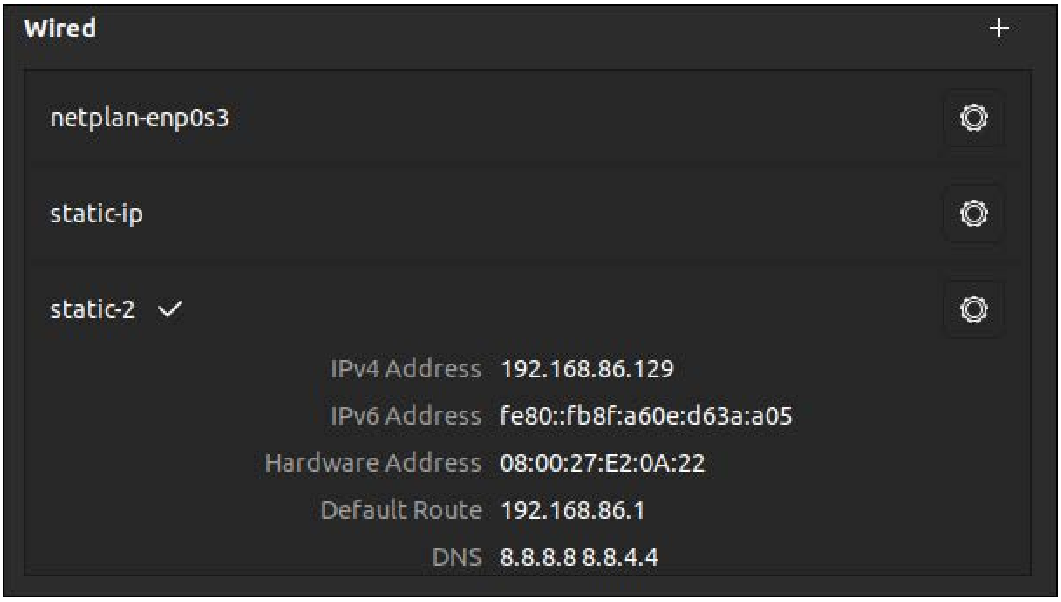
Address	Netmask	Gateway
192.168.86.129	255.255.255.0	192.168.86.1

DNS Automatic

8.8.8.8,8.8.4.4

Select the "Manual" radio button, and complete the "Address", "Netmask", and "Gateway" fields. Also complete the DNS field, and then click the green "Apply" button. Note the comma between the DNS entries.

Our new connection is listed in the "Wired" connections pane.



Wired

netplan-enp0s3

static-ip

static-2 ✓

IPv4 Address 192.168.86.129

IPv6 Address fe80::fb8f:a60e:d63a:a05

Hardware Address 08:00:27:E2:0A:22

Default Route 192.168.86.1

DNS 8.8.8.8 8.8.4.4

Connection Convenience

Using the `nmcli` command or the GNOME desktop and apps, you can hop between network connections very easily and very quickly.

It's more convenient to have a selection of connection profiles and move between them as you need to, rather than to have one that you keep editing. If something goes horribly wrong with the connection you're editing or adding, you can always fall back on one of the existing connections.

RELATED: ***How to Use bmon to Monitor Network Bandwidth on Linux***

DAVE MCKAY

Dave McKay first used computers when punched paper tape was in vogue, and he has been programming ever since. After over 30 years in the IT industry, he is now a full-time technology journalist. During his career, he has worked as a freelance programmer, manager of an international software development team, an IT services project manager, and, most recently, as a Data Protection Officer. His writing has been published by howtogeek.com, cloudsavvyit.com, itenterpriser.com, and opensource.com. Dave is a Linux evangelist and open source advocate.

READ FULL BIO »