Linux Lab: Network Configuration

Linux Command List: Network Configuration

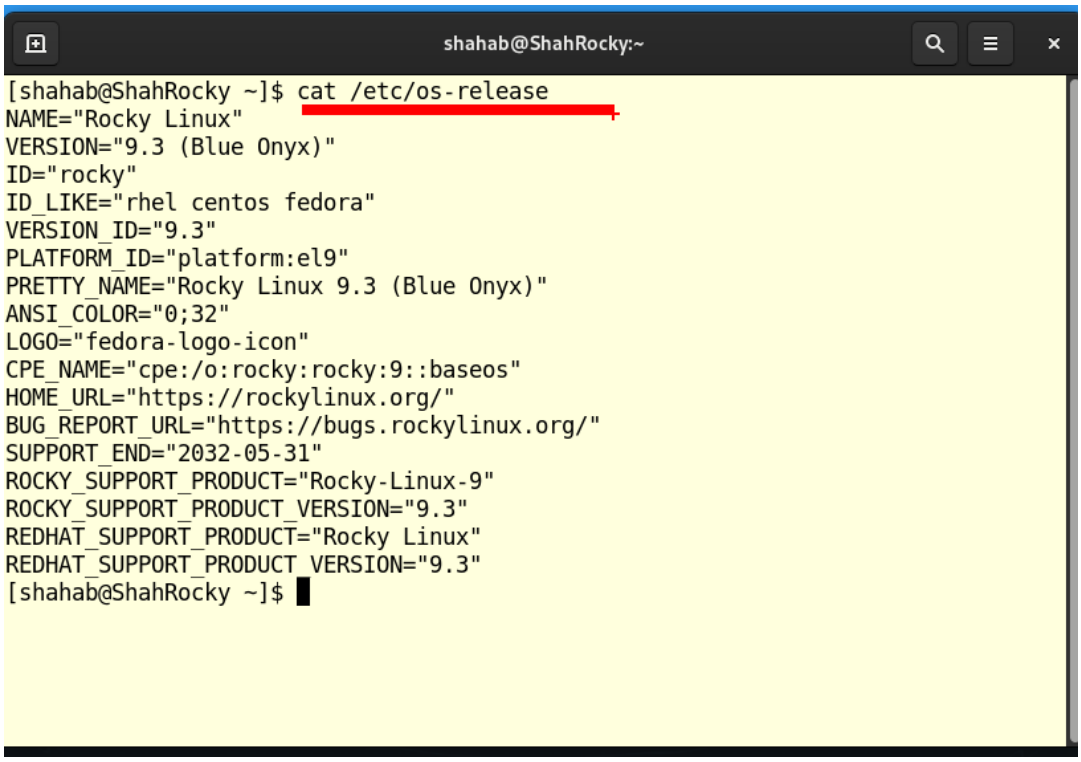
- **`ifconfig`** → Display network interface information → ifconfig eth0
- **`ip addr`** → Manage network interfaces and addresses → ip addr show eth0
- **`nmcli`** → Display network interface information (after NetworkManager installed)
- **`route`** → Display and manipulate routing table → route -n
- **`netstat`** → Display and checks on the open and listening ports of applications → netstat -and
- **`ping`** → Test network connectivity → ping 8.8.8.8
- **`traceroute`** → Trace network path to a host → traceroute google.com
- **`host`** → Get host information → host google.com
- **`dig`** → Perform DNS lookups → dig A google.com
- **`nslookup`** → Lookup domain names and IP addresses → nslookup google.com
- **`hostname`** → Set or get the system hostname → hostname server1
- **`ip link`** → View and manage network links → ip link show eth0
- **`ip address`** → Manage IP addresses on interfaces
- **`uname`** → Prints information about your machine's kernel, name, and hardware
- **`ss, ss -tuln`** → shows all listening TCP and UDP connections

Linux Command List: Network Services

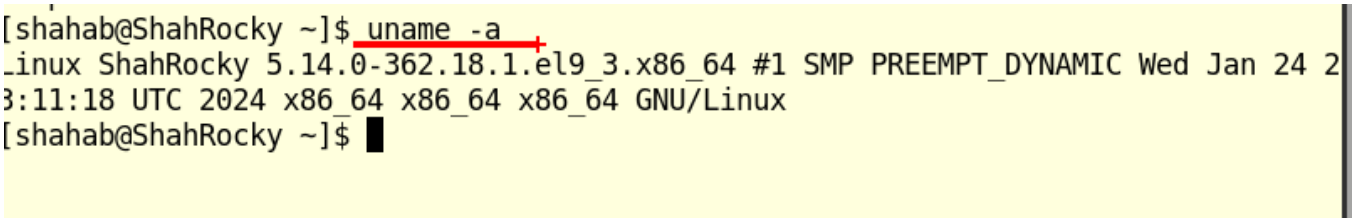
- **`service`** → Manage system services → service ssh start
- **`systemctl`** → Manage systemd services → systemctl enable ssh
- **`sshd`** → SSH daemon → sshd -D
- **`telnet`** → Connect to a remote host → telnet mailserver 25
- **`ftp`** → File transfer protocol → ftp ftp.example.com
- **`scp`** → Securely copy files → scp /etc/hosts user@server:/tmp
- **`wget`** → Download files from the internet → wget http://example.com/file.tar.gz

Rocky Server Linux: Network Interface Config

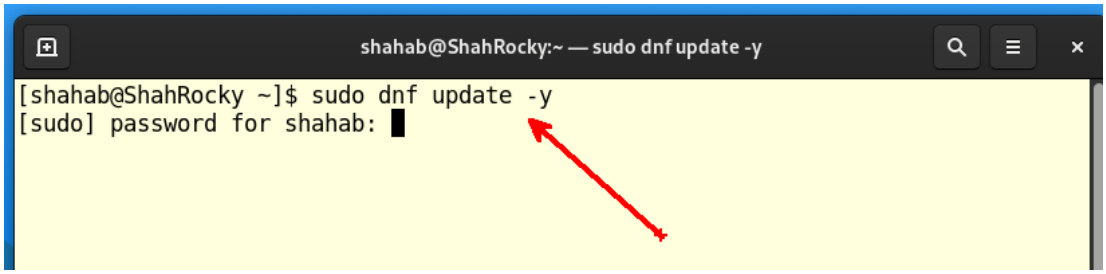
First, we check the operating system we have installed: **cat /etc/os-release**



Also can use **uname -a**



Then we update the system using our **dnf** repository

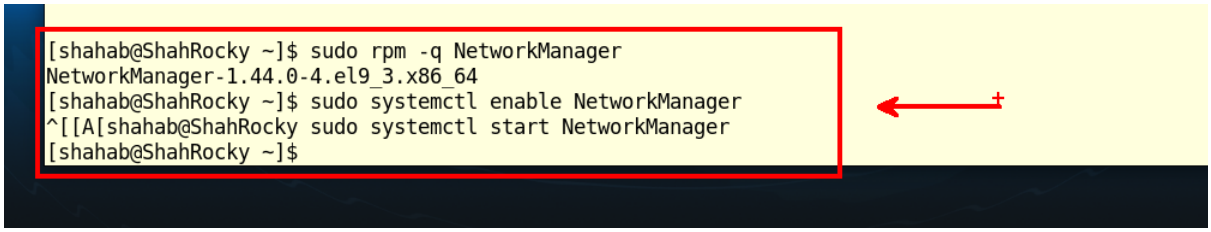


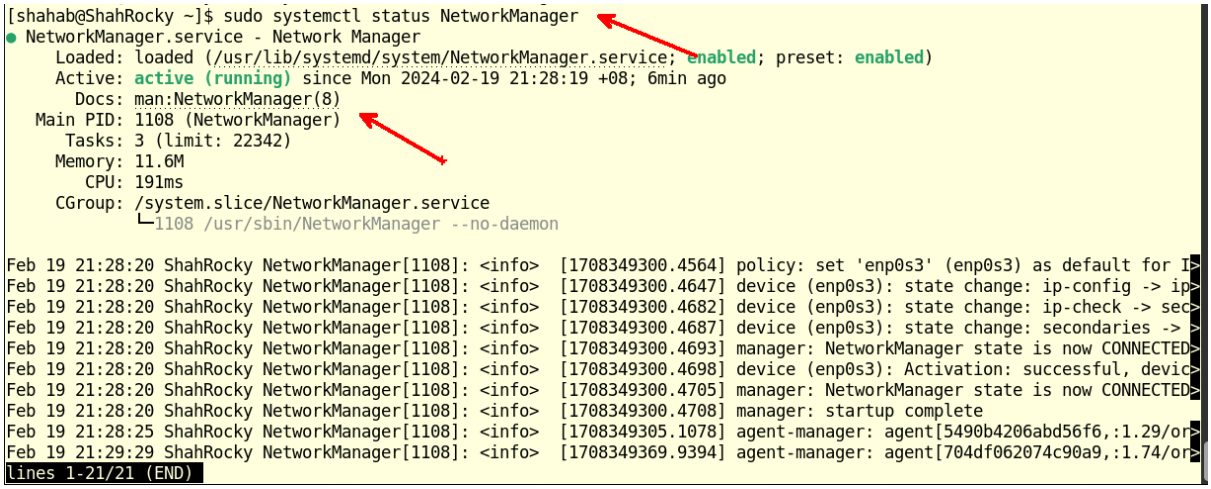
Then we **reboot**

We will use one common method which is using **NetworkManager**.

Check if it is installed, most of the time is installed by default in the RockyLinux

- **sudo rpm -q NetworkManager**
- **sudo systemctl enable NetworkManager**
- **sudo systemctl start NetworkManager**
- **sudo systemctl status NetworkManager**

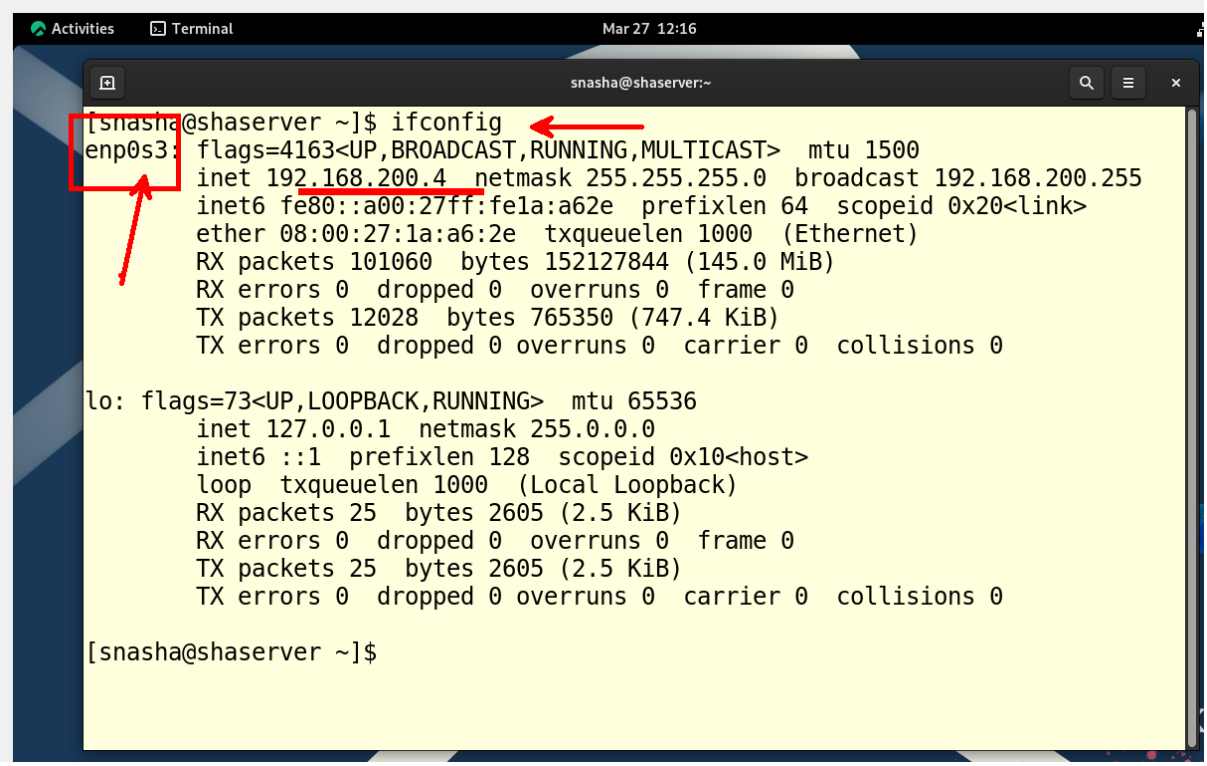




Next, we can use CLI to execute commands related to the network interface.

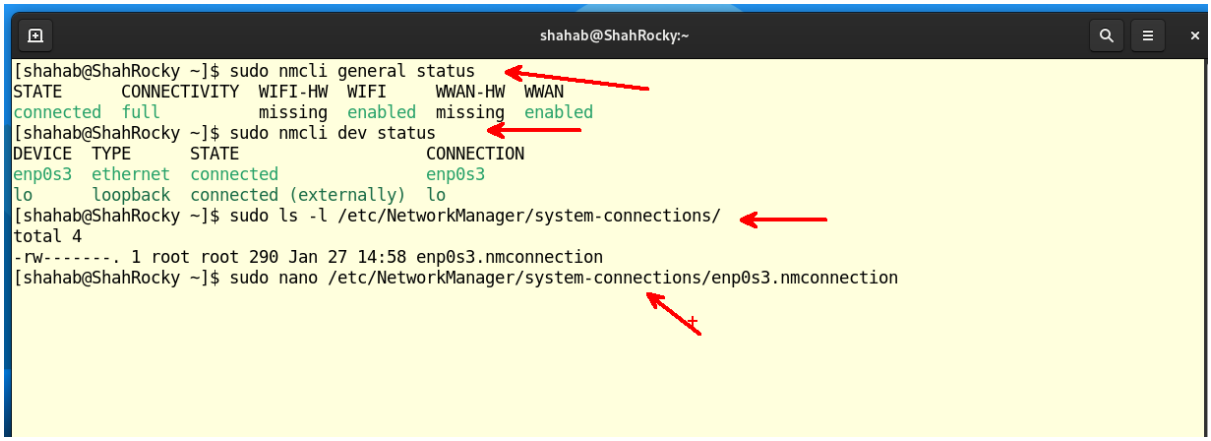
To check the current network address configuration, we use the command, make a note of your Rocky IP address (192.168.200.4) and your network interface name (enp0s3)

- **`ifconfig`**



To check for other information about the network

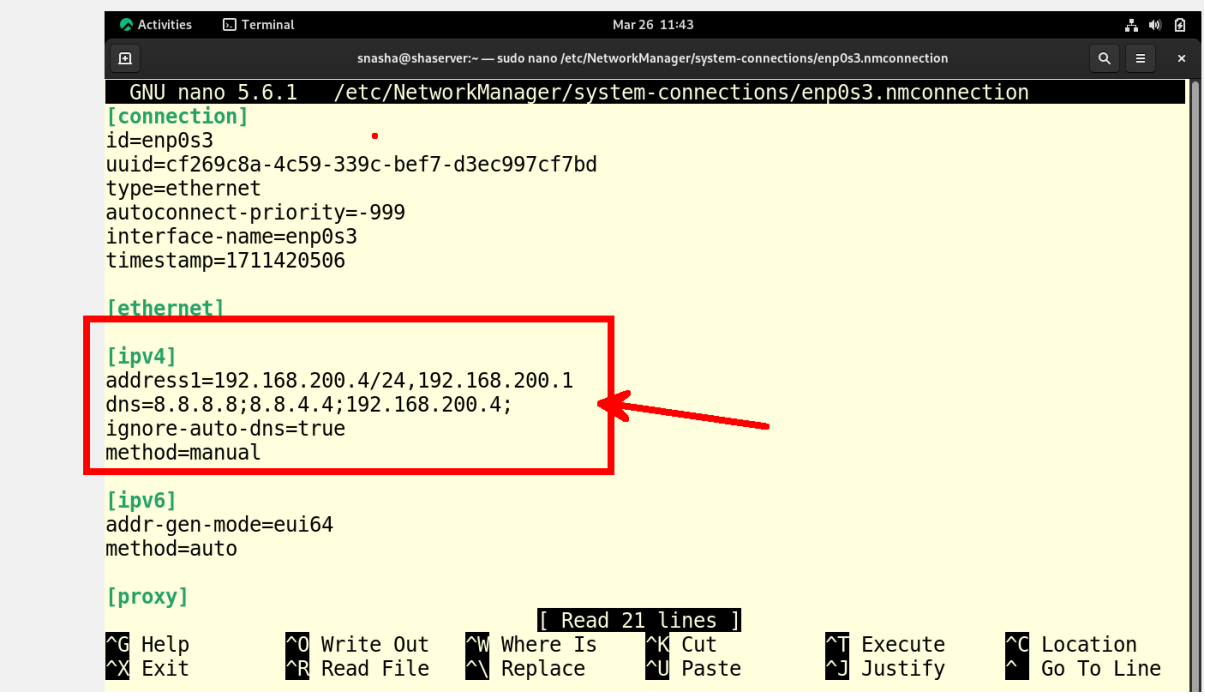
- **`sudo nmcli general status`**
- **`sudo nmcli device status`**
- **`sudo ls -l /etc/NetworkManager/system-connections`**
- **`sudo nano /etc/NetworkManager/system-connections/enp0s3.nmconnection`**



We can add the IP address of our DNS server as one example (google DNS IP address)
Note: address1=192.168.200.4/24

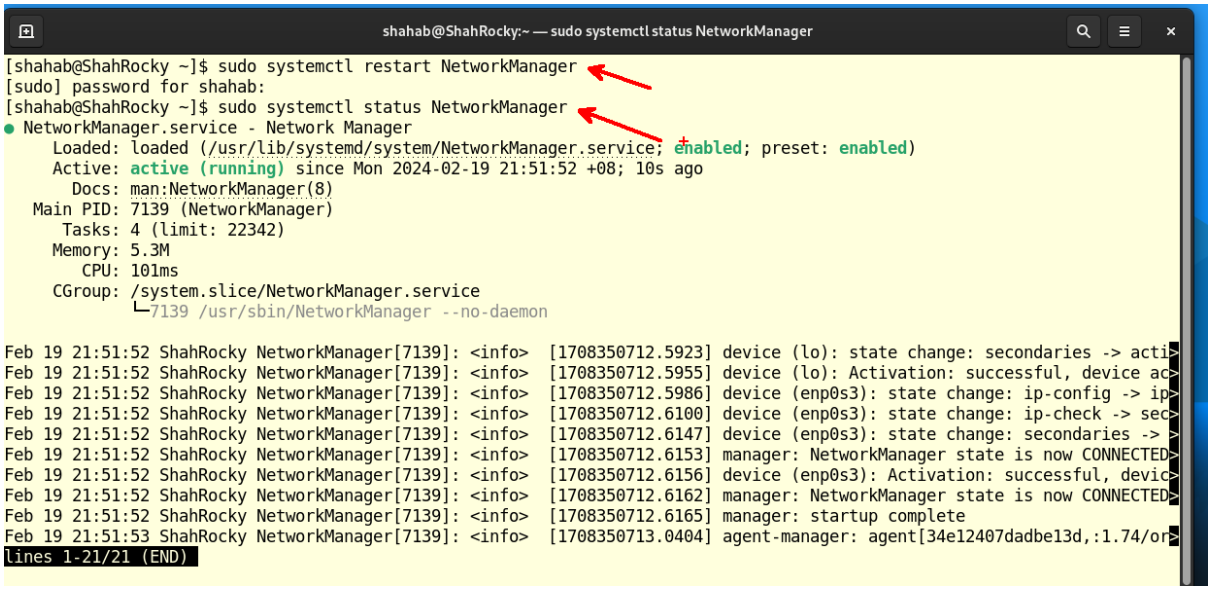
Our Default router is 192.168.200.1

Our server also will act as a DNS server (next lab session we will set the DNS at the Rocky server)



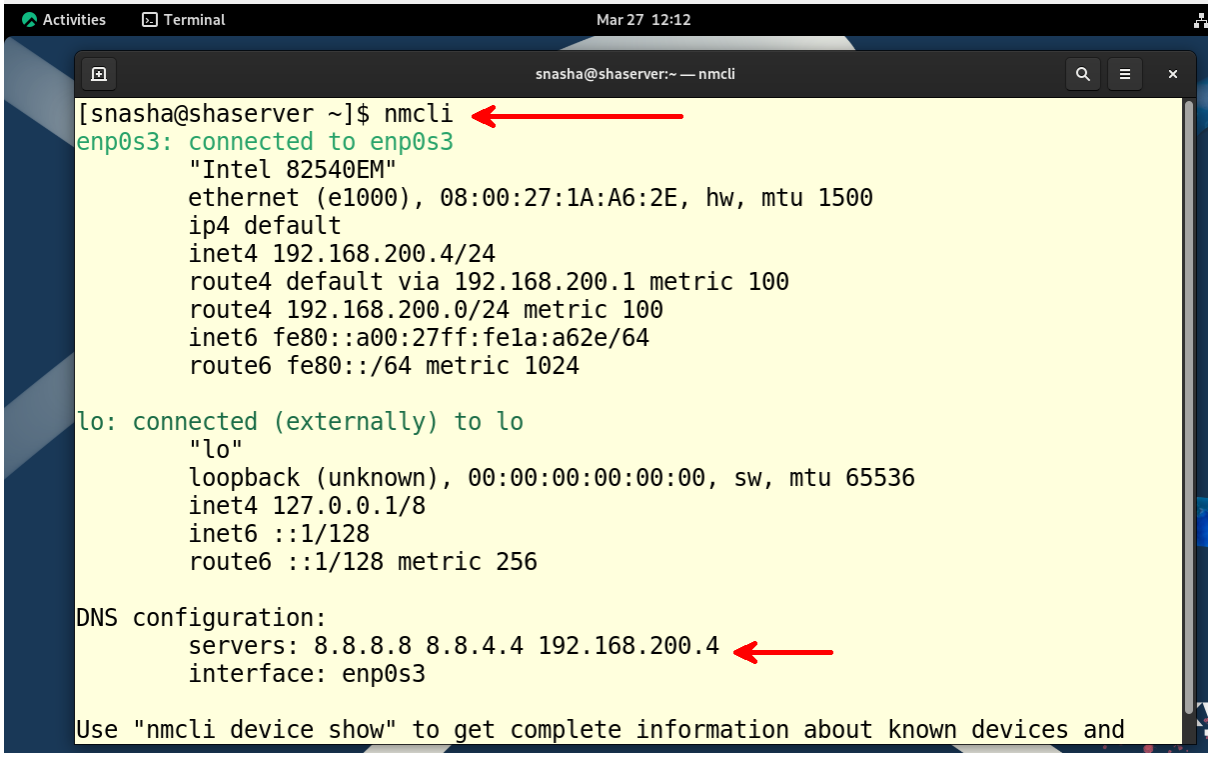
Then we restart

- **`sudo systemctl restart NetworkManager`**
- **`sudo systemctl status NetworkManager`**



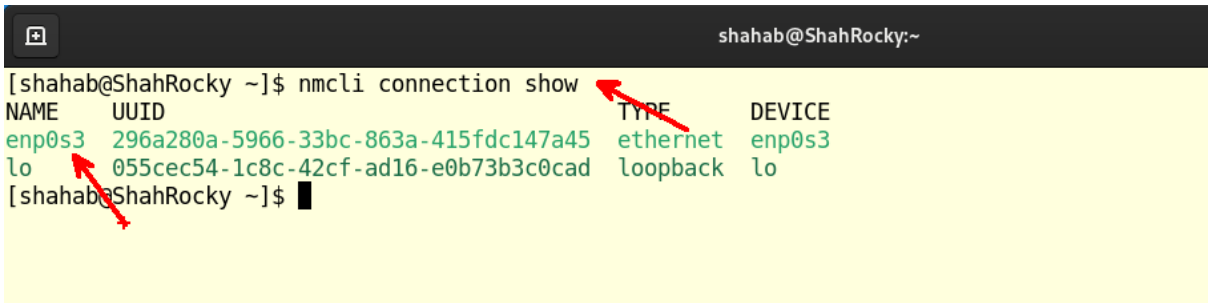
Then we run

- **nmcli (to see changes taking effect)**



We can do with the CLI command as well, we need to see first the name of our connection

- **nmcli connection show**
- **nmcli connection show enp0s3 (in this case enp0s3)**




shahab@ShahRocky:~ — nmcli connection show enp0s3
[shahab@ShahRocky ~]$ nmcli connection show
NAME UUID TYPE DEVICE
enp0s3 296a280a-5966-33bc-863a-415fdc147a45 ethernet enp0s3
lo 055cec54-1c8c-42cf-ad16-e0b73b3c0cad loopback lo
[shahab@ShahRocky ~]$ nmcli connection show enp0s3
connection.id: enp0s3
connection.uuid: 296a280a-5966-33bc-863a-415fdc147a45
connection.stable-id: --
connection.type: 802-3-ethernet
connection.interface-name: enp0s3
connection.autoconnect: yes
connection.autoconnect-priority: -999
connection.autoconnect-retries: -1 (default)
connection.multi-connect: 0 (default)
connection.auth-retries: -1
connection.timestamp: 1708350712
connection.permissions: --
connection.zone: --
connection.master: --
connection.slave-type: --
connection.autoconnect-slaves: -1 (default)
connection.secondaries: --
connection.gateway-ping-timeout: 0
connection.metered: unknown
connection.lldp: default

Ubuntu Client: Network Interface Config

Static IP configuration for our interface using nmcli (the same as Rocky Linux)

We will use one common method which is using NetworkManager.

The common method to set a static IP address on Ubuntu 22.04 LTS is by using the **nmcli tool.**

- **`su -   (or use sudo -i)`**
- **`apt-get update &&  apt-get upgrade`**

In case you can't log in as a root user continue using the sudo command:

- **`sudo apt-get update && apt-get upgrade`**

If you can't log in as a root user then we must use sudo

- **`sudo apt install network-manager`**

however, is installed most of the time by default and we can the following commands:

- **`sudo systemctl enable NetworkManager`**
- **`sudo systemctl start NetworkManager`**
- **`sudo systemctl status NetworkManager`**

```
shah@shahUbu:~/Desktop$ sudo systemctl status NetworkManager
● NetworkManager.service - Network Manager
     Loaded: loaded (/lib/systemd/system/NetworkManager.service; enabled; vendo>
     Active: active (running) since Mon 2024-02-19 21:13:37 +08; 1 day 20h ago
       Docs: man:NetworkManager(8)
   Main PID: 611 (NetworkManager)
      Tasks: 3 (limit: 4598)
     Memory: 10.8M
        CPU: 501ms
     CGroup: /system.slice/NetworkManager.service
             └─611 /usr/sbin/NetworkManager --no-daemon

Feb 19 21:13:42 shahUbu NetworkManager[611]: <info>  [1708348422.7839] manager:>
Feb 19 21:13:42 shahUbu NetworkManager[611]: <info>  [1708348422.7843] manager:>
Feb 19 21:13:42 shahUbu NetworkManager[611]: <info>  [1708348422.7844] policy:  >
Feb 19 21:13:42 shahUbu NetworkManager[611]: <info>  [1708348422.7852] device (>
Feb 19 21:13:42 shahUbu NetworkManager[611]: <info>  [1708348422.7896] manager:>
Feb 19 21:13:44 shahUbu NetworkManager[611]: <info>  [1708348424.1233] manager:>
Feb 19 21:13:45 shahUbu NetworkManager[611]: <info>  [1708348425.3060] modem-ma>
Feb 19 21:13:45 shahUbu NetworkManager[611]: <info>  [1708348425.3531] modem-ma>
Feb 21 17:29:37 shahUbu NetworkManager[611]: <info>  [1708507777.0486] agent-ma>
Feb 21 17:30:02 shahUbu NetworkManager[611]: <info>  [1708507802.0172] agent-ma>
lines 1-21/21 (END)
```

To check the current configuration, we use the command

- **ifconfig**

```
snasha@shaclient: ~/Desktop
snasha@shaclient:~/Desktop$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.200.5  netmask 255.255.255.0  broadcast 192.168.200.255
        inet6 fe80::16bb:6fb0:aa1a:1498  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:51:40:35  txqueuelen 1000  (Ethernet)
        RX packets 39  bytes 9852 (9.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 131  bytes 14487 (14.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 214  bytes 16844 (16.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 214  bytes 16844 (16.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

snasha@shaclient:~/Desktop$
```

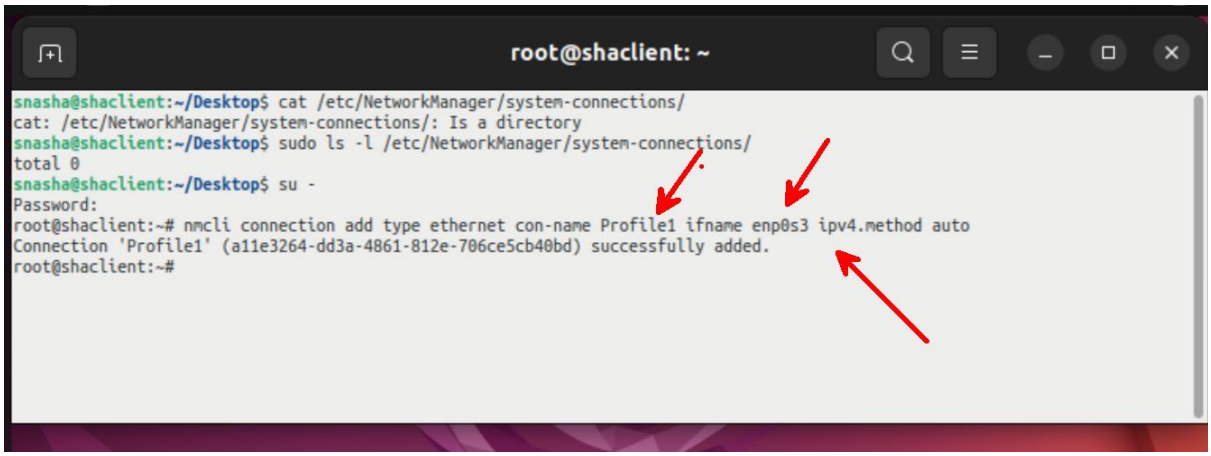
To check for other information about the network

- **sudo nmcli general status**
- **sudo nmcli dev status**
- **sudo ls -l /etc/NetworkManager/system-connections**

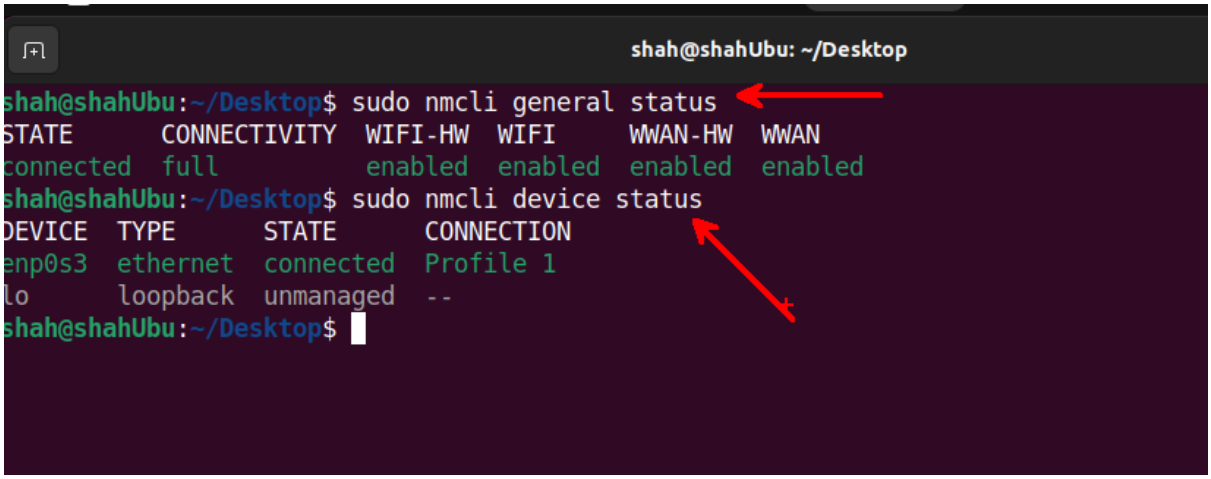
IMPORTANT STEP TO DO:

The nmcli configuration profiles are located in */etc/NetworkManager/system-connections/*. Since that folder is empty by default after install, we will need to create a connection profile file called "*Profile1*" and assign it to the device enp0s3. Since there is only one network card, we will have to switch connection profiles.

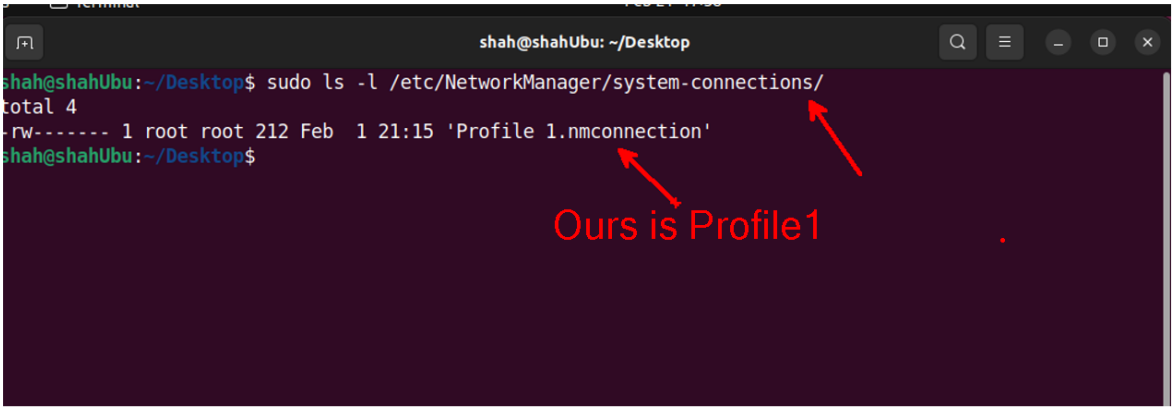
- **Nmcli connection add type ethernet con-name Profilel ifname enp0s3 ipv4.method auto**



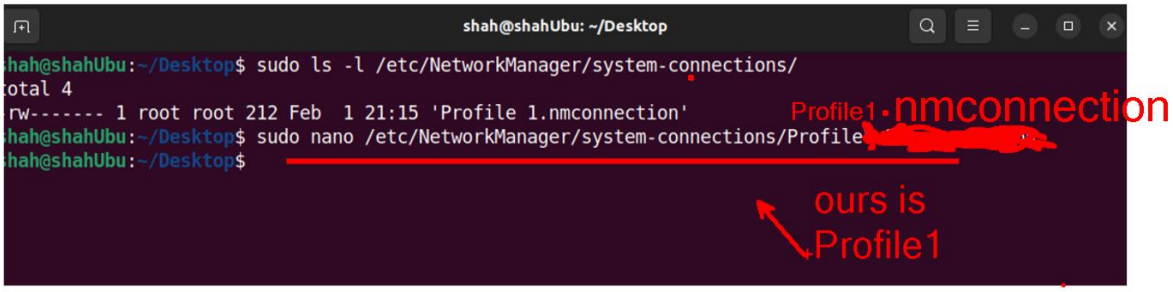
- **sudo nmcli general status**



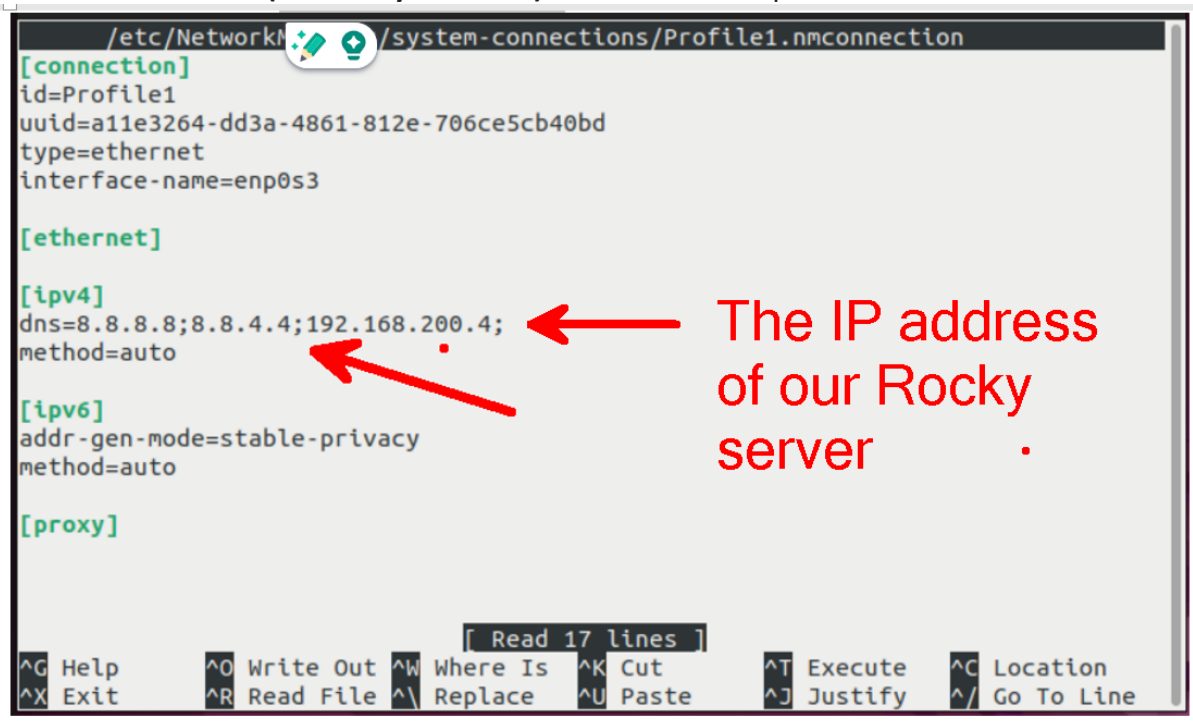
- **sudo ls -l /etc/NetworkManager/system-connections/**



- **sudo nano /etc/NetworkManager/system-connections/Profile1.nmconnection**



We can add the DNS **(our Rocky Server IP)** in here for example

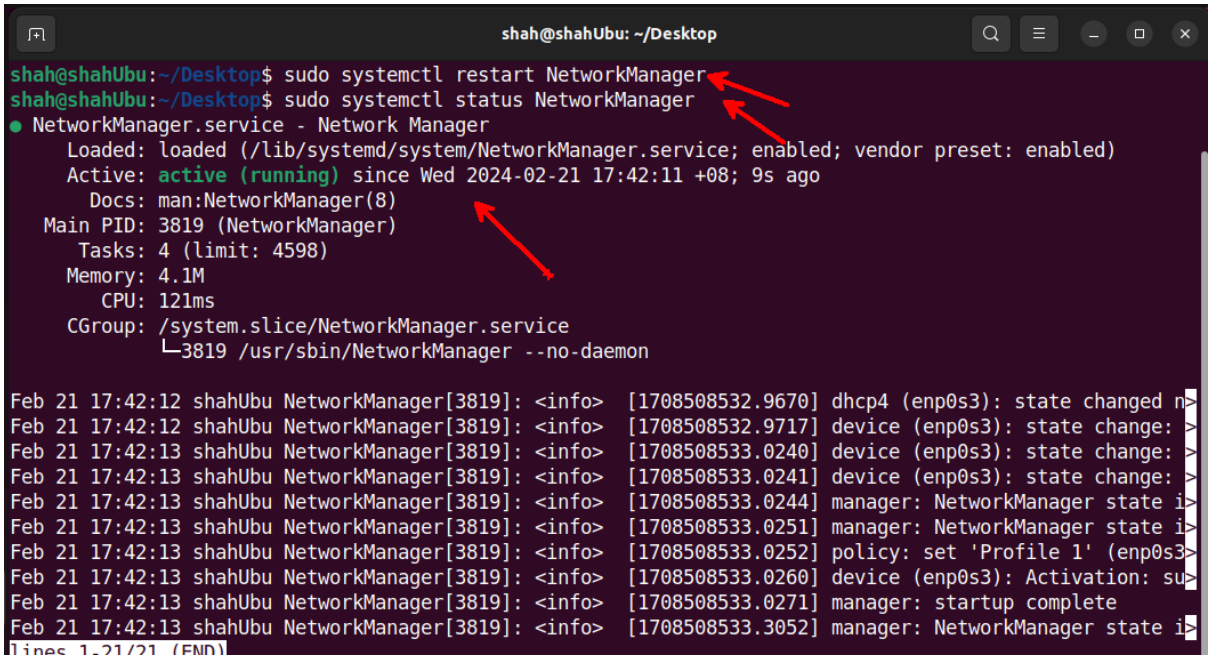


Exit root

- **`exit`**

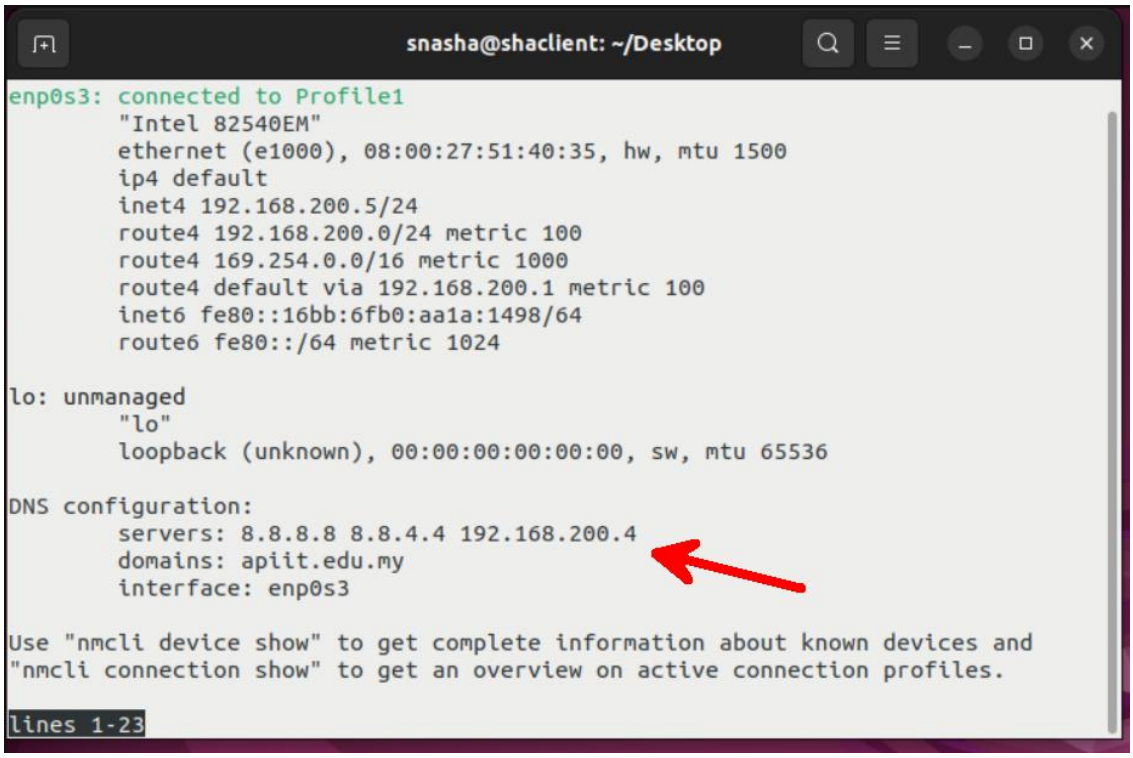
Then we restart

- **`sudo systemctl restart NetworkManager`**
- **`sudo systemctl status NetworkManager`**
- **`reboot`**



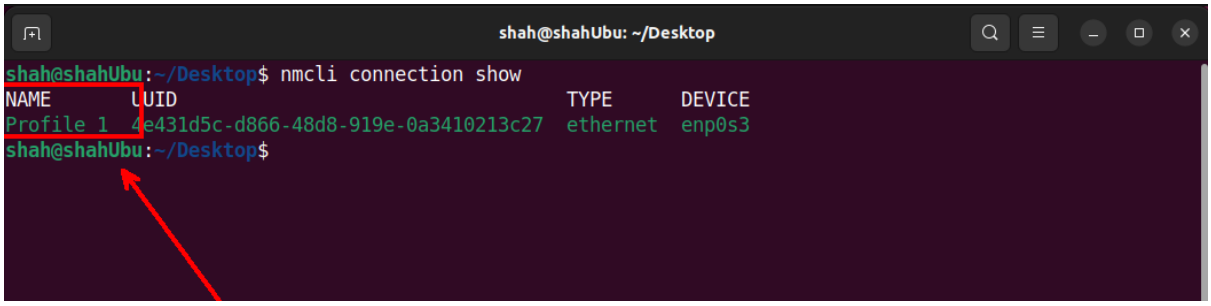
Then we run

- **nmcli (to see changes taking effect)**



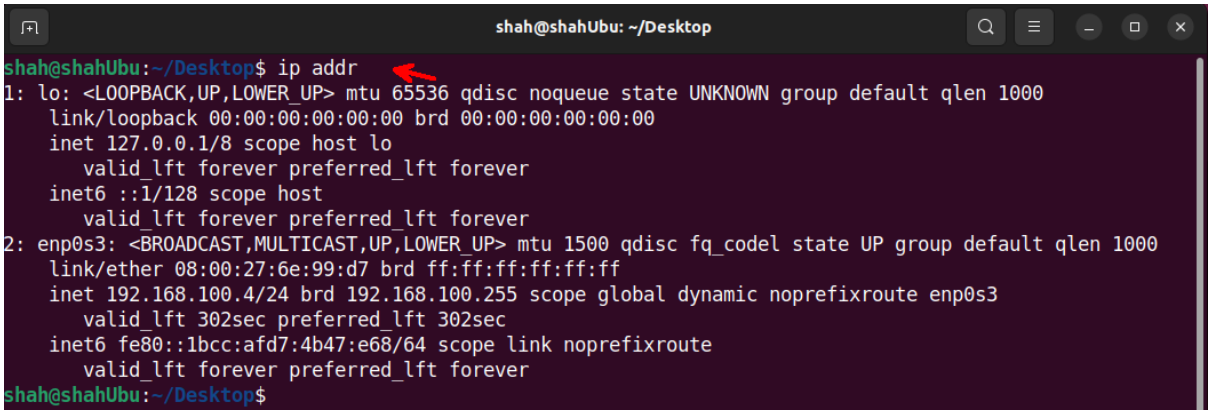
We can do this with the CLI command as well, we need to see first the name of our connection

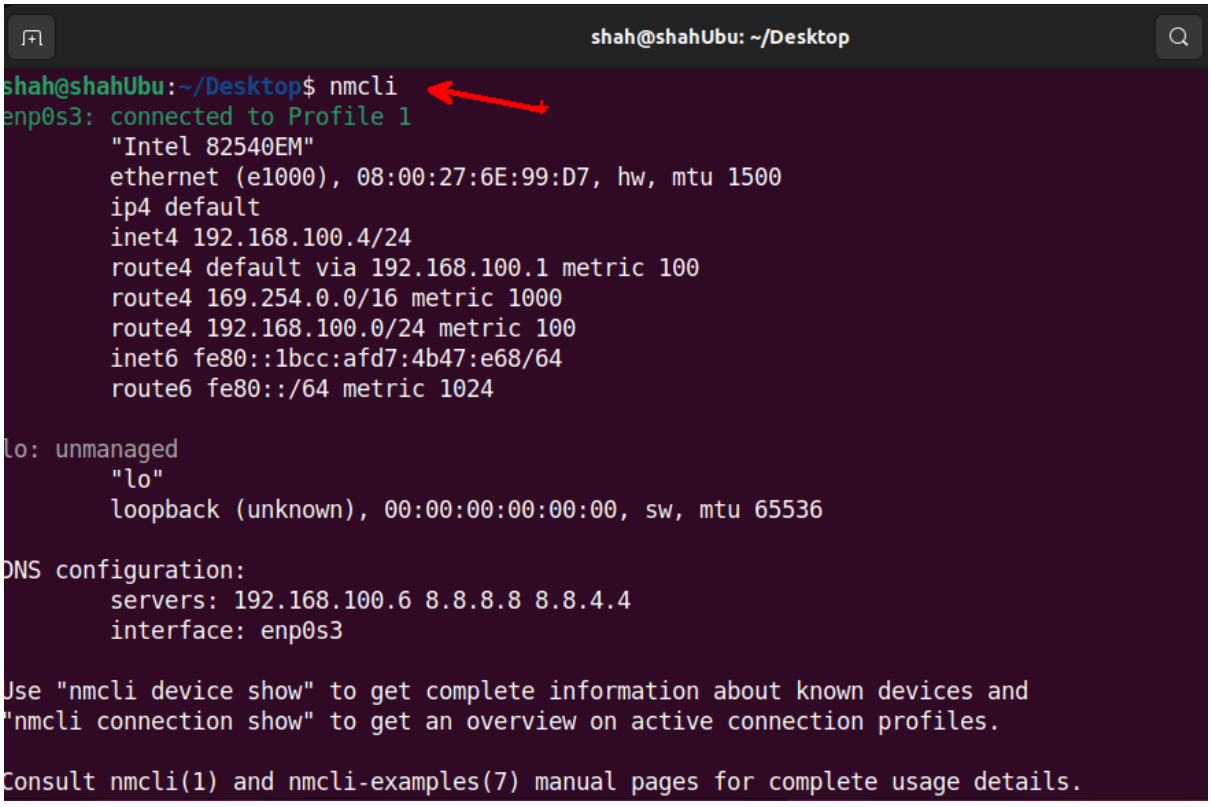
- **nmcli connection show**
- **nmcli connection show Profile (in this case Profile1)**

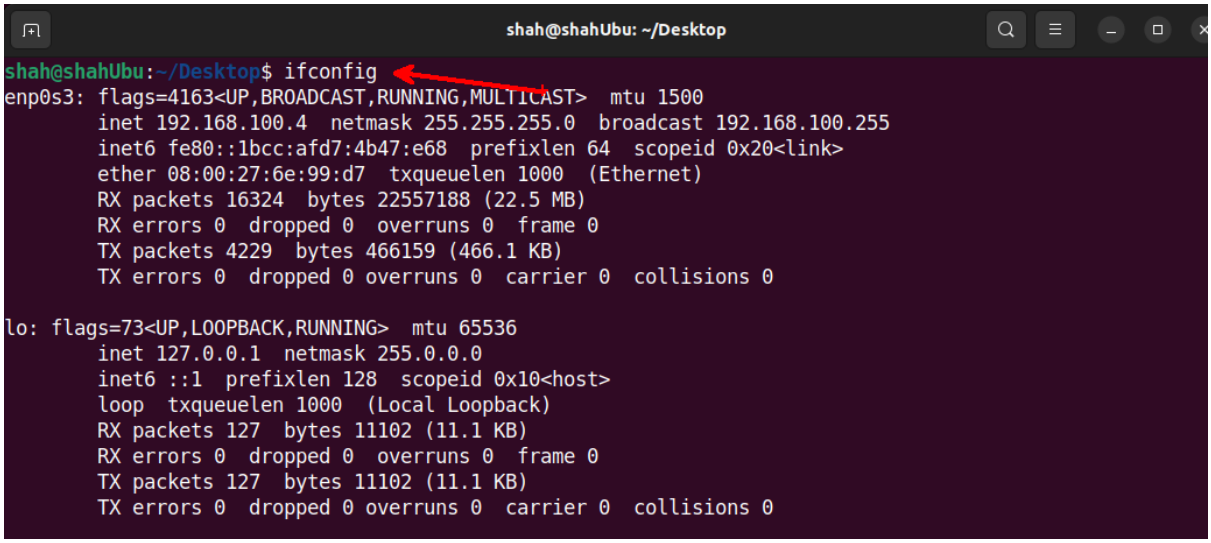


Alternative commands to check the IP

- **Nmcli**
- **Ip addr**
- **Ifconfig**






shah@shahUbu: ~/Desktop
shah@shahUbu:~/Desktop$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 192.168.100.4 netmask 255.255.255.0 broadcast 192.168.100.255
inet6 fe80::1bcc:afd7:4b47:e68 prefixlen 64 scopeid 0x20<link>
ether 08:00:27:6e:99:d7 txqueuelen 1000 (Ethernet)
RX packets 16324 bytes 22557188 (22.5 MB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 4229 bytes 466159 (466.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 127 bytes 11102 (11.1 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 127 bytes 11102 (11.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0