

Project Name: **Cybersecurity Club's Raspberry Pi Networking Project**

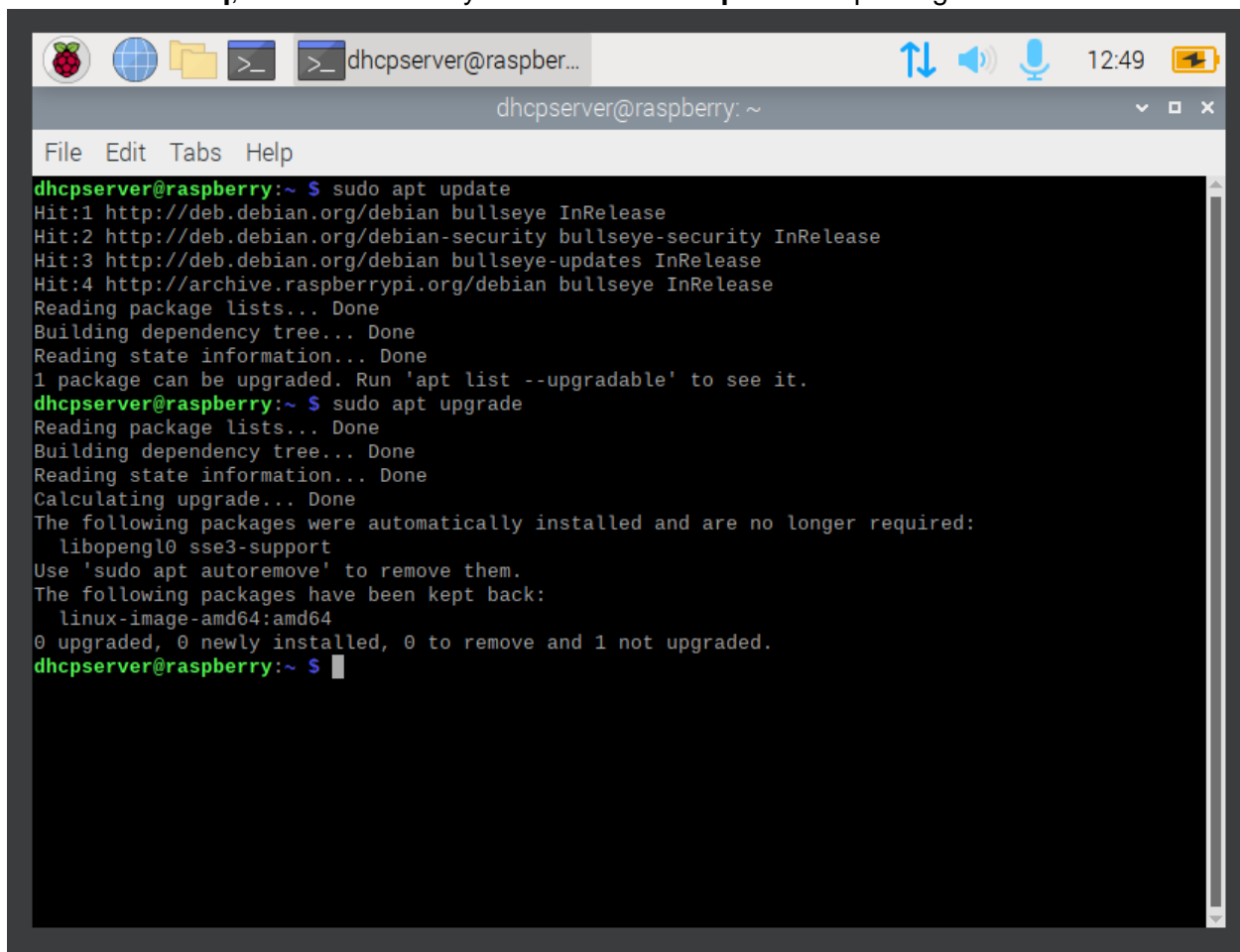
Date: **July 11, 2023**

Created by: **Jason Patrick Salerno**

Purpose: **Documentation for the DHCP Server**

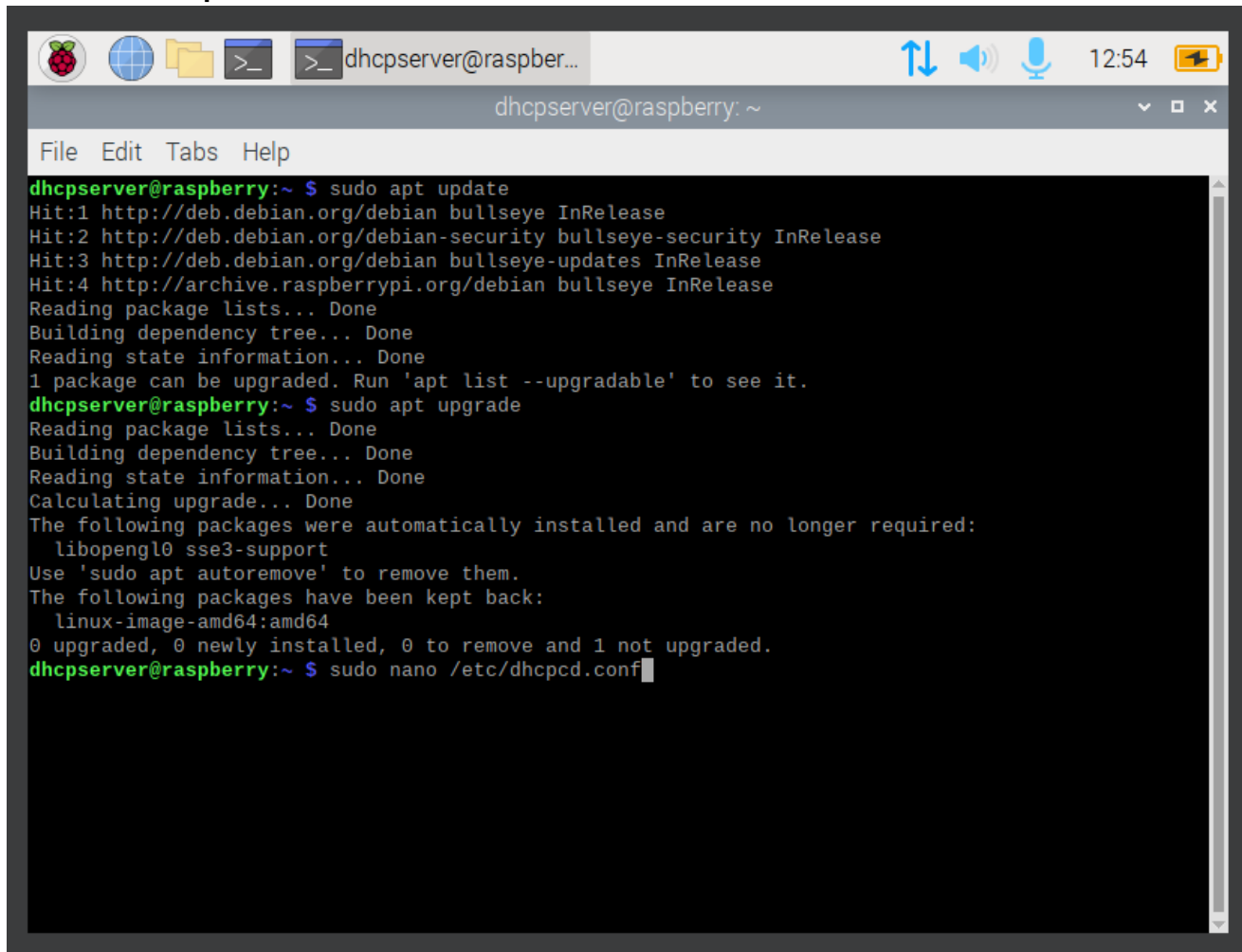
## Set up a Raspberry Pi as a DHCP Server

1. The first command I run is **sudo apt update**, this command updates a database of readily available software packages and their versions known as the local package index. The second command I run is **sudo apt upgrade**, this command upgrades the software packages on a Linux system. **Note:** I've forgot to add **sudo apt install dnsmasq**, but I have already installed **dnsmasq** software packages on this **RPI OS**.

A screenshot of a terminal window on a Raspberry Pi. The window title is 'dhcpcserver@raspberr...'. The terminal shows the following output:

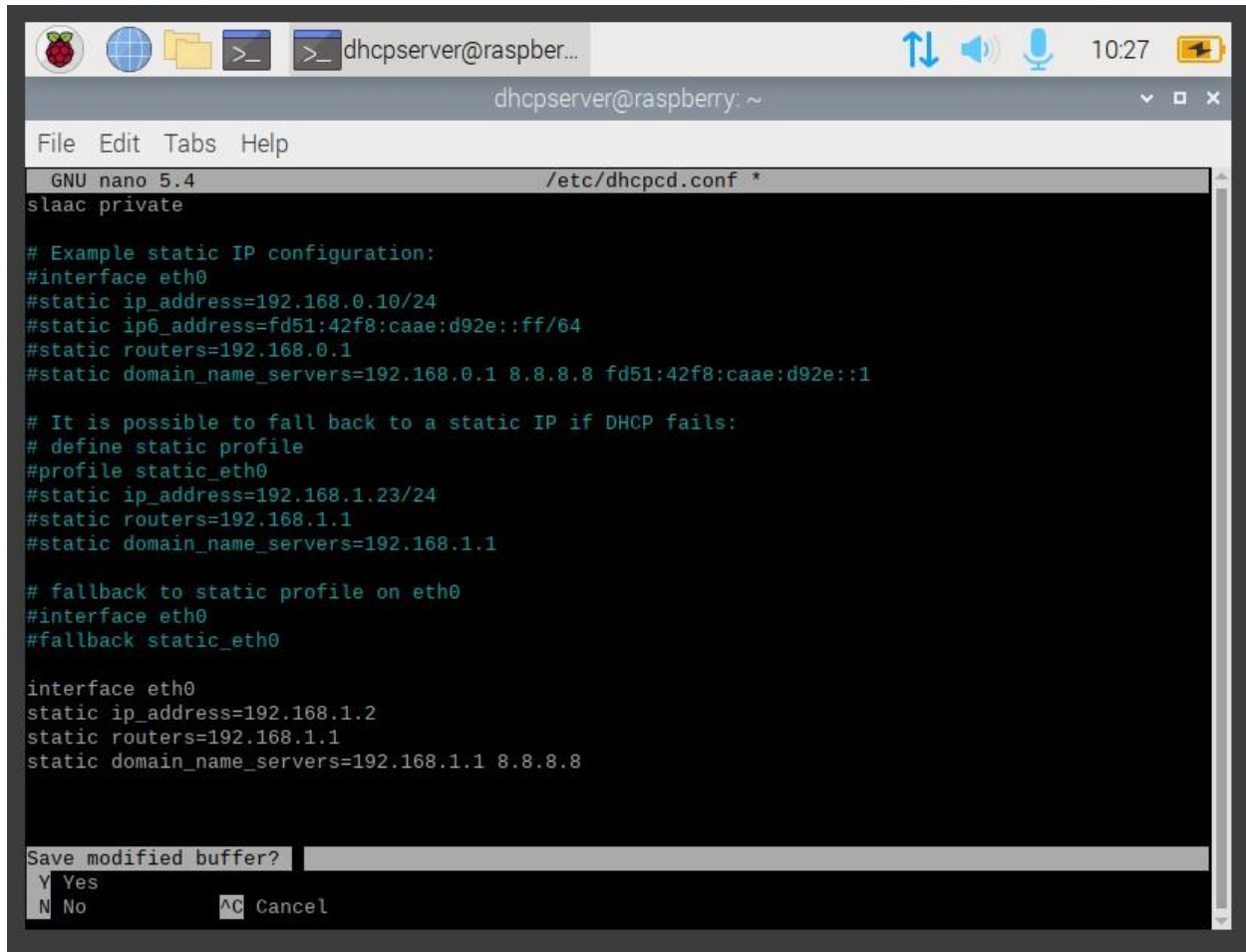
```
dhcpcserver@raspberr~ $ sudo apt update
Hit:1 http://deb.debian.org/debian bullseye InRelease
Hit:2 http://deb.debian.org/debian-security bullseye-security InRelease
Hit:3 http://deb.debian.org/debian bullseye-updates InRelease
Hit:4 http://archive.raspberrypi.org/debian bullseye InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
1 package can be upgraded. Run 'apt list --upgradable' to see it.
dhcpcserver@raspberr~ $ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libopengl0 sse3-support
Use 'sudo apt autoremove' to remove them.
The following packages have been kept back:
  linux-image-amd64:amd64
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
dhcpcserver@raspberr~ $
```

2. The next step is to open the `dhcpcd.conf` file, using the command: **sudo nano /etc/dhcpcd.conf** and add a few lines to it.



```
dhcpcserver@raspberry: ~  
File Edit Tabs Help  
dhcpcserver@raspberry:~ $ sudo apt update  
Hit:1 http://deb.debian.org/debian bullseye InRelease  
Hit:2 http://deb.debian.org/debian-security bullseye-security InRelease  
Hit:3 http://deb.debian.org/debian bullseye-updates InRelease  
Hit:4 http://archive.raspberrypi.org/debian bullseye InRelease  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
1 package can be upgraded. Run 'apt list --upgradable' to see it.  
dhcpcserver@raspberry:~ $ sudo apt upgrade  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
Calculating upgrade... Done  
The following packages were automatically installed and are no longer required:  
  libopengl0 sse3-support  
Use 'sudo apt autoremove' to remove them.  
The following packages have been kept back:  
  linux-image-amd64:amd64  
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.  
dhcpcserver@raspberry:~ $ sudo nano /etc/dhcpcd.conf
```

3. In the configuration file, I Added: **interface eth0,static ip\_address=192.168.1.2, static routers=192.168.1.1, static domain\_name\_servers=192.168.1.1 8.8.8.8**. After uncommenting those lines, to save the changes I pressed the keys **CTRL + X** and entered to exit the configuration file.



```
dhcpserver@raspber... 10:27
dhcpserver@raspberrypi: ~
File Edit Tabs Help
GNU nano 5.4 /etc/dhcpd.conf *
slaac private

# Example static IP configuration:
#interface eth0
#static ip_address=192.168.0.10/24
#static ip6_address=fd51:42f8:caae:d92e::ff/64
#static routers=192.168.0.1
#static domain_name_servers=192.168.0.1 8.8.8.8 fd51:42f8:caae:d92e::1

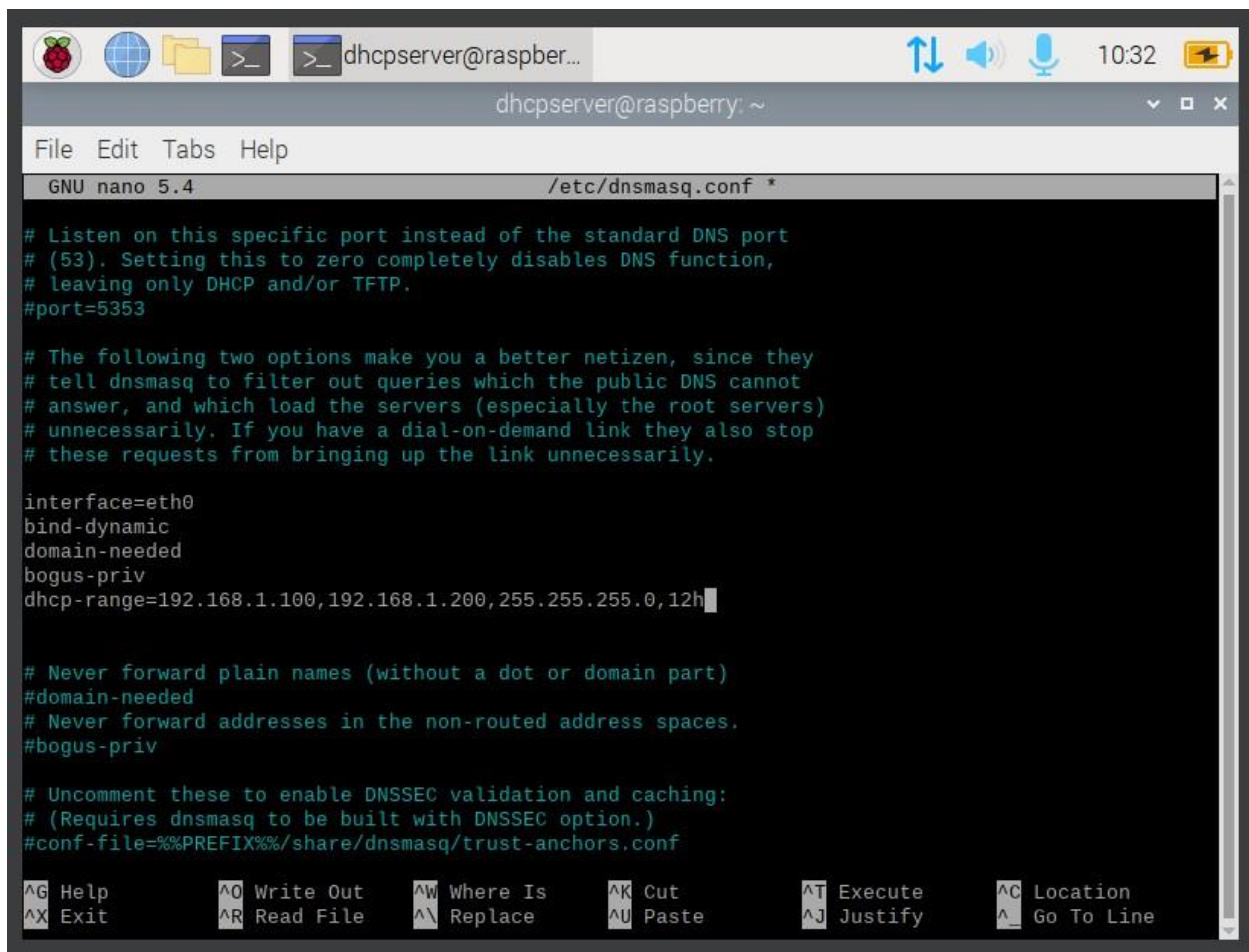
# It is possible to fall back to a static IP if DHCP fails:
# define static profile
#profile static_eth0
#static ip_address=192.168.1.23/24
#static routers=192.168.1.1
#static domain_name_servers=192.168.1.1

# fallback to static profile on eth0
#interface eth0
#fallback static_eth0

interface eth0
static ip_address=192.168.1.2
static routers=192.168.1.1
static domain_name_servers=192.168.1.1 8.8.8.8

Save modified buffer?
Y Yes
N No      ^C Cancel
```

4. Next I open the configuration file using the command: **sudo nano /etc/dnsmasq.conf** and I added a few lines to our configuration file which are: **interface=eth0, bind-dynamic, domain-needed, bogus-priv, dhcp-range=192.168.1.100, 192.168.1.200, 255.255.255.0, 12h**. Here is a short explanation of each line **interface=eth0**: Indicates that the Ethernet interface with the identifier eth0 will use the following configurations. **bind-dynamic**: Tells the DHCP server's IP address-assigning daemon to dynamically bind to it. **domain-needed**: Ensures that only fully qualified domain names are used while performing DNS domain resolution. To avoid the usage of possibly false or malicious IP addresses, **bogus-priv** enables verification for reserved private IP address ranges in DNS answers.



```
dhcpcserver@raspber...
dhcpcserver@raspberrypi: ~
File Edit Tabs Help
GNU nano 5.4 /etc/dnsmasq.conf *
# Listen on this specific port instead of the standard DNS port
# (53). Setting this to zero completely disables DNS function,
# leaving only DHCP and/or TFTP.
#port=5353

# The following two options make you a better netizen, since they
# tell dnsmasq to filter out queries which the public DNS cannot
# answer, and which load the servers (especially the root servers)
# unnecessarily. If you have a dial-on-demand link they also stop
# these requests from bringing up the link unnecessarily.

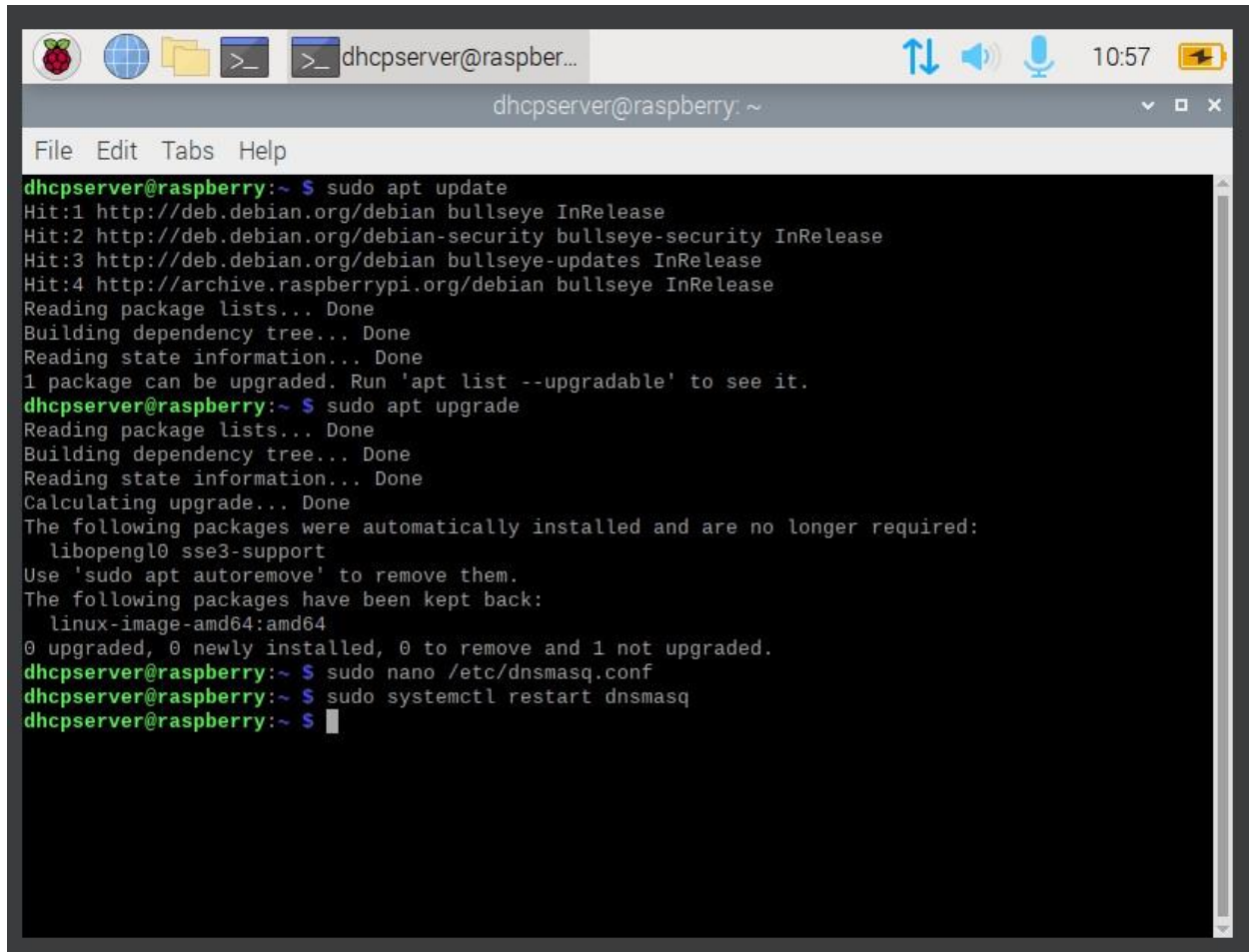
interface=eth0
bind-dynamic
domain-needed
bogus-priv
dhcp-range=192.168.1.100,192.168.1.200,255.255.255.0,12h

# Never forward plain names (without a dot or domain part)
#domain-needed
# Never forward addresses in the non-routed address spaces.
#bogus-priv

# Uncomment these to enable DNSSEC validation and caching:
# (Requires dnsmasq to be built with DNSSEC option.)
#conf-file=%PREFIX%/share/dnsmasq/trust-anchors.conf

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

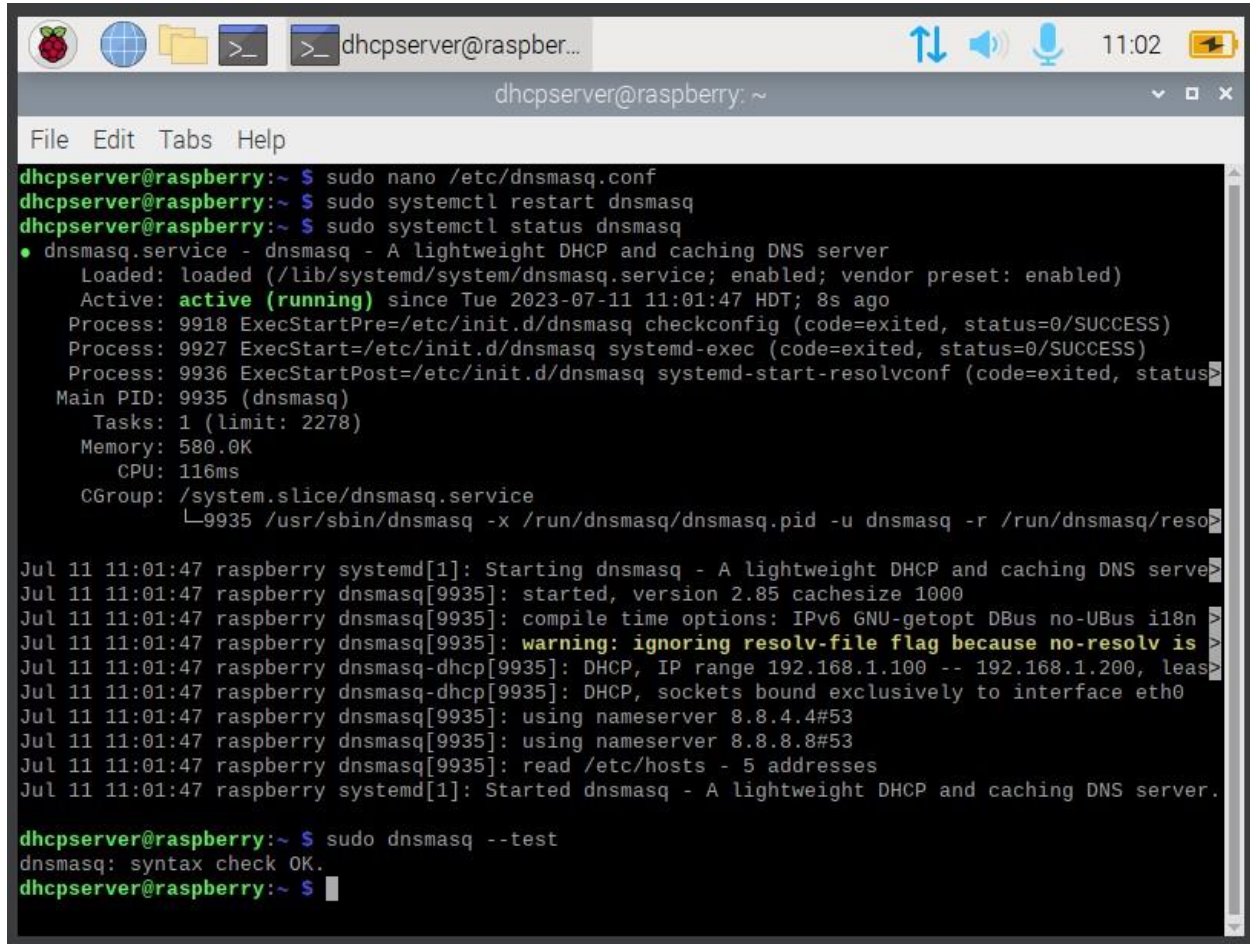
5. After **saving the changes** in the configuration file, the next thing to do is **apply these changes**, and to do that we need to restart the dnsmasq service, using the command **sudo systemctl restart dnsmasq**



```
dhcpcserver@raspberr... 10:57
dhcpcserver@raspberry: ~
File Edit Tabs Help
dhcpcserver@raspberry:~ $ sudo apt update
Hit:1 http://deb.debian.org/debian bullseye InRelease
Hit:2 http://deb.debian.org/debian-security bullseye-security InRelease
Hit:3 http://deb.debian.org/debian bullseye-updates InRelease
Hit:4 http://archive.raspberrypi.org/debian bullseye InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
1 package can be upgraded. Run 'apt list --upgradable' to see it.
dhcpcserver@raspberry:~ $ sudo apt upgrade
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libopengl0 sse3-support
Use 'sudo apt autoremove' to remove them.
The following packages have been kept back:
  linux-image-amd64:amd64
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
dhcpcserver@raspberry:~ $ sudo nano /etc/dnsmasq.conf
dhcpcserver@raspberry:~ $ sudo systemctl restart dnsmasq
dhcpcserver@raspberry:~ $
```

- The next step to do is to restart the dnsmasq and to do that we need to run the command: **sudo systemctl restart dnsmasq**, after restarting the dnsmasq.service the next step is to verify its status if it is active and running, by executing the command: **sudo systemctl status dnsmasq**. To verify it again I also executed the command: **sudo dnsmasq --test**, as we can see it displays **dnsmasq: syntax check OK**.

**Note:** I've attempted to verify that our Raspberry Pi OS is running as a DHCP server. I will verify it **4 times**. In this step, I've already verified it **twice(2)**.



```
dhcpcserver@raspberr... 11:02
dhcpcserver@raspberr: ~
File Edit Tabs Help
dhcpcserver@raspberr:~ $ sudo nano /etc/dnsmasq.conf
dhcpcserver@raspberr:~ $ sudo systemctl restart dnsmasq
dhcpcserver@raspberr:~ $ sudo systemctl status dnsmasq
● dnsmasq.service - dnsmasq - A lightweight DHCP and caching DNS server
   Loaded: loaded (/lib/systemd/system/dnsmasq.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2023-07-11 11:01:47 HDT; 8s ago
     Process: 9918 ExecStartPre=/etc/init.d/dnsmasq checkconfig (code=exited, status=0/SUCCESS)
     Process: 9927 ExecStart=/etc/init.d/dnsmasq systemd-exec (code=exited, status=0/SUCCESS)
     Process: 9936 ExecStartPost=/etc/init.d/dnsmasq systemd-start-resolvconf (code=exited, status=0/SUCCESS)
    Main PID: 9935 (dnsmasq)
       Tasks: 1 (limit: 2278)
      Memory: 580.0K
         CPU: 116ms
      CGroup: /system.slice/dnsmasq.service
              └─9935 /usr/sbin/dnsmasq -x /run/dnsmasq/dnsmasq.pid -u dnsmasq -r /run/dnsmasq/resolvconf

Jul 11 11:01:47 raspberry systemd[1]: Starting dnsmasq - A lightweight DHCP and caching DNS server.
Jul 11 11:01:47 raspberry dnsmasq[9935]: started, version 2.85 cachesize 1000
Jul 11 11:01:47 raspberry dnsmasq[9935]: compile time options: IPv6 GNU-getopt DBus no-UBus i18n
Jul 11 11:01:47 raspberry dnsmasq[9935]: warning: ignoring resolv-file flag because no-resolv is
Jul 11 11:01:47 raspberry dnsmasq-dhcp[9935]: DHCP, IP range 192.168.1.100 -- 192.168.1.200, lease
Jul 11 11:01:47 raspberry dnsmasq-dhcp[9935]: DHCP, sockets bound exclusively to interface eth0
Jul 11 11:01:47 raspberry dnsmasq[9935]: using nameserver 8.8.4.4#53
Jul 11 11:01:47 raspberry dnsmasq[9935]: using nameserver 8.8.8.8#53
Jul 11 11:01:47 raspberry dnsmasq[9935]: read /etc/hosts - 5 addresses
Jul 11 11:01:47 raspberry systemd[1]: Started dnsmasq - A lightweight DHCP and caching DNS server.

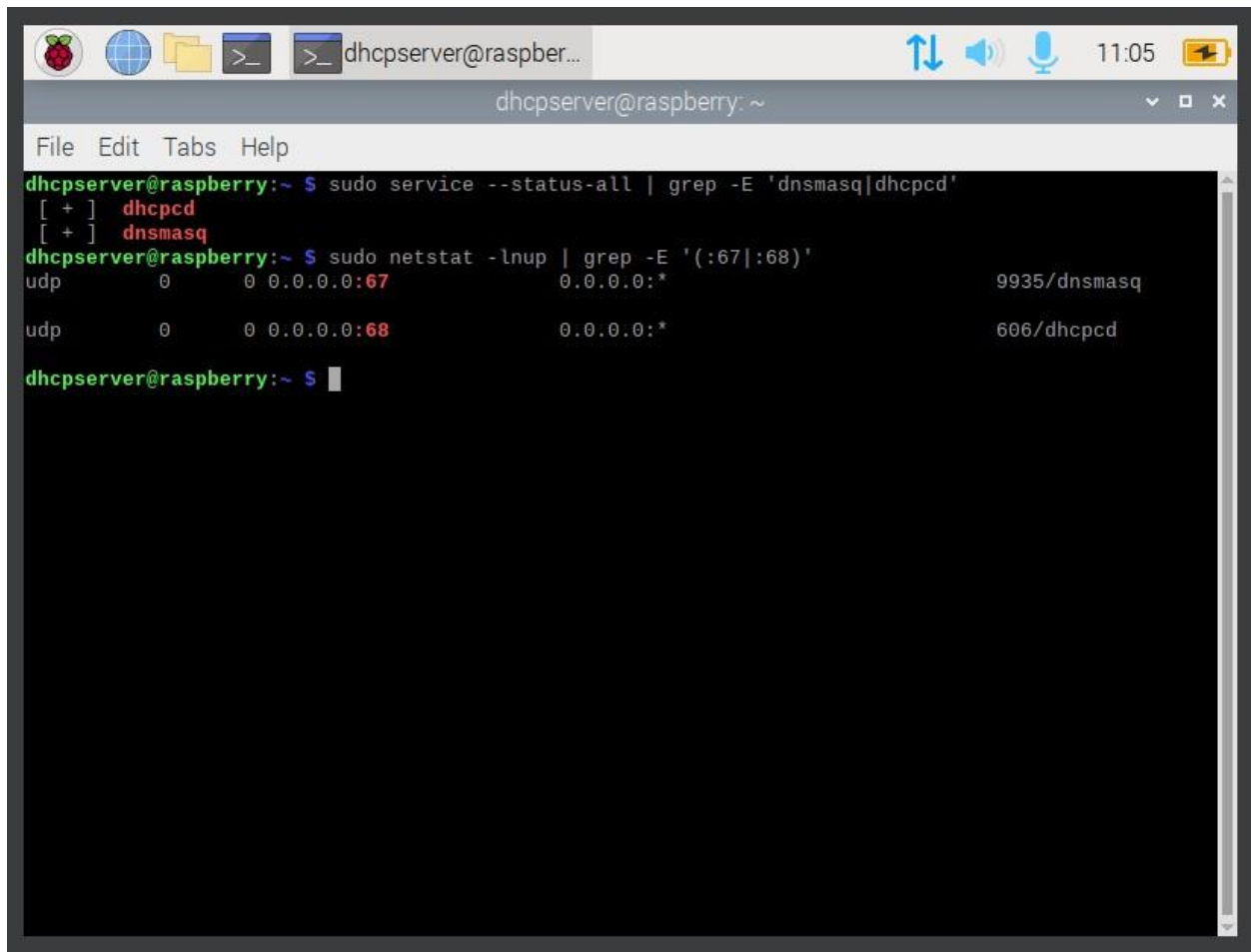
dhcpcserver@raspberr:~ $ sudo dnsmasq --test
dnsmasq: syntax check OK.
dhcpcserver@raspberr:~ $
```

- In this step, I will verify the Raspberry Pi OS is running as a DHCP server, **2 more times**. By executing the command: **sudo service --status-all | grep -E**



**'dnsmasq|dhcpcd'** this command will display all the processes/services that are currently running/On however I used a pipe grep to find the dnsmasq & dhcpcd services and display only those services. The next command I executed is: **sudo netstat -lnup | grep -E '(:67|:68)'** this command will list the listening UDP ports on our raspberry pi OS, which are ports 67 and 68. It filters and displays only the lines that contain either ":67" or ":68" in the output of the netstat command. Now we can say that our Raspberry Pi OS is now running as a DHCP server.

**Note:** On page 8 I have a command summary of all of this, in case you want to try and test this out for yourself.



```
dhcpcserver@raspberr... 11:05
dhcpcserver@raspberr: ~
File Edit Tabs Help
dhcpcserver@raspberr:~ $ sudo service --status-all | grep -E 'dnsmasq|dhcpcd'
[ + ] dhcpcd
[ + ] dnsmasq
dhcpcserver@raspberr:~ $ sudo netstat -lnup | grep -E '(:67|:68)'
udp      0      0 0.0.0.0:67          0.0.0.0:*          9935/dnsmasq
udp      0      0 0.0.0.0:68          0.0.0.0:*          606/dhcpcd
dhcpcserver@raspberr:~ $
```

8. Here is a summary or list of the commands to execute if you want to try this out:

- sudo apt update

- sudo apt upgrade

- sudo apt install dnsmasq

- sudo nano /etc/dhcpd.conf # 1st conf file

# when you have opened the DHCPD.conf configuration file, Add the following:

- interface eth0

- static ip\_address=192.168.1.2

- static routers=192.168.1.1

- static domain\_name\_servers=192.168.1.1 8.8.8.8

- sudo reboot

- sudo apt install dnsmasq

- sudo nano /etc/dnsmasq.conf # 2nd conf file

# When you have opened the DNSmasq.conf configuration file, Add the following:

- interface=eth0 # OR: 'wlan0' for Wi-Fi

- bind-dynamic

- domain-needed

- bogus-priv

- dhcp-range=192.168.1.100,192.168.1.200,255.255.255.0,12h

# Note CTRL + X, enter y to save the changes

- sudo systemctl restart dnsmasq # OR: sudo service dnsmasq restart

- sudo systemctl status dnsmasq

- sudo dnsmasq --test

- sudo service -s -status-all | grep -E 'dnsmasq|dhcpd'

- sudo netstat -lnup | grep -E '(:67|:68)'



By: **JPS**

**Reference:** <https://raspberrytips.com/dhcp-server-on-raspberry-pi/>