

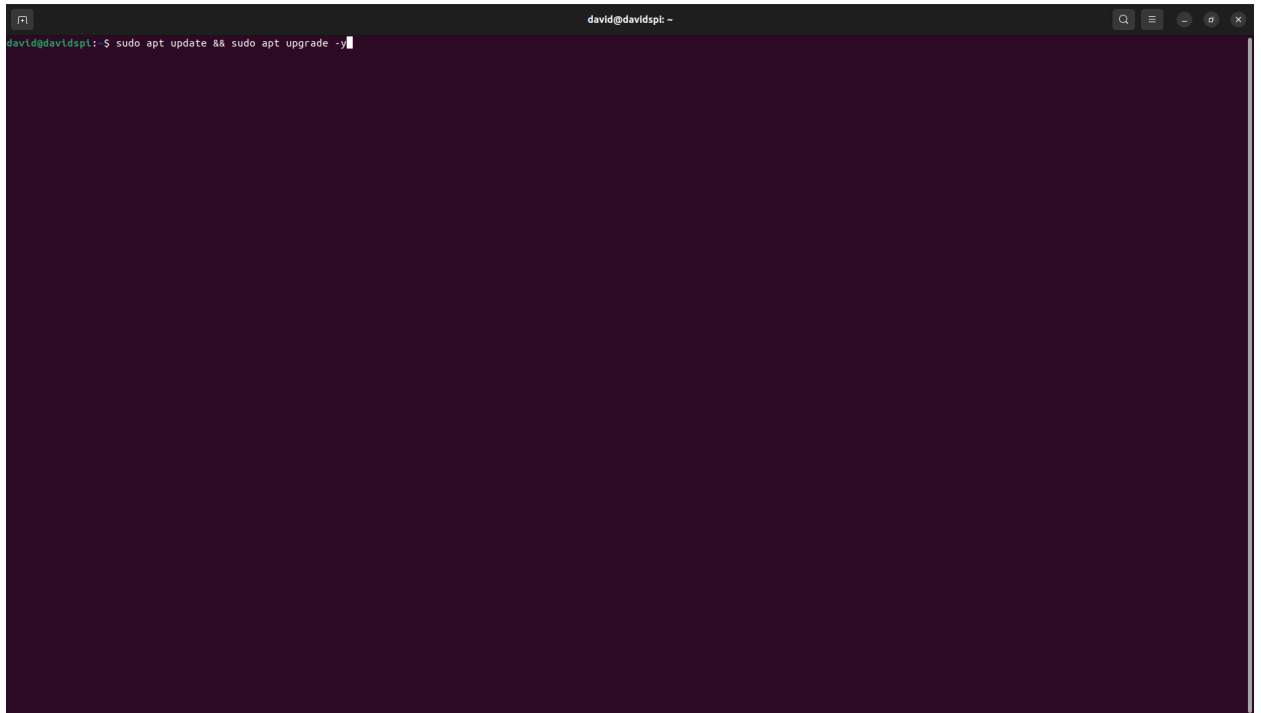
# WireGuard

**Objective:** For this project, I set up a secure VPN using WireGuard. I installed WireGuard on both the server and client machines, generated key pairs, and configured the connection settings to define how the devices communicate. The server was set up to listen on a specific port and handle traffic for certain IP ranges, and the client was configured to connect using the server's public key and address. I also updated firewall rules and routing to make sure everything worked smoothly. In the end, I built a lightweight and fast VPN tunnel that securely connects the devices over the internet.

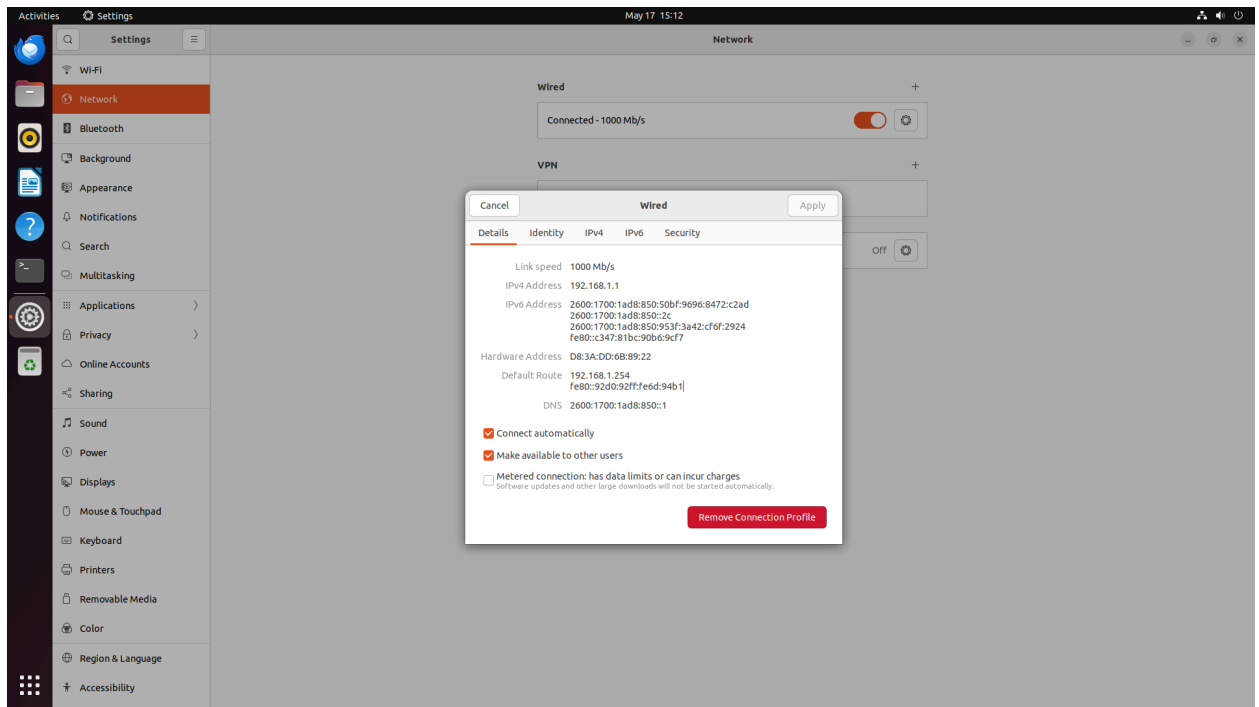
**Equipment:** Raspberry Pi 4, Wireguard, and AT&T Router

## Steps:

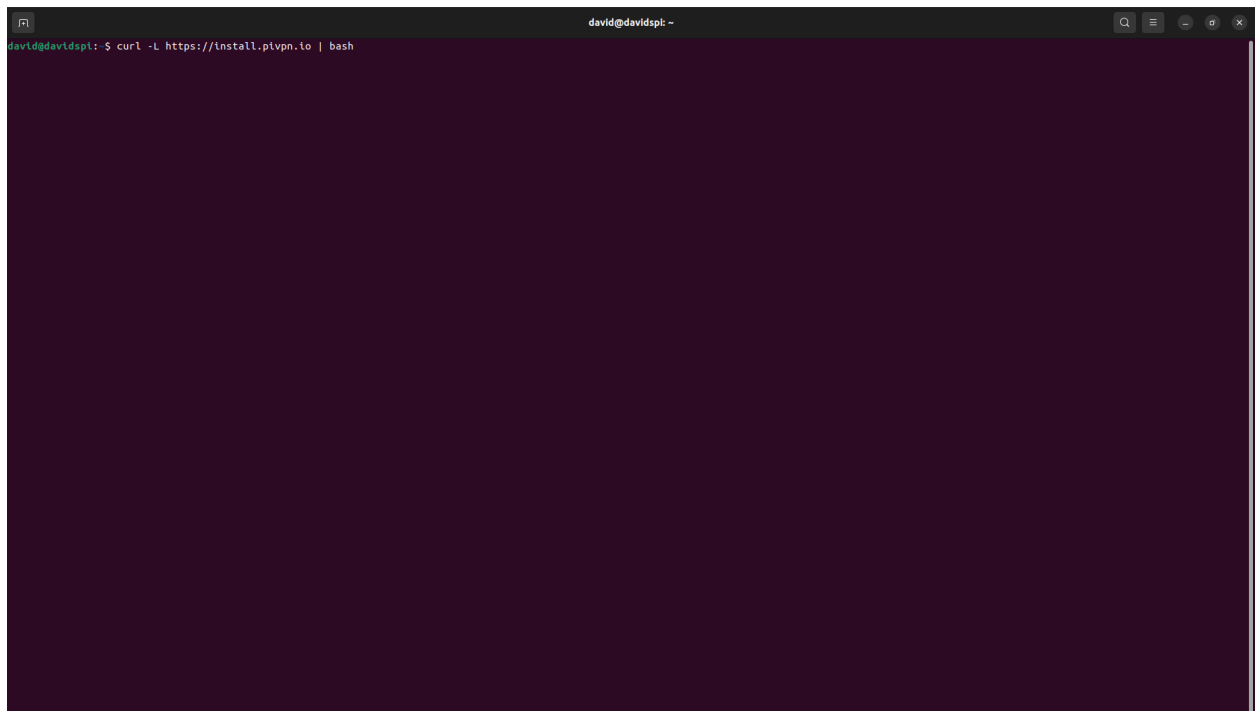
1. I ran the command **sudo apt update && sudo apt upgrade -y** to make sure my pi is up to date

A screenshot of a terminal window on a Raspberry Pi. The window title is 'david@davidspt: -'. The prompt is 'david@davidspt: \$' and the command 'sudo apt update && sudo apt upgrade -y' is entered. The terminal background is dark purple. The window has standard Linux window controls (minimize, maximize, close) in the top right corner.

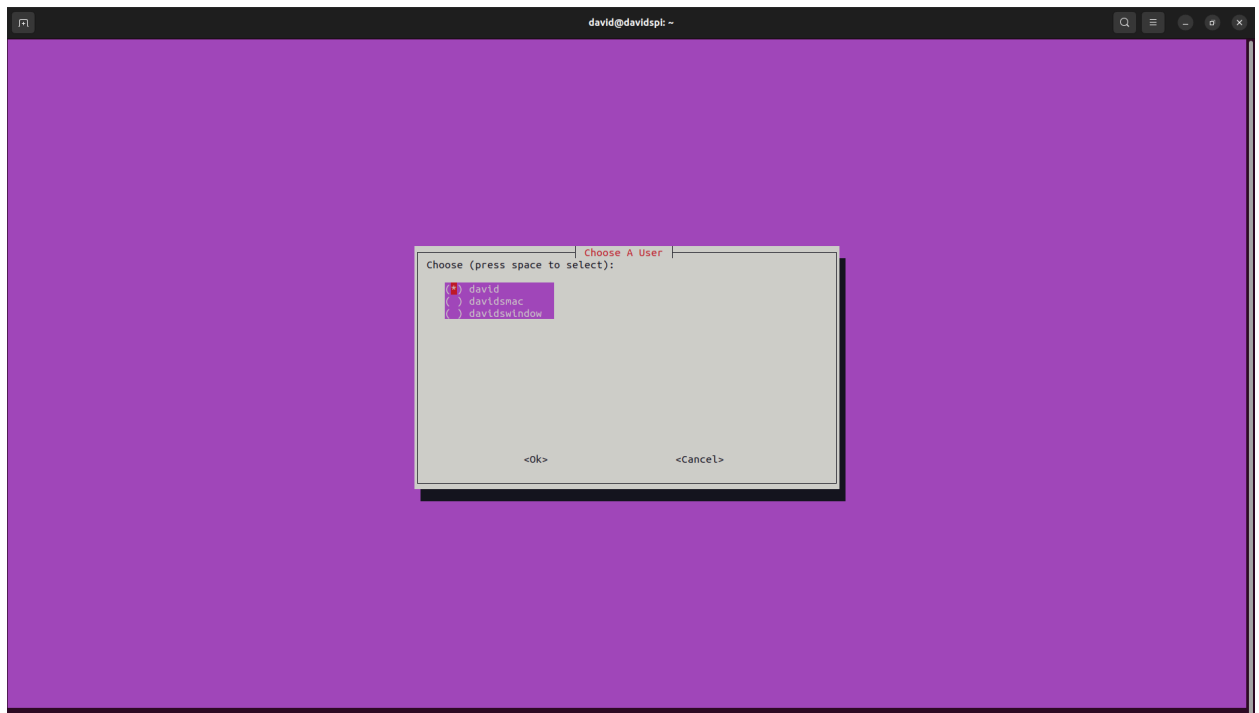
## 2. I created a static IP profile



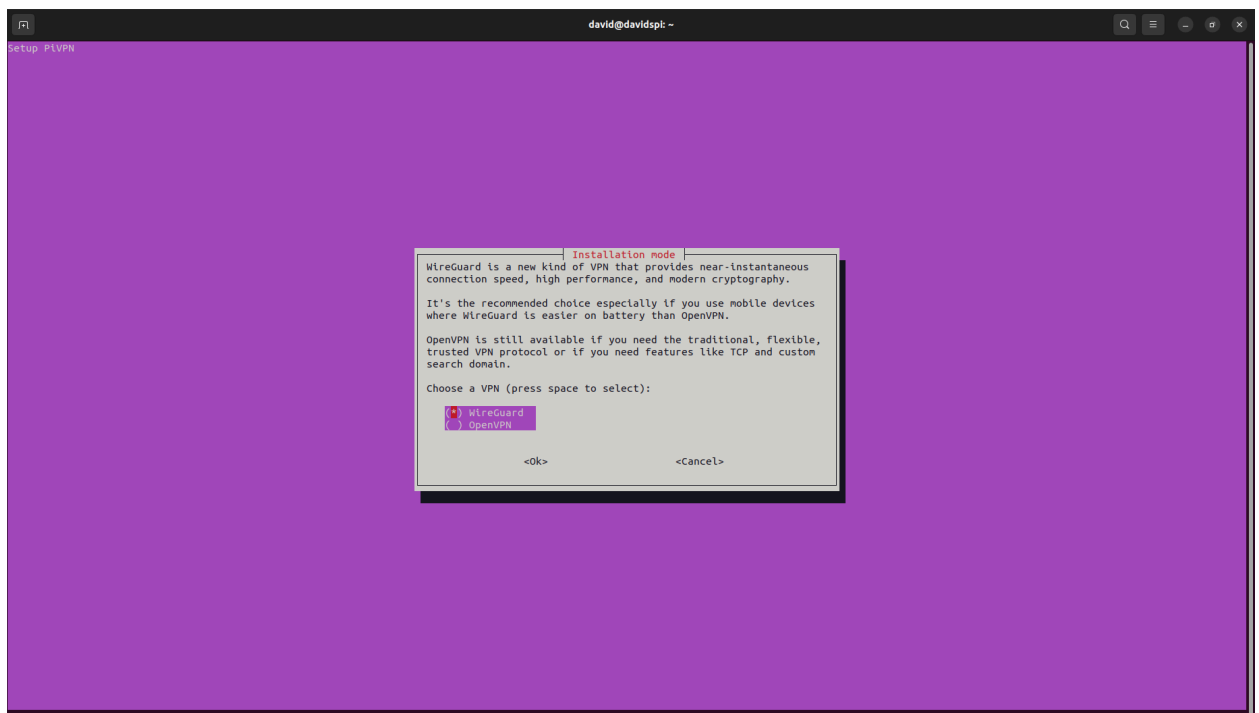
## 3. I ran the command `curl -L https://install.pivpn.io | bash` to install wireguard



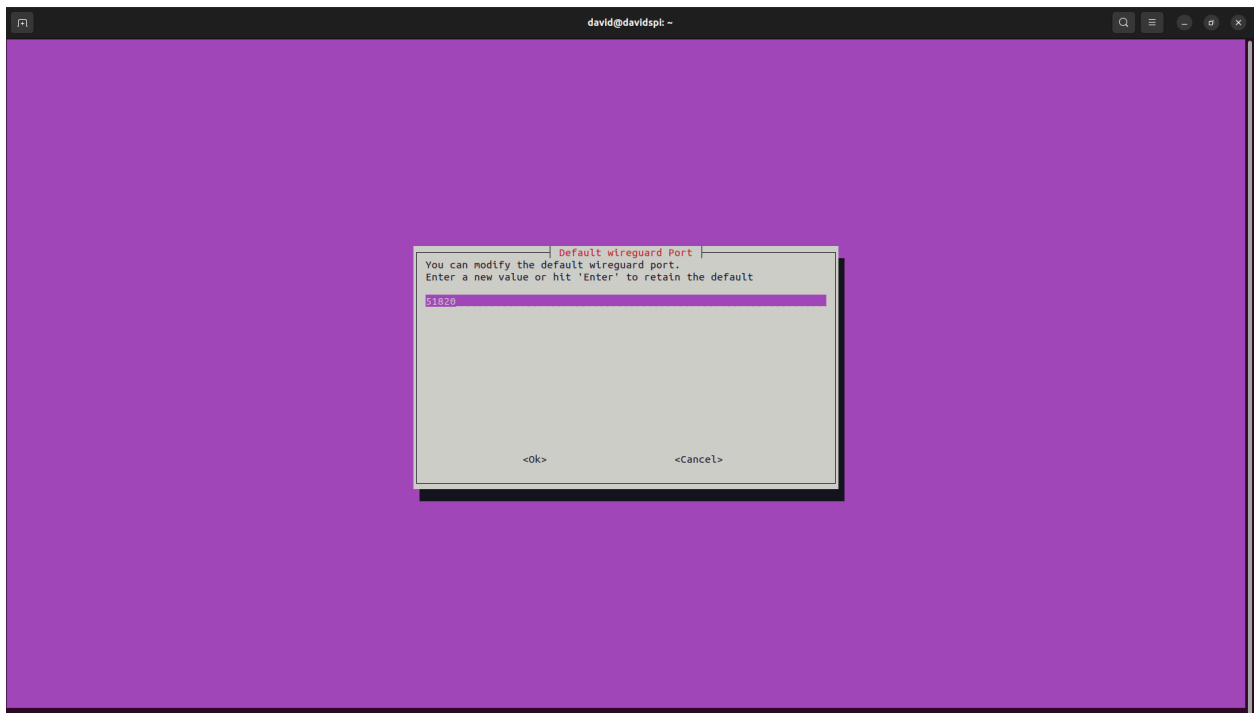
4. I make my **main user on the raspberry pi as the admin**



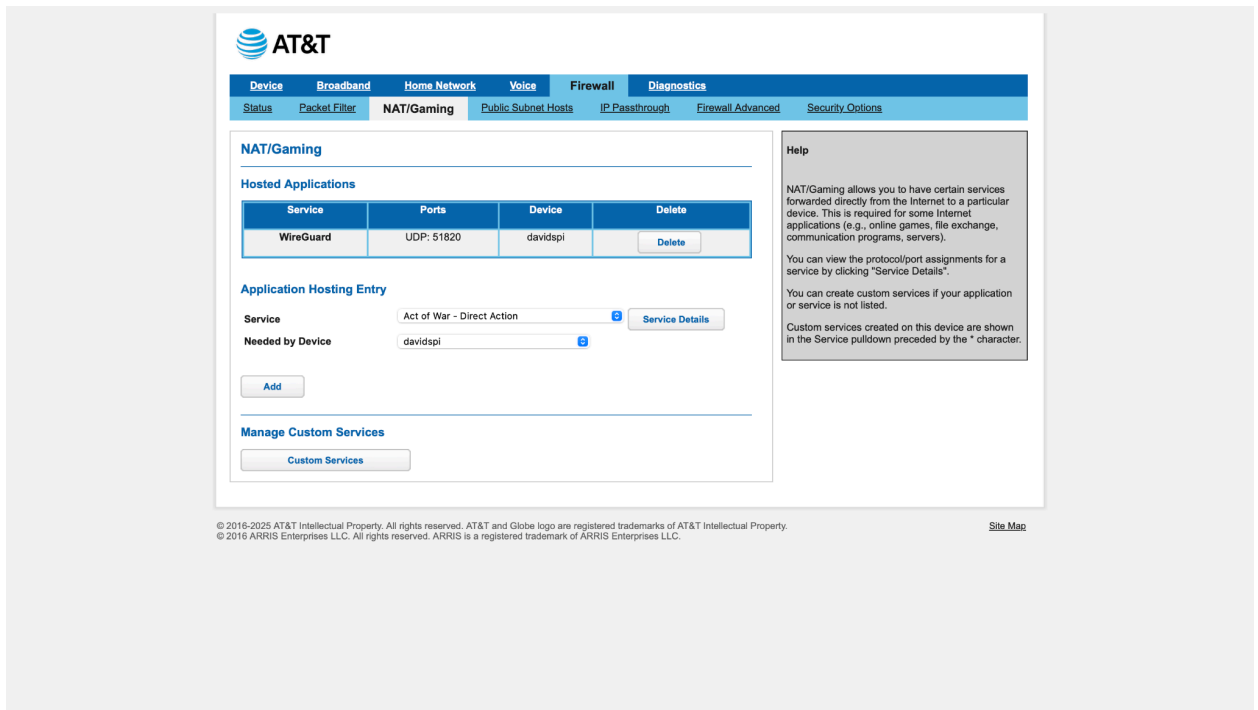
5. I installed WireGuard



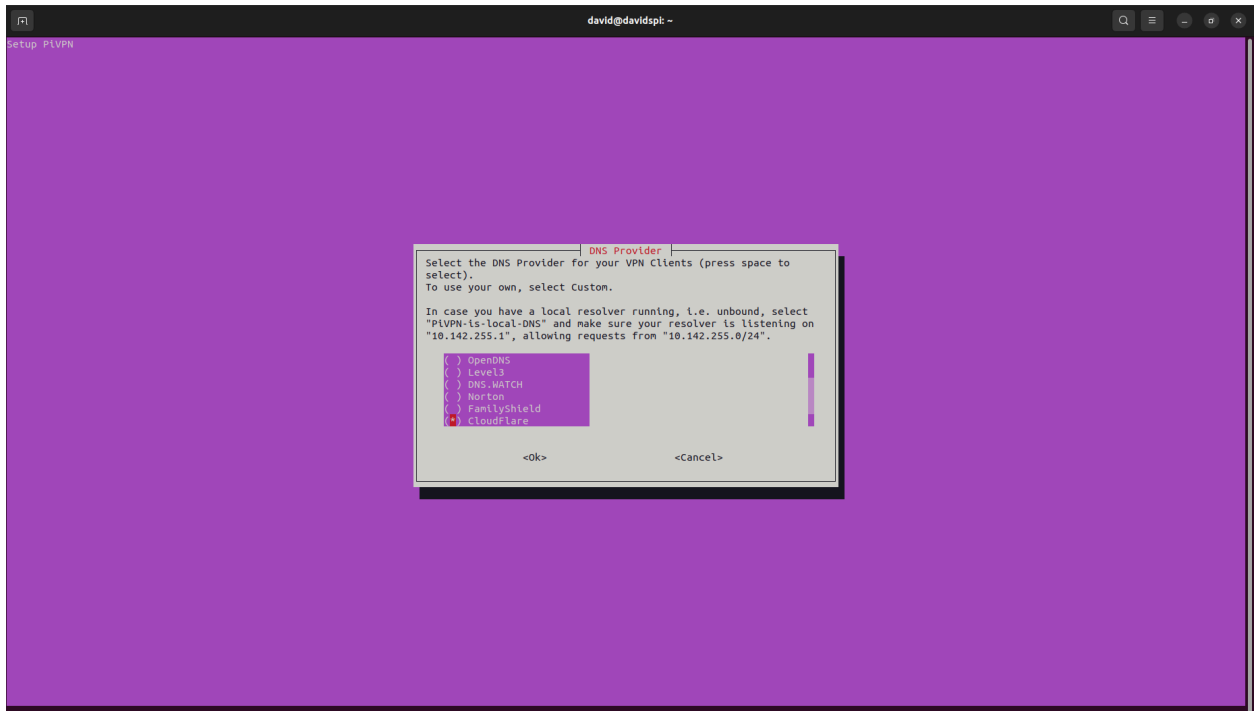
6. I selected the **port 51820**



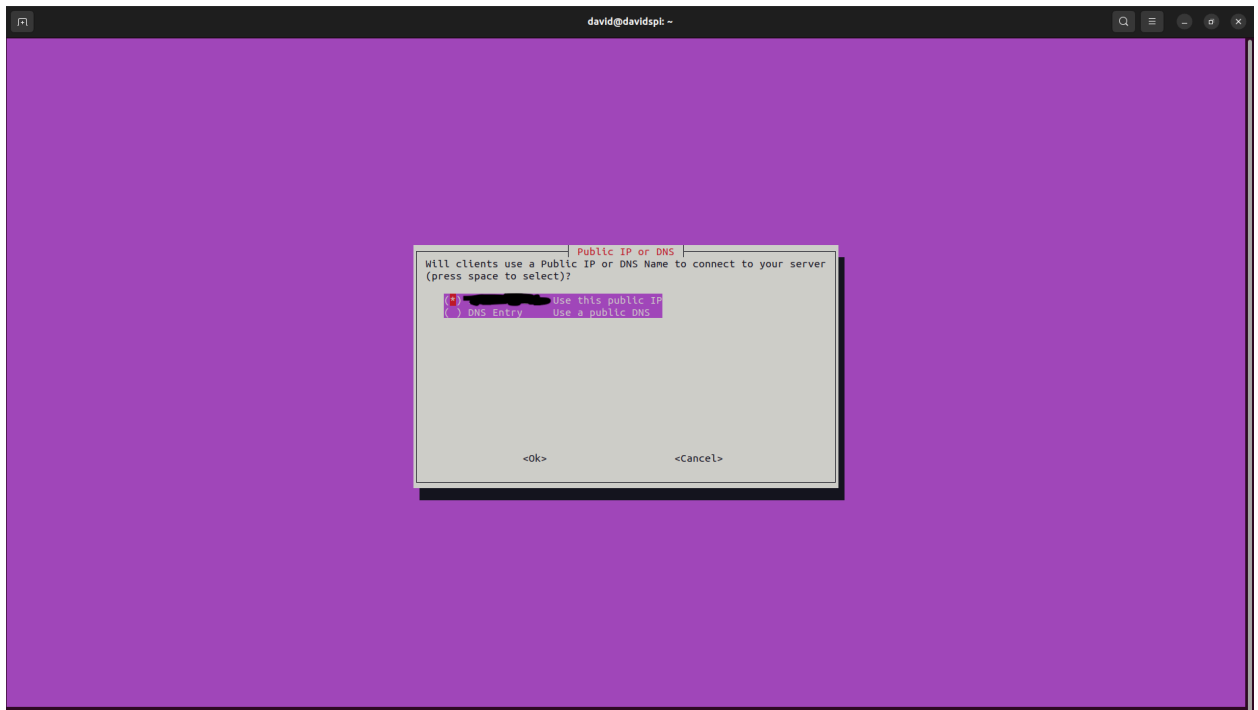
7. I opened the port on my firewall settings within my router



8. I selected cloudflare as the DNS Server



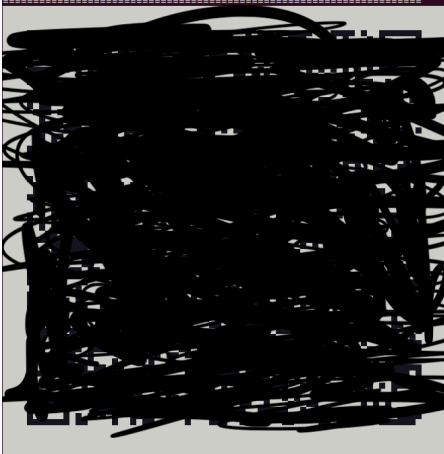
9. I selected my **public IP** address to connect the server



## 10. I created a user my Iphone

```
david@davidspt: ~$ sudo pvpn -a
[sudo] password for david:
Enter the Client IP from range 10.142.255.2 - 10.142.255.254 (optional):
::: Chosen Client IP: 10.142.255.2
Enter a Name for the Client (default: 'davidspt'): DavidsIphone
::: Client Keys generated
::: Client config generated
::: Updated server config
::: WireGuard reloaded
::: Done! DavidsIphone.conf successfully created!
::: DavidsIphone.conf was copied to /home/david/configs for easytransfer.
::: Please use this profile only on one device and create additional
::: profiles for other devices. You can also use pvpn -qr
::: to generate a QR Code you can scan with the mobile app.
david@davidspt: ~$
```

## 11. I generated a QR code for easy setup

```
david@davidspt: ~$ pvpn -qr
::: Client config generated
::: Updated server config
::: WireGuard reloaded
::: Done! DavidsIphone.conf successfully created!
::: DavidsIphone.conf was copied to /home/david/configs for easytransfer.
::: Please use this profile only on one device and create additional
::: profiles for other devices. You can also use pvpn -qr
::: to generate a QR Code you can scan with the mobile app.
david@davidspt: ~$ pvpn -qr
::: Client list ::
1) DavidsIphone
Please enter the Index/Name of the Client to show: DavidsIphone
::: Showing client DavidsIphone below
=====

=====
david@davidspt: ~$
```

12. I configured the tunnel and made sure everything is correct

The screenshot shows the WireGuard configuration interface for a tunnel named "Home-Pi". The interface is dark-themed and includes a back arrow, the "WireGuard" logo, the tunnel name "Home-Pi", and an "Edit" button. The configuration is organized into sections: Tunnel Settings, Peer Settings, and On-Demand Activation. Some fields are redacted with black boxes.

WireGuard Home-Pi Edit	
Name	Home-Pi
Public key	[REDACTED]
Addresses	[REDACTED]
Listen port	51820
MTU	1420
DNS servers	1.1.1.1, 1.0.0.1
PEER	
Public key	[REDACTED]
Preshared key	enabled
Endpoint	[REDACTED]
Allowed IPs	0.0.0.0/0, ::/0
Persistent keepalive	every 25 seconds
ON-DEMAND ACTIVATION	
On-demand	Off

13. I ran the command **sudo wg** to verify the connection

```

david@davidsp1: $ sudo wg
[sudo] password for david:
interface: wg0
  public key: r
  private key: (hidden)
  listening port: 51820

peer: Rq5yBwaw2203meNnHpEYCaB4sb60szGLK9H5KLy9bnY=
  preshared key: (hidden)
  endpoint:
  allowed ips: 10.142.255.2/32,
  latest handshake: 18 seconds ago
  transfer: 811.21 KiB received, 12.67 MiB sent
david@davidsp1: $
```

14. Lastly I tried accessing my home router while on the VPN (as you can see I'm on 5G)

