



headscales

By Richard  
Barlow

A close-up photograph of a server rack. The image is dominated by dark, vertical server units. Several circular indicator lights are visible, some of which are glowing with a bright green light. The lighting is dim, with the primary light source being the green LEDs. The background is dark and out of focus.

## Background

- Needed a solution for backing a remote location nas which is on a wisp after my first idea of using a backup app on the nas didn't workout. I wasn't keen on 8 hours of driving to fix the problem. So, I started to look for a solution to fix this. I settled on using a vpn.
- The first vpn I tried was a L2TP with a pre shared key. Which I then set up on my unifi usg. This did work but was a bit slow.
- I am using a rsync script to pull the files from the remote nas to my server.
- Duplicati is used to upload the backups to Wasabi.



- I wanted to move to wiregrade for my vpn but after some research I discovered that I couldn't use it due to this model of synology nas not supporting docker.
- I Started to investigate other vpn options and came across tailscale which is based off the wiregrade technology.
- This workout as the synology nas had app for tailscale,

 **tailscale**

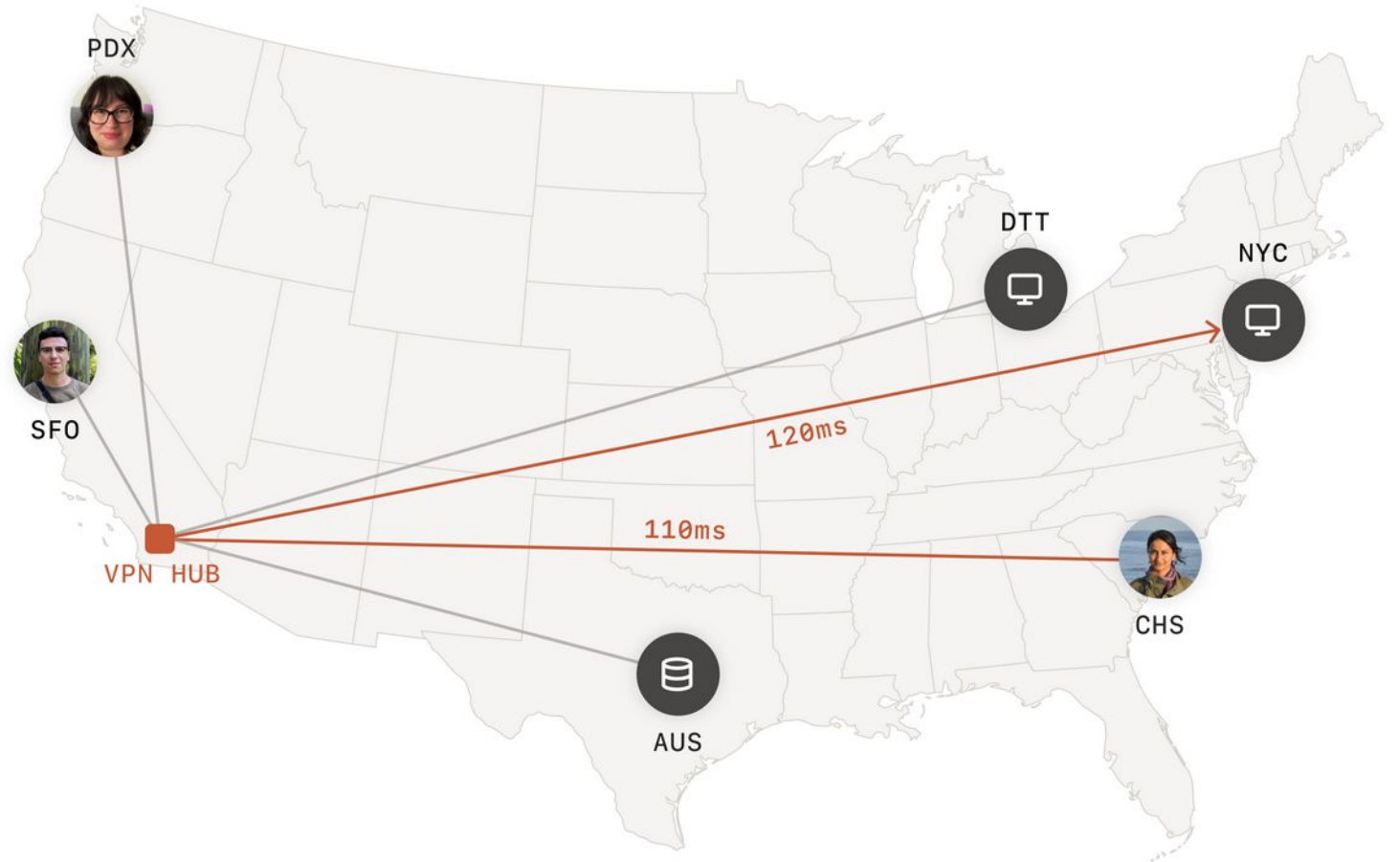


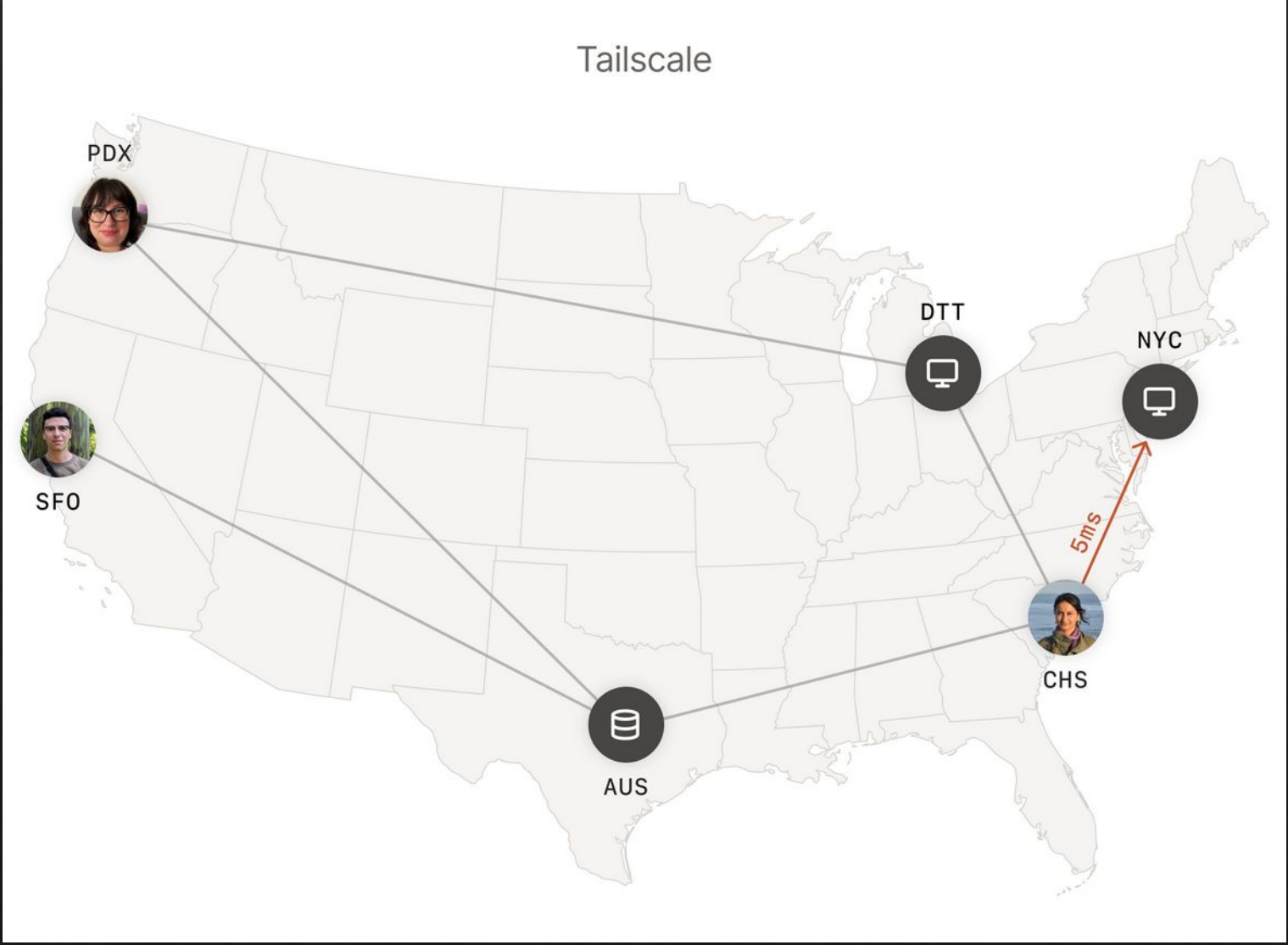
## Difference Between A Traditional And Mesh Vpn


- Traditional VPNs, which tunnel all network traffic through a central gateway server. The central gateway may or may not be close to users, thus resulting in higher latency. Because traffic is centralized, it can also act as a bottleneck, slowing down connections further.
- MeshVpn's, Creates a peer-to-peer mesh network. Mesh approach avoids centralization where possible, resulting in both higher throughput and lower latency as network traffic can flow directly between machines. Additionally, decentralization improves stability and reliability by reducing single points of failure.



## Traditional VPN





- 
- A close-up, low-angle shot of a server rack. The image is dominated by dark, metallic surfaces. Several circular ports or lights are visible, some of which are glowing with a bright green light. In the lower right corner, a red button with a white circular icon is partially visible. The overall lighting is dim, with the green and red lights providing the primary illumination.
- I was very surprised, how easy it was to install and get up and running, both on windows and Linux clients.

## Windows Installed

Just download and install the app and then authenticate the client in your tailscale admin console at <https://tailscale.com/>

## Linux Installed

Add Tailscale's package signing key and repository:

```
curl -fsSLhttps://pkgs.tailscale.com/stable/debian/bookworm.noarmor.gpg | sudo tee /usr/share/keyrings/tailscale-archive-keyring.gpg >/dev/null
```

```
curl -fsSL https://pkgs.tailscale.com/stable/debian/bookworm.tailscale-keyring.list | sudo tee /etc/apt/sources.list.d/tailscale.list
```

A close-up photograph of a network switch or patch panel. The device is dark blue or black with several ports. Two ports in the foreground have bright green lights glowing from them. Below the ports, there are some red buttons or indicators, one of which has a white symbol. The background is dark and out of focus.

2. Install Tailscale:

```
sudo apt-get update
```

```
sudo apt-get install Tailscale
```

3. Connect your machine to your Tailscale network:

```
Sudo tailscale up
```

4. Authenticate the client in your tailscale admin console at

<https://tailscale.com/>

5. You're connected! You can find your Tailscale IPv4 address by running:

```
tailscale ip -4
```



# Tailscale Admin Console

The screenshot shows the Tailscale Admin Console for the account `barlowengineering.nz`. The top navigation bar includes links for `Download`, `Support`, `Docs`, and a user profile icon. Below this is a secondary navigation menu with `Machines` (selected), `Apps`, `Services`, `Users`, `Access controls`, `Logs`, `DNS`, and `Settings`.

## Machines

Manage the devices connected to your tailnet. [Learn more](#) → [Add device](#)

### Build your network

The magic of Tailscale happens when it's installed on multiple devices. How about you use Tailscale to...

- Get started in under 10 min.** (with play button icon)
- kubernetes**: Use the Tailscale Kubernetes operator
- VS Code**: Edit remote files with Tailscale and VS Code
- GitHub**: Connect GitHub Actions to private resources
- /24**: Connect an entire VPC or LAN to your tailnet

Search by name, owner, tag, version... Filters Download

### Get Started

Begin by downloading the Tailscale app and signing in with your `@barlowengineering.nz` account.

[Download Tailscale](#) →




- I was very impressed with tailscale and the speed was noticeably quicker than the L2tp vpn.
- But there was one major issue that was bugging me about tailscale and that was that the controller was at their end.
- So, I started looking into if whether or not there was an open-source version of Tailscale and came across Headscale.
- Headscale is a re-implemented version of the Tailscale coordination server, developed independently and completely separate from Tailscale. Which allows people to use the official Tailscale client with a self-hosted command and control server.

# Headscale Install

I am running my command and control server on a docker container. Here is the docker compose file:

```
1 version: '3.9'
2 services:
3   headscale:
4     container_name: headscale
5     volumes:
6       - /srv/path/etc/headscale/
7       - /srv/path:/var/lib/headscale/
8     ports:
9       - 7070:8080
10      - 9090:9090
11     image: headscale/headscale:0.23.0-alpha3
12     command: headscale serve
13     environment:
14       - TZ=Pacific/Auckland
15     restart: unless-stopped
```



2. Prepare a directory on the Docker host in your directory of choice. This is used to hold headscale configuration and the SQLite database:

```
mkdir -p ./headscales/config  
cd ./headscales
```

3. Create an empty SQLite database in the headscales directory:

```
touch ./config/db.sqlite
```

4. Download a copy of the example config into your config file from the headscales repository :

```
wget -O ./config/config.yaml  
https://raw.githubusercontent.com/juanfont/headscales/main/config-  
example.yaml
```

[Config Example](#)

A close-up photograph of a server rack. The image is dominated by a deep blue color. Several circular ports or lights are visible, each emitting a bright green glow. The focus is sharp on the central part of the rack, with the background and foreground slightly blurred. The overall aesthetic is technical and futuristic.

## 5. Edit the Config File

- Change to your hostname or host IP:  
*server\_url: https://your-host-domain*
- Change the listening port to 0.0.0.0 so it's accessible outside the container:  
*metrics\_listen\_addr: 0.0.0.0:9090*
- *That is the minimum to get Headscale to work.*



Demonstration Time