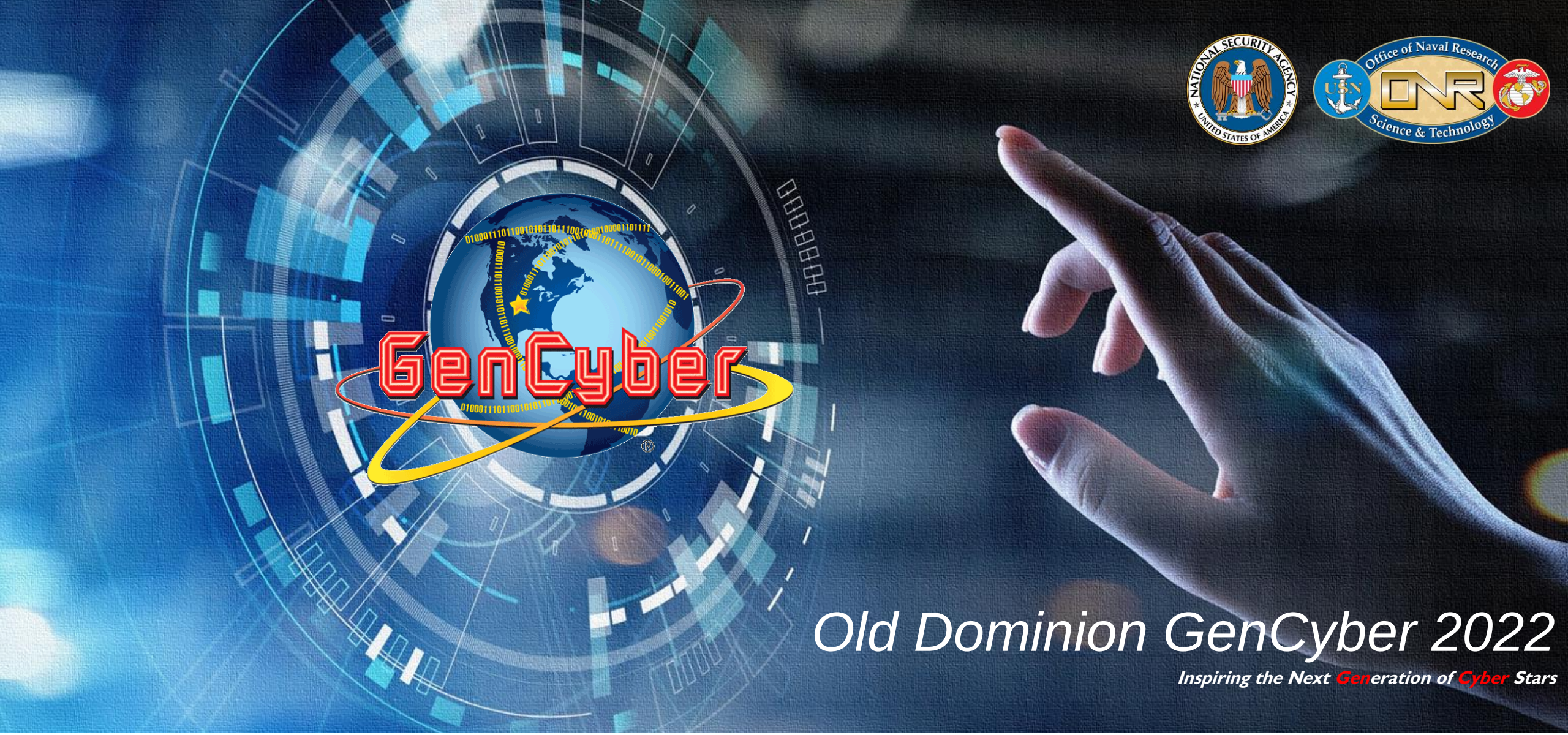


OLD DOMINION UNIVERSITY®

CYBERSECURITY
WWW.ODU.EDU/CCSER

Tutorial: Hosting a VPN with a Raspberry PI

Uddom Lee

Brief Description

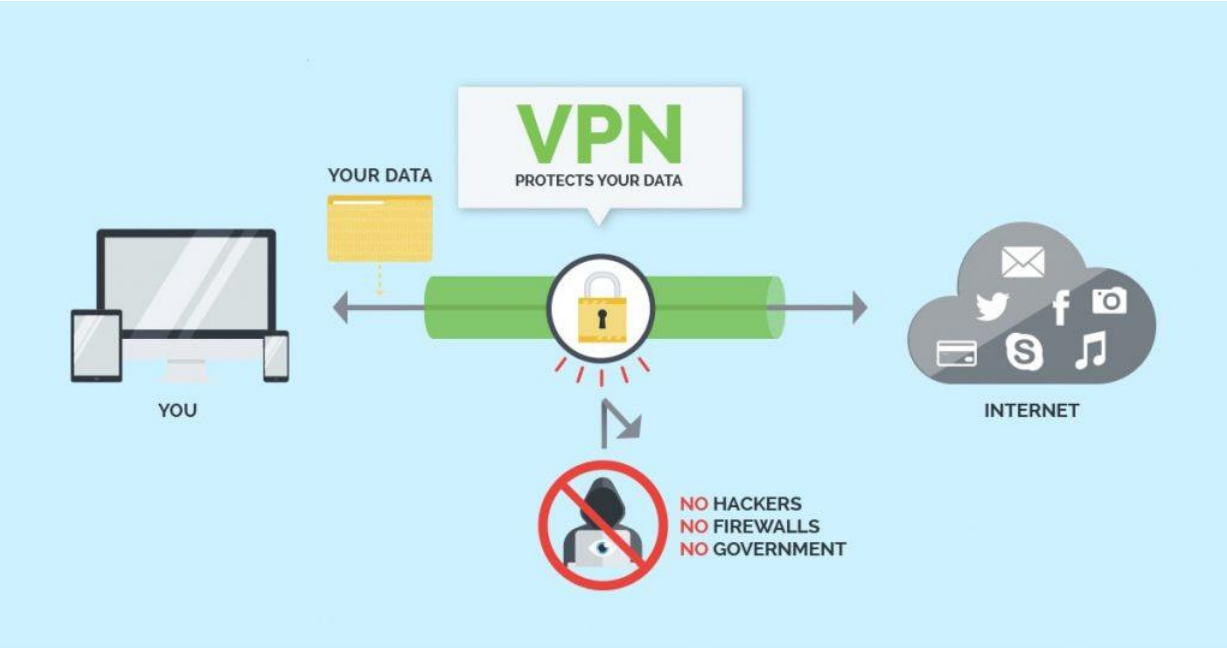
- Tools Required:
  - Raspberry Pi
  - Access to Routers Port Forward Settings
  - Account for freedns.afriaid.org (or any dynamic DNS service)

- Objective:

The goal of this project is for users to setup a home VPN so they can have access to their home network. The user will practice setting up static IPs, CLI commands, editing configurations, and using APIs.

What problems can be solved with a VPN?

- Going to a new location:
  - Coffee Shop
    - Unencrypted Network
  - Need Access to Home Network
  - Want to be in another location



Commercial VPNs vs. Home VPNs

Home VPNs

- Servers are limited to the locations you install them
- Data is protected by you
- You are responsible for patches and updates
- Takes some amount of time to setup

Commercial VPNs

- More servers in different locations
- Data is protected by someone else
- Patches and updates are done automatically
- Plug and play

OPENVPN

WIREGUARD
FAST, MODERN, SECURE VPN TUNNEL

NordVPN

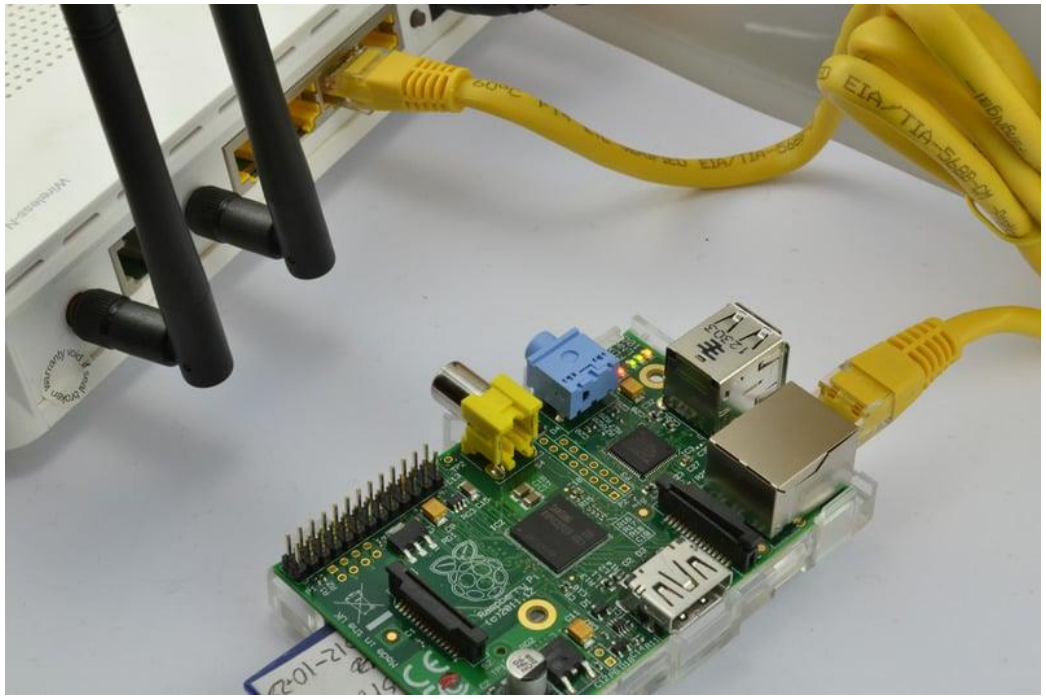
Surfshark

ExpressVPN

Which one would you want to use?

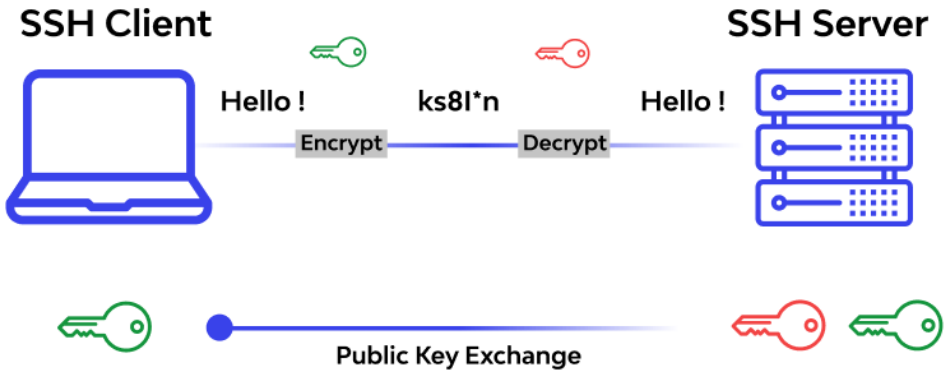
Physical Setup

- Very simple you plug in the Raspberry Pi4B with an ethernet cable or connect it to your wireless network.
  - Note: if you connect it to your wireless network the VPN will be slower and the configurations are slightly different
- Ensure the Pi is in a comfortable place



SSH Setup

- Secure Socket Shell or Secure Shell
- Allows us to securely connect to any network device on our network



SSH Setup Cont.

1. Click the Terminal Icon
2. Type in `sudo raspi-config`
3. Using the arrow keys and enter scroll down to Interfacing Options
4. Select SSH
5. Choose Yes
6. Select Ok
7. SSH is now enabled

SSH Setup Cont.

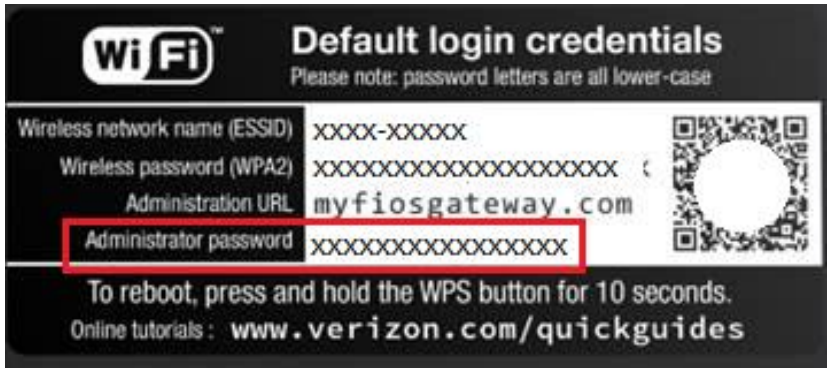
- Type `hostname -I` on the Raspberry Pi
- Take that IP address and the name before the "@"
- Type `ssh [hostname]@[IP address]` on your home computer terminal
- After that follow, the prompt to log into your Pi if you choose to SSH into it
- Within the Raspberry Pi type ifconfig to get the MAC address, we'll use it for later

Pi Setup

- The rest of the Pi can be setup remotely or through the Pi itself.

Getting into your Router

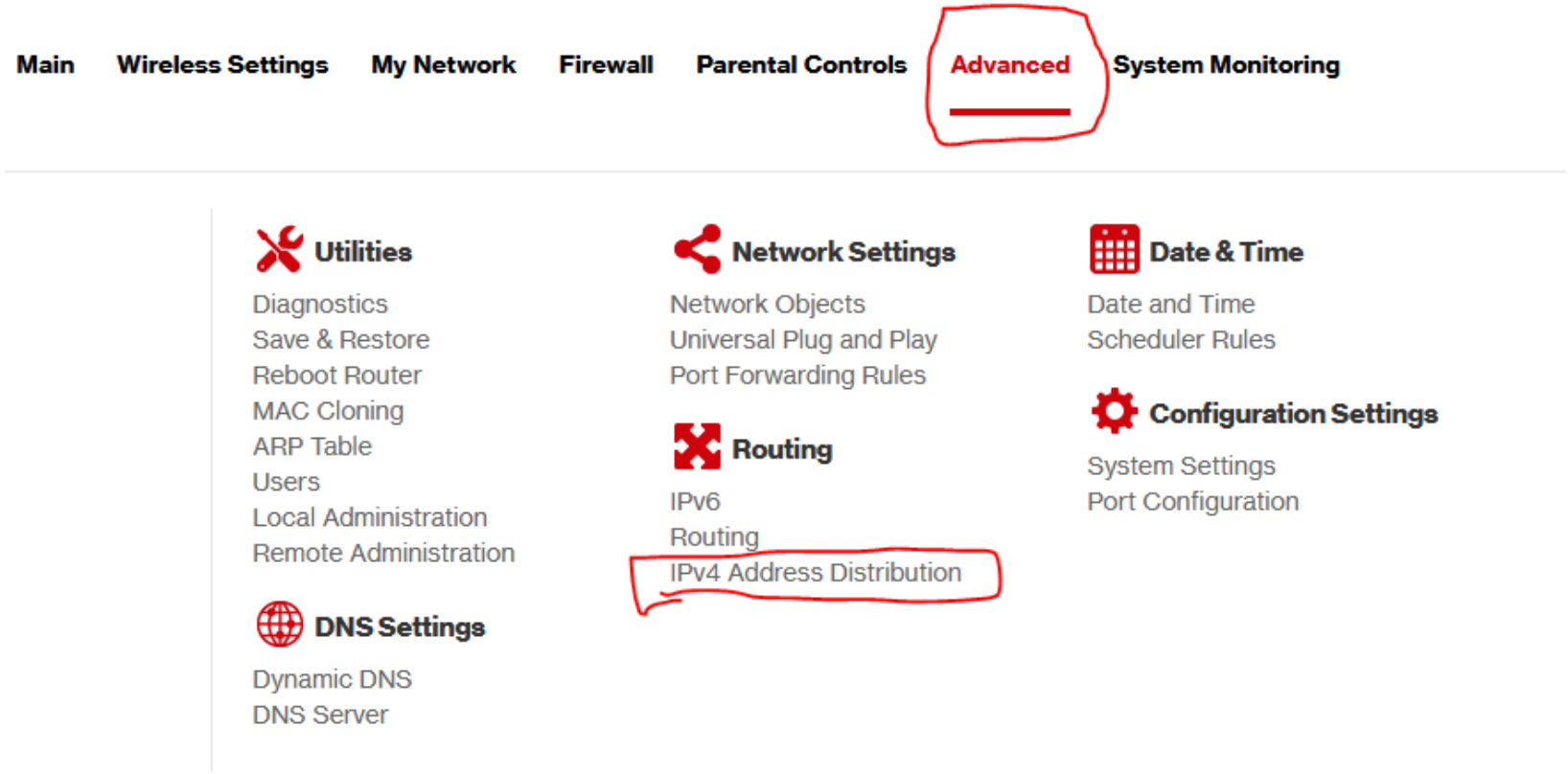
- This will differ depending on the Internet Service Provider (ISP) but the instructions are on the box of your router or on the sticker besides the router



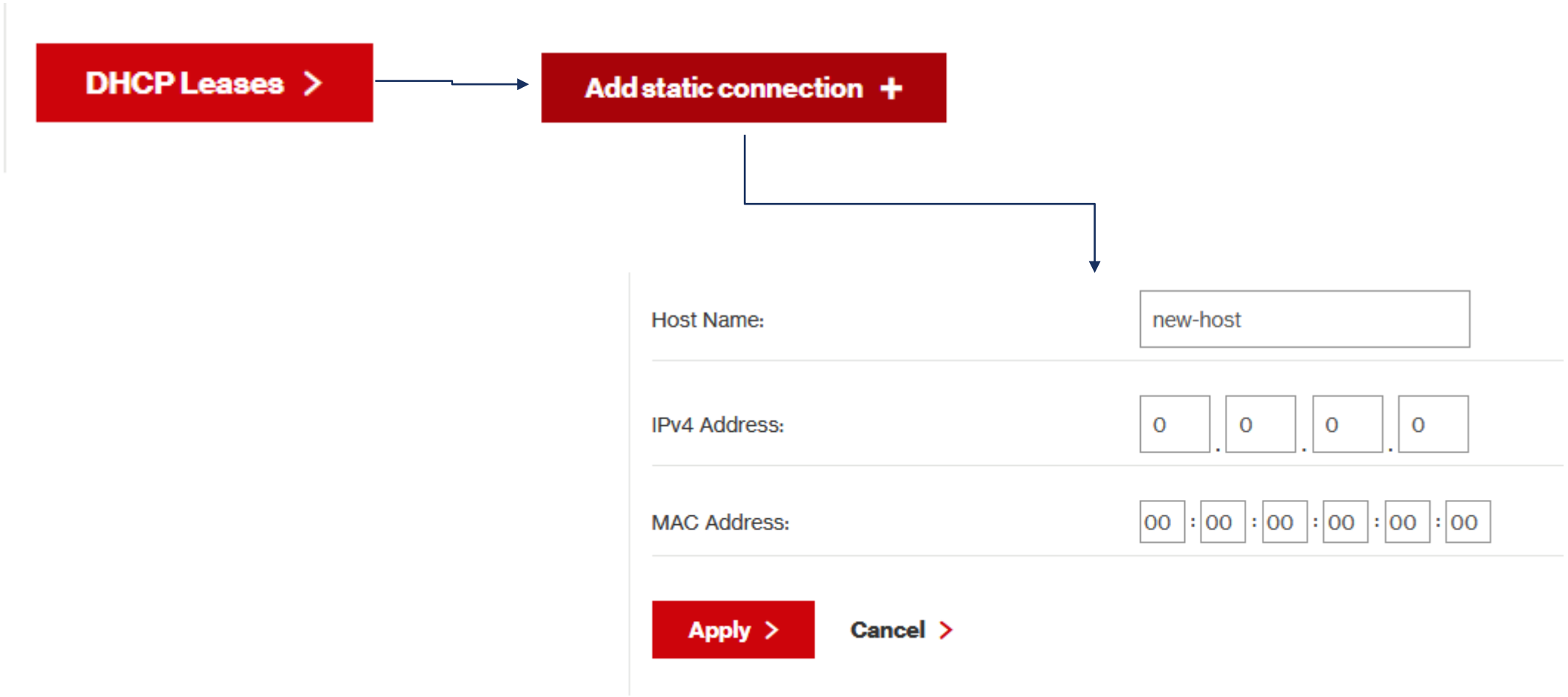
Setting up a Static IP

- To use a VPN and to SSH to the Raspberry Pi we want to ensure the IP does not change
- Whenever a device gets on and off a network unless settings are in place the network will assign the device a new IP
- This will interfere with our settings for the VPN and make SSH into the device more complicated

Setting up a Static IP Cont.



Setting up a Static IP Cont.



Updating our OS

- Type `sudo apt update && sudo apt upgrade`
- Type `sudo restart`
  - This WILL restart the Pi to ensure everything is installed properly
  - Either SSH back into the Pi or use the Pi's terminal to configure everything

What is Dynamic DNS? & Why is it necessary?

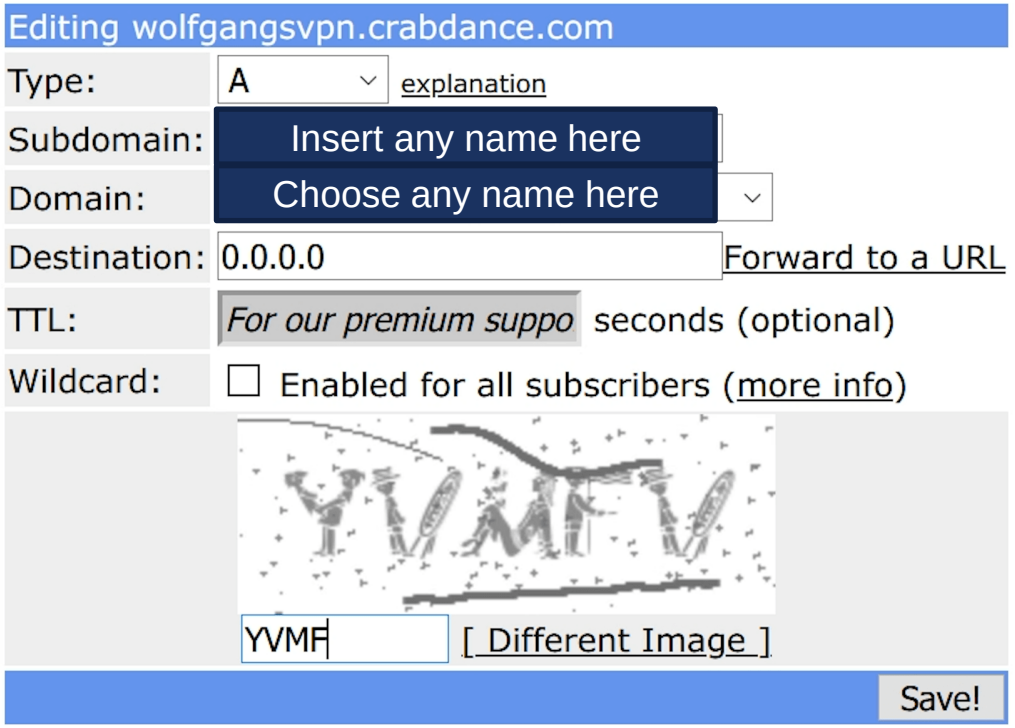
- DNS stands for Domain Named System which allows us to turn 8.8.8.8 -> Google.com
- This is helpful because our devices also have a public IP or a public IP through our router
- This can be found with a google search of what is my ip
  - NOTE: BE VERY CAREFUL WITH THIS INFO. It is almost advertising to others this is my home address

What is Dynamic DNS? & Why is it necessary? Cont.

- Many devices are still using IPv4 which have all been used up.
- ISPs work around this by leasing the IP address out
- So randomly your public IP will change which as mentioned before a moving IP is annoying to work work
- Dynamic DNS will allow us to take our public IP and have it associate with a DNS name

Dynamic DNS

- We will be using a free Dynamic DNS service called `freedns.afraid.org` for this tutorial
- Once your account is created click add a subdomain and enter the information as listed here:
- You can use any domain & subdomain name just remember it or write it down for later



Dynamic DNS Cont.

- Now it is time to install the service that will tell our Dynamic DNS service to update itself
- Type `sudo apt install ddclient`
- Keep pressing enter until the prompts stop. We will change the configurations ourself
- Type `sudo nano /etc/ddclient.conf`

Dynamic DNS Cont.

- Type or Copy and paste this configuration:
- Press Control + O to save & Control + X to exit

```
daemon=5m
timeout=10
syslog=no # log update msgs to syslog
#mail=root # mail all msgs to root
#mail-failure=root # mail failed update msgs to root
pid=/var/run/ddclient.pid # record PID in file.
ssl=yes # use ssl-support. Works with
# ssl-library

use=if, if=eth0
server=freedns.afraid.org
protocol=freedns
login=[YOUR FREEDNS LOGIN]
password=[YOUR FREEDNS PASSWORD]
[your.freedns.domain]
```

If you are on the wireless network change

`if=eth0`
to
`if=wlan0`

Dynamic DNS Cont.

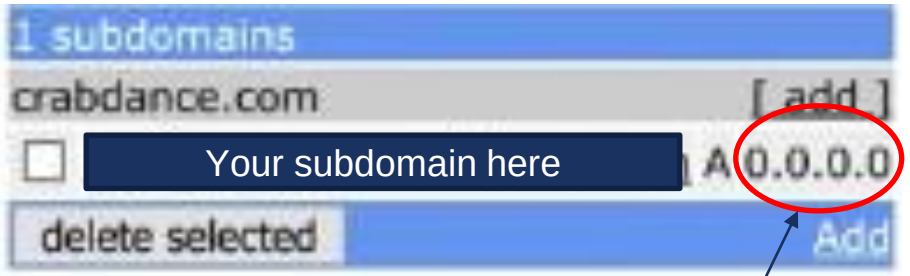
- Type nano `/etc/default/ddclient`
- Change the configurations to match these ones
- Press Control + O to save & Control + X to exit

```
# Set to "true" if ddclient should be run every time DHCP client ('dhclient'
# from package isc-dhcp-client) updates the systems IP address.
run_dhclient="false"

# Set to "true" if ddclient should be run every time a new ppp connection is
# established. This might be useful, if you are using dial-on-demand.
run_ipup="false"

# Set to "true" if ddclient should run in daemon mode
# If this is changed to true, run_ipup and run_dhclient must be set to false.
run_daemon="true"

# Set the time interval between the updates of the dynamic DNS name in
seconds.
# This option only takes effect if the ddclient runs in daemon mode.
daemon_interval="300"
```

Dynamic DNS Cont.

- Type sudo systemctl restart ddclient
- Type sudo systemctl status ddclient
- Type sudo systemctl enable ddclient

```
sudo systemctl status ddclient
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING: Connection: close
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING: Vary: Accept-
Encoding
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING: Cache-Control:
no-store, no-cache, must-revalidate
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING: Cache-Control:
post-check=0, pre-check=0
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING: Pragma: no-cache
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING: Expires: Mon, 26
Jul 1997 05:00:00 GMT
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING: X-Cache: MISS
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING:
Feb 16 15:05:25 raspberrypi ddclient[1806]: WARNING: Updated 1 host(s)
your.freedns.domain to 13.37.420.69
Feb 16 15:05:25 raspberrypi ddclient[1806]: **FAILED: updating
your.freedns.domain: Invalid reply.**
```



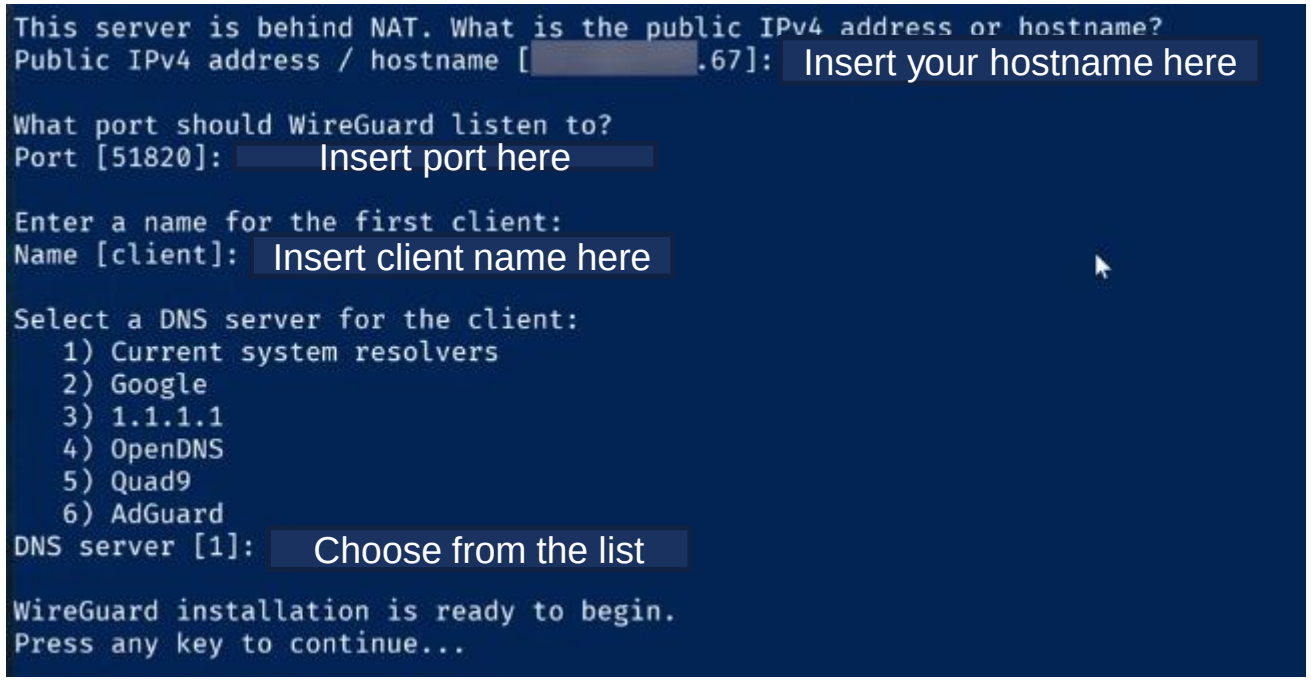
The last line should say failed. On your Dynamic DNS service, the 0.0.0.0 should be updated with the public IP.

Port Forwarding

- Allows us to open a port to the Internet for specific services
- For this tutorial, any port can be used just remember which one you used
- We will be going back into our router

Time to install our VPN service - Wireguard

- Type `wget https://git.io/wireguard -O wireguard-install.sh && sudo bash wireguard-install.sh`
- This will ask you for your hostname type your dynamic dns hostname
- Follow the instructions and wait till everything is installed

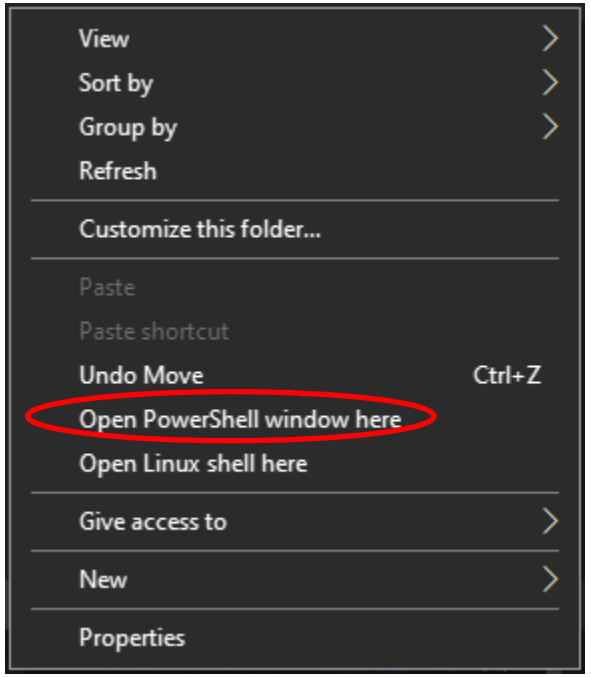


Adding Clients

- Every time you want to add a new user run the previous command again but change the client's name
- For logging purposes, you will know which client is connecting to your VPN
- For phones: using your app store download Wireguard
- Scan the QR code shown

Adding Clients - PC

- We will need to move our client configs to our home directory on the Pi
- Type `sudo su`
- Type `cp /root/*.conf /home/pi`
- Create a folder called Wireguard on your PC
- Within the folder hold Shift and then Right click
- Go to the “Open PowerShell window here”



Adding Clients – Windows Cont.

- Type sftp `[pi hostname]@[pi IP address]`
- You should be at the sftp shell noted by `sftp>` in the terminal
- Type `get *.conf` or `get [client name].conf` if you want a specific client
- Type `exit` or click the X button in powershell
- Download Wireguard and install
- Launch then click import tunnel from file
- Find the file you got from SFTP
- Now you can use the VPN on your Windows PC