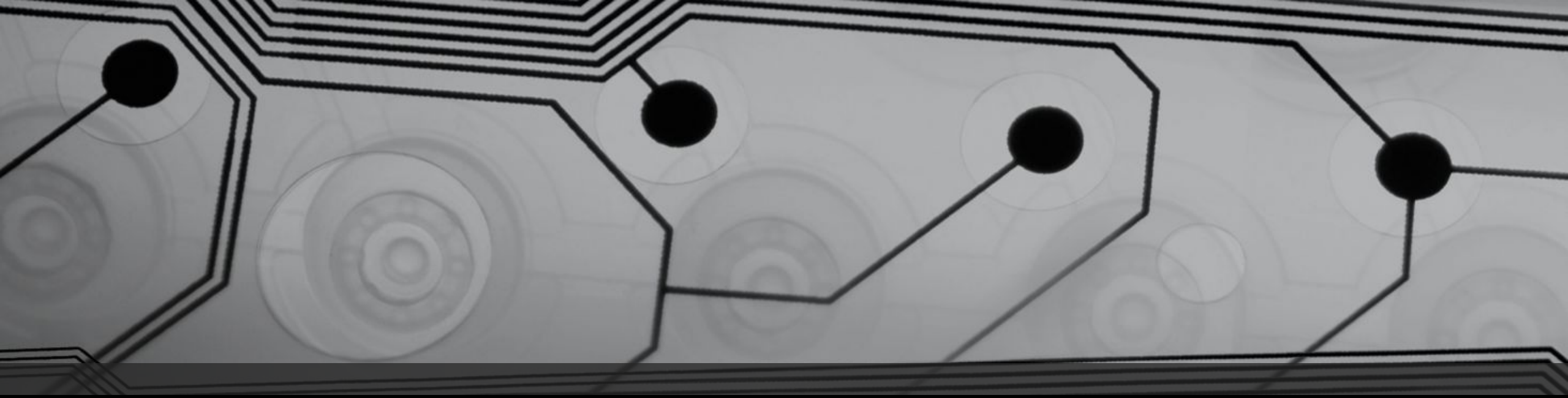

Wireguard

VPN

Packages Installed

- `sudo apt install wireguard`
- `sudo apt install vim`

Set Static IP

- Check network devices, ethernet is likely "Wired connection 1"
  - `sudo nmcli -p connection show`
- Set static ip address
  - `sudo nmcli c mod "Wired connection 1" ipv4.addresses 10.0.0.x/24 ipv4.method manual`
  - `sudo nmcli con mod "Wired connection 1" ipv4.gateway 10.0.0.1`
  - `sudo nmcli c down "Wired connection 1" && sudo nmcli c up "Wired connection 1"`

Choose IPv4 Range

- What ip set is going to be available for vpn clients, cannot overlap with exists local network
- For our purpose 10.1.0.x/24 was chosen

Create Public Private Key

- Create private key
  - `wg genkey | sudo tee /etc/wireguard/private.key`
- Restrict access to root
  - `sudo chmod 0600 /etc/wireguard/private.key`
- Create public key
  - `sudo cat /etc/wireguard/private.key | wg pubkey | sudo tee /etc/wireguard/public.key`

Configure Wireguard

- Create/edit the wg0 file
  - sudo vim /etc/wireguard/wg0.conf

Configure Wireguard

```
[Interface]

PrivateKey = private_key_goes_here

Address = 10.1.0.x

ListenPort = 51820


[Peer]

PublicKey = peer_public_key

AllowedIPs = peer_vpn_ip (10.1.0.x)/24

Endpoint = peer_lan_ip (10.0.0.x) :51820
```

Example

```
[Interface]

PrivateKey = gN6T69MvOj/SOt+sIcIrrDSclaFMQqGyd2kT85qJaGg=

Address = 10.1.0.10

ListenPort = 51820


[Peer]

PublicKey = ayQeKe0pd6bKUlfMZEE4dcXQcops7aYMu4TYCjGylzs=

AllowedIPs = 10.1.0.11/24

Endpoint = 10.0.0.11:51820
```

Enable IPv4 Forwarding

- Edit the sysctl.conf file
  - sudo vim /etc/sysctl.conf
  - Uncomment line net.ipv4.ip_forward=1
    - Or add the line to the bottom of the file
  - Reload file: sudo sysctl -p

Enable Wireguard Connection

Run the following to enable, start, and check wireguard

```
sudo systemctl enable wg-quick@wg0.service
```

```
sudo systemctl start wg-quick@wg0.service
```

```
systemctl status wg-quick@wg0.service
```

Test Wireguard

- Show wireguard peer
  - sudo wg
- Test ping peer
  - ping 10.1.0.x

Adding a Client

On the client host

Create Wireguard Connection on Peer

- Generate public private key pair for the peer

```
wg genkey | sudo tee /etc/wireguard/peer-private.key

sudo chmod 0600 /etc/wireguard/peer-private.key

sudo cat /etc/wireguard/peer-private.key | wg pubkey | sudo
tee /etc/wireguard/peer-public.key
```

Peer Wireguard Configuration

Edit the configuration file:

```
vim /etc/wireguard/wg0.conf
```

Add the following to the configuration file:

```
[Interface]
PrivateKey = private_key_goes_here
Address = 10.9.0.a

[Peer]
PublicKey = server_public_key
AllowedIPs = 10.9.0.b
Endpoint = 10.8.0.c:51820
```

Note

Typically, you would have PostUp and PreDown rules for firewalls rules

```
PostUp = ufw route allow in on wg0 out on eth0

PostUp = iptables -t nat -I POSTROUTING -o eth0 -j
MASQUERADE

PostUp = ip6tables -t nat -I POSTROUTING -o eth0 -j
MASQUERADE

PreDown = ufw route delete allow in on wg0 out on eth0

PreDown = iptables -t nat -D POSTROUTING -o eth0 -j
MASQUERADE

PreDown = ip6tables -t nat -D POSTROUTING -o eth0 -j
MASQUERADE
```

Add Wireguard Client

- On the server
  - `sudo wg set wg0 peer peer_client_public_key allowed-ips 10.9.0.x endpoint 10.8.0.x:51820`
  - `sudo wg`

Start Wireguard Client

- On the client
  - sudo systemctl enable wg-quick@wg0.service
  - sudo systemctl start wg-quick@wg0.service
  - sudo systemctl status wg-quick@wg0.service
- Test connection by pinging the other peer
  - ping 10.9.0.x