WireGuard VPN Setup

Phase 1: Ubuntu Server Setup

1. Install WireGuard

First, update your repositories and install the package.

```
sudo apt update
sudo apt install wireguard -y
```

2. Generate Key Pairs

We need a Private/Public key pair for the Server, and a Private/Public key pair for the Client. We will generate both on the server for convenience.

```
# Move to the WireGuard directory and set restrictive permissions
cd /etc/wireguard
umask 077
```

```
# Generate Server Keys
wg genkey | tee server_private.key | wg pubkey > server_public.key
```

```
# Generate Client Keys
wg genkey | tee client_private.key | wg pubkey > client_public.key
```

3. Create the Server Configuration

Find your default network interface name (usually eth0, ens3, etc.) because we need it for traffic routing.

```
ip route list default
```

(Note the word after dev. E.g., default via ... dev eth0 means your interface is eth0.)

Create the configuration file:

```
sudo nano /etc/wireguard/wg0.conf
```

Paste the following configuration into the file. **Replace <SERVER_PRIVATE_KEY_CONTENT>** with the content of server_private.key you generated earlier.

```
[Interface]
# The IP address of this server inside the VPN network
Address = 10.8.0.1/24
SaveConfig = true
PostUp = ufw route allow in on wg0 out on eth0
PostUp = iptables -t nat -I POSTROUTING -o eth0 -j MASQUERADE
PostDown = ufw route delete allow in on wg0 out on eth0
PostDown = iptables -t nat -D POSTROUTING -o eth0 -j MASQUERADE
ListenPort = 51820
PrivateKey = <SERVER_PRIVATE_KEY_CONTENT>
```

```
[Peer]
# The Windows Client
PublicKey = <CLIENT_PUBLIC_KEY_CONTENT>
AllowedIPs = 10.8.0.0/24
```

Note: If your server interface found earlier was not eth0, replace eth0 in the PostUp/PostDown lines with your actual interface name.

4. Enable IP Forwarding

For the VPN to route internet traffic, you must enable packet forwarding.

Open sysctl.conf:

```
sudo nano /etc/sysctl.conf
```

Find and uncomment (remove the #) this line:

```
net.ipv4.ip_forward=1
```

Save and apply changes:

```
sudo sysctl -p
```

5. Configure Firewall (UFW)

Allow the WireGuard UDP port and SSH (so you don't lock yourself out), then enable the firewall.

```
sudo ufw allow 51820/udp
sudo ufw allow OpenSSH
sudo ufw enable
```

(Check status with sudo ufw status).

6. Start WireGuard

```
sudo systemctl enable wg-quick@wg0
sudo systemctl start wg-quick@wg0
```

Phase 2: Windows Client Setup

1. Prepare Client Config

Before leaving your Ubuntu server, print the values you will need for the Windows client.
Run this command to see the keys you generated earlier:

```
cat server_public.key client_private.key
```

2. Install Client

Download and install the official Windows Installer from wireguard.com.
https://www.wireguard.com/install/

3. Configure the Tunnel

1. Open the WireGuard application on Windows.
2. Click the arrow next to **"Add Tunnel"** and select **"Add empty tunnel..."**.
3. Name it wg0.

4. Paste the following configuration (replace placeholders with actual values):

```
[Interface]
# The Client's Key (from client_private.key)
PrivateKey = <CLIENT_PRIVATE_KEY_CONTENT>
# The Client's IP inside the VPN
Address = 10.8.0.2/24
# DNS (Optional, Google DNS used here)
DNS = 8.8.8.8

[Peer]
# The Server's Public Key (from server_public.key)
PublicKey = <SERVER_PUBLIC_KEY_CONTENT>
# The Public IP address of your Ubuntu VPS + Port
Endpoint = <YOUR_UBUNTU_PUBLIC_IP>:51820
# Route ALL traffic through VPN (0.0.0.0/0)
AllowedIPs = 10.8.0.0/24
# Keeps connection alive if behind NAT
PersistentKeepalive = 25
```

5. Click **Save**.
6. Click **Activate**.

Phase 3: Verification & Troubleshooting

1. **Check Connection:** On Windows, open a browser and search "What is my IP". It should show the IP address of your Ubuntu server, not your home ISP.
2. **Ping Test:** On Windows command prompt: ping 10.8.0.1.