Your own VPN

- This guide will help you set up wireguard VPN with pihole as the DNS server.

What is pi-hole

Pi-hole is basically a DNS level adblocker. Now for people who are new to this,

What is DNS

- DNS stands for Domain Name Server and is basically the backbone of the modern internet. The internet works on IP addresses ie if you go 172.217.167.238 using your browser , you would be directed to google.com. Beautiful right ?
- But as humans we can't be expected to remember these ip addresses, This is where DNS comes in .So when you go to google.com on your browser , your browser asks the DNS server you have set - What is the Ip of google.com and it gets a response and hence it takes you to that ip.
- Here is a more detailed guide on how DNS works

What is Pi-hole then ?

- When you load up a site with ads, it phones home on several different domains for ads and when you use regular DNS servers, they just allow those domains to resolve .
- Now when you change your DNS to pi-hole , as soon as a site phones home for ads pi-hole blocks that DNS request and that's how the ads get blocked.

What is wireguard

- Wireguard is this new age VPN technology that consumes less battery, processing power than openvpn and is just plain faster.

What are we going to setup today

?

We are basically going to setup your own VPN server which will have it's dns set as pihole and hence give you adblocking whenever you connect to your VPN

Docker-compose file

For this setup we are going to use docker-compose paste the below into a docker-compose.yml file and then run

```
docker-compose up -d
```

in the same directory where you have saved this file

docker-compose.yml

```
version: "3"

networks:
  private_network:
    ipam:
      driver: default
      config:
        - subnet: 10.2.0.0/24

services:
  unbound:
    image: "klutchell/unbound:latest"
    container_name: unbound
    restart: unless-stopped
    hostname: "unbound"
    volumes:
      - "./unbound:/opt/unbound/etc/unbound/"
    networks:
      private_network:
        ipv4_address: 10.2.0.200
```

```
  wireguard:
    depends_on: [unbound, pihole]
    image: linuxserver/wireguard
    container_name: wireguard
    cap_add:
      - NET_ADMIN
      - SYS_MODULE
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=America/Los_Angeles # Change to your timezone
      - SERVERPORT=51820
      #- SERVERURL=my.ddns.net #optional - For use with DDNS (Uncomment to use)
      - PEERS=1 # How many peers to generate for you (clients)
      - PEERDNS=10.2.0.100 # Set it to point to pihole
      - INTERNAL_SUBNET=10.6.0.0

    volumes:
      - ./wireguard:/config
      - /lib/modules:/lib/modules
    ports:
      - "51820:51820/udp"
    dns:
      - 10.2.0.100 # Points to pihole
      - 10.2.0.200 # Points to unbound
    sysctls:
      - net.ipv4.conf.all.src_valid_mark=1

    restart: unless-stopped
    networks:
      private_network:
        ipv4_address: 10.2.0.3

  pihole:
    depends_on: [unbound]
    container_name: pihole
    image: pihole/pihole:latest
    restart: unless-stopped
    hostname: pihole
```

```
    dns:
      - 127.0.0.1
      - 8.8.8.8 # Points to unbound
    environment:
      TZ: "America/Los_Angeles"
      WEBPASSWORD: "" # Blank password - Can be whatever you want.
      ServerIP: 10.1.0.100 # Internal IP of pihole
      DNS1: 8.8.8.8 # Unbound IP
      DNS2: 8.8.8.8 # If we don't specify two, it will auto pick google.
    # Volumes store your data between container upgrades
    volumes:
      - "./etc-pihole/:/etc/pihole/"
      - "./etc-dnsmasq.d/:/etc/dnsmasq.d/"
    # Recommended but not required (DHCP needs NET_ADMIN)
    #   https://github.com/pi-hole/docker-pi-hole#note-on-capabilities
    cap_add:
      - NET_ADMIN
    networks:
      private_network:
        ipv4_address: 10.2.0.100
```

Step 2

- Now that you have the containers running , just do a docker-compose logs -f and you will see a qr code.
- Install the wireguard android app and scan this qr code to add this tunnel
- Now try connecting to the tunnel
- Congrats you have a vpn server now which has adblocking built into it

Configuring a split tunnel

- Wireguard also lets you configure something called a split tunnel which in my opinion is an amazing feature.
- So when you connect to your vpn server, your internet speed is bottlenecked by the speed of the VPN server.
- To get fast speeds and adblocking as well you can configure what is called a split tunnel, which means that only your dns queries will be sent to the server and all the other queries

will be routed directly
- To do that change the AllowedIPs in the wireguard config to `10.2.0.0/24`.

Troubleshooting

- For troubleshooting the first step is to check if all containers are still alive `docker-compose logs -f`
- Now if the containers are running properly, check if port 51820/udp is accessible over the internet.

Revision #2
Created 20 June 2021 16:16:30 by Manav Sethi
Updated 21 June 2021 07:03:22 by Manav Sethi