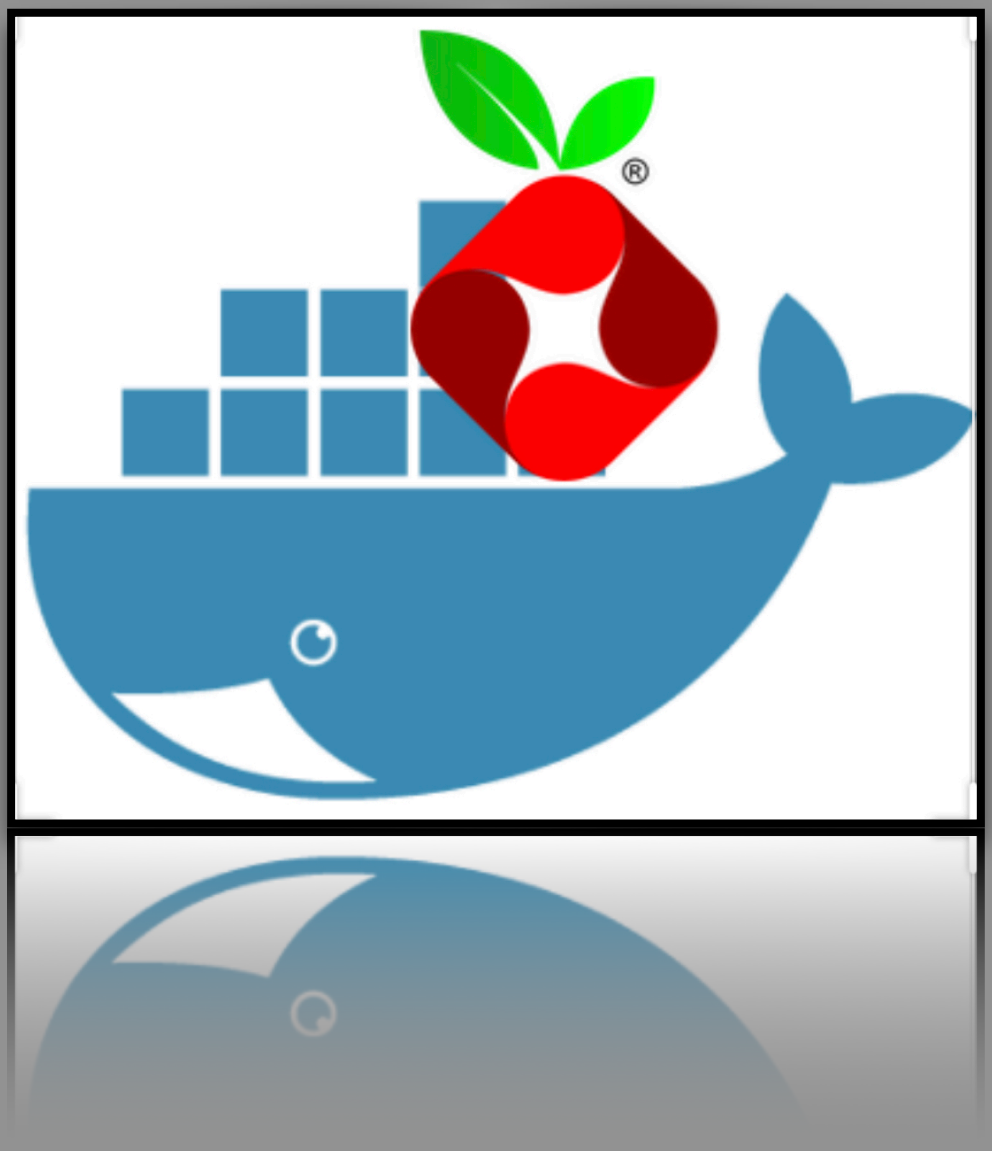
DOCKER SERVER SETUP



DOCKER ON UBUNTU 20.04
WITH PIHOLE IMAGE

INSTALL DOCKER ENGINE

```
$ sudo apt-get update

$ sudo apt-get install \
  apt-transport-https \
  ca-certificates \
  curl \
  gnupg-agent \
  software-properties-common

$ curl -fsSL https://download.docker.com/linux/ubuntu/
gpg | sudo apt-key add -

$ sudo apt-key fingerprint 0EBFCD88

$ sudo add-apt-repository \
```

"deb [arch=amd64] https://download.docker.com/linux/ubuntu \
$(lsb_release -cs) \
stable"

INSTALL DOCKER COMPOSE

$sudo apt install apt-transport-https ca-certificates curl software-properties-common

$curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

$sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"

Enable a firewall on your server

It is necessary to block off your Pi-Hole from random incoming traffic since it could be hijacked for DNS amplification attacks if made public. I am using a server running Ubuntu 18.04, so my firewall of choice is UFW.

You'll need to block ports 53 and 80 for incoming traffic.

Also, allow any traffic that will be coming from your wireguard clients:

```
# Example of firewall rules

$ ufw allow 41194/udp
$ ufw allow 22/any
$ ufw deny 80/any
$ ufw deny 53/any
$ ufw allow from 10.0.0.0/24
$ ufw reject https
```

ADD PI-HOLE CONTAINER

$ docker pull pihole/pihole

 # Create a new text file with the .sh extension (I'll call it skyHole.sh).

$ nano skyHole.sh

Insert the following:


```
#!/bin/bash

# https://github.com/pi-hole/docker-pi-hole/blob/master/README.md
```

```
docker run -d \
   --name pihole \
   -p 192.168.20.1:53:53/tcp -p 192.168.20.1:53:53/udp \
   -p 192.168.20.1:80:80 \
   -p 192.168.20.1:443:443 \
   -e TZ="America/New_York" \
   -v "$(pwd)/etc-pihole/:/etc/pihole/" \
   -v "$(pwd)/etc-dnsmasq.d/:/etc/dnsmasq.d/" \
   --dns=127.0.0.1 --dns=1.1.1.1 \
   --restart=unless-stopped \
   pihole/pihole:latest

printf 'Starting up pihole container '
for i in $(seq 1 20); do
   if [ "$(docker inspect -f "{{.State.Health.Status}}" pihole)"
== "healthy" ] ; then
      printf ' OK'
      echo -e "\n$(docker logs pihole 2> /dev/null | grep
'password:') for your pi-hole: https://${IP}/admin/"
      exit 0
   else
      sleep 3
```

```
    printf '.'
  fi

  if [ $i -eq 20 ] ; then
    echo -e "\nTimed out waiting for Pi-hole start start,
consult check your container logs for more info (\`docker
logs pihole\`)"
    exit 1
  fi
done;
```

RUN THE DOCKER FILE

 # Make the script executable.

$ chmod +x skyHole.sh

 # Run it by executing this command

$ ./skyHole.sh

#Check if the container is running with

$ docker ps -a

 # Pi-Hole's web interface on 192.168.20.1:80/admin/index.php from any device connected to the wireguard tunnel.

All your client's traffic should be routed through the tunnel.

ENTER THE PI-HOLE CONTAINER

 # Change pi-hole password

$ docker exec -it pihole pihole -a -p

 # Reconfigure pi-hole interface

$ docker exec -it pihole pihole -r

https://www.reddit.com/r/pihole/duplicates/bl4ka8/
guide_pihole_on_the_go_with_wireguard/
mkdir /etc/pihole

$ curl -fsSL https://get.docker.com -o get-docker.sh
$ sudo sh get-docker.sh
$ sudo usermod -aG docker lilgubna
Password:
$ docker stop pihole
$ docker rm pihole
$ docker pull pihole/pihole:latest
$ docker-compose up -d
$ curl -fsSL https://get.docker.com -o get-docker.sh
$ docker image prune <-remove old image files
$ sudo systemctl status docker