

# OpenWRT Netbird

version 5

Latest version:

<https://raw.githubusercontent.com/egc112/OpenWRT-egc-add-on/main/notes/OpenWRT%20Netbird.pdf>

This is a WIP and just some poorly redacted personal notes, I am working to make a real install guide

## Introduction

NetBird combines a WireGuard®-based overlay network with Zero Trust Network Access, providing a unified open source platform for reliable and secure connectivity

This sounds amazing and you can use it for remote access to your home network or to connect multiple routers and other clients (phone/PC/Mac etc.) but you are using a commercial third party and although it is advertised as free and it is to some extent, they do have an incentive to pull you into a paid tier.

Usually you can do the same setting up your own WireGuard server and clients.

[WireGuard Server Setup Guide](#)

[WireGuard Client Setup Guide](#)

But this only works if you have at least a public IP address on one side of the connection.

If you are behind CGNAT, so do not have a public IPv4 address and do not have a public IPv6 address (check with: *ifstatus wan6*) or using IPv6 is not applicable then you have to involve a commercial third party as man-in-the-middle

This can be a VPN provider which supports port forwarding (e.g. ProtonVPN), or you can rent a Virtual Private Server ( I have an Oracle VPS which can be had for free, see at the bottom of this guide), or use things like [Netbird](#), [Zerotier](#), [Cloudflared](#), [Tailscale](#) or [ngrok](#) and there are more.

I favour Netbird because it is open source and has some [advantages](#) over Tailscale, but all things mentioned will get the job done, so consider using Netbird my personal choice

## Table of Contents

Introduction.....	1
Make a free account on Netbird.....	2
Install Netbird on OpenWRT router.....	4
Netbird log.....	4
Firewall setup:.....	4
Allow SSH access from Dashboard.....	6
Create routing rules.....	8
Install on Oracle VPS with Ubuntu (24.04).....	11
Setup Oracle free OpenVPN cloud server.....	12

Start with viewing: <https://docs.netbird.io/how-to/getting-started>  
All the docs can be found at: <https://docs.netbird.io/>

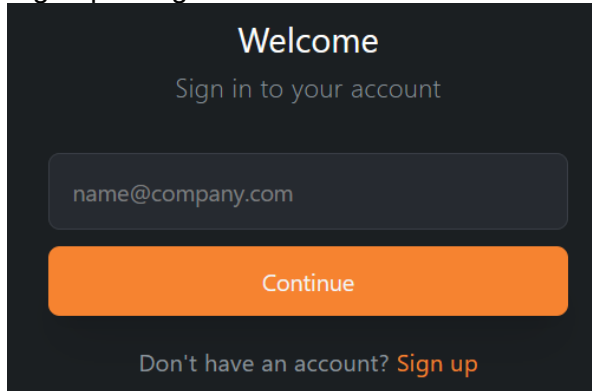
## Make a free account on Netbird

go to: <http://netbird.io>

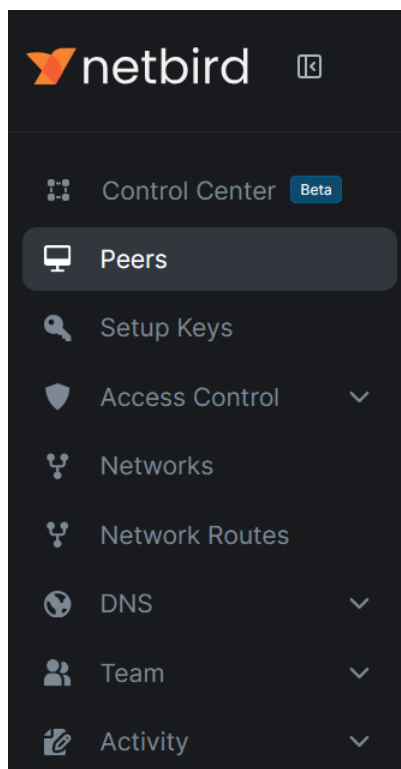
Click:



Sign up or login:

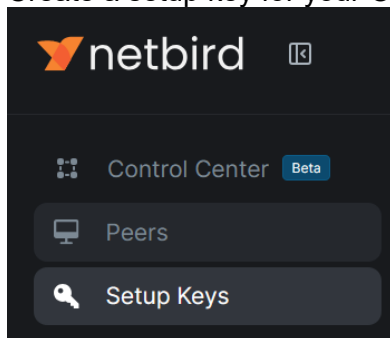
A dark-themed login and signup form. At the top, it says "Welcome" in white, followed by "Sign in to your account" in a lighter gray. Below this is a text input field with the placeholder "name@company.com". Underneath the input field is a large orange button labeled "Continue". At the bottom, there is a link that says "Don't have an account? Sign up", where "Sign up" is in orange and the rest is in white.

Now you are connected to your Netbird Dashboard the central administration:

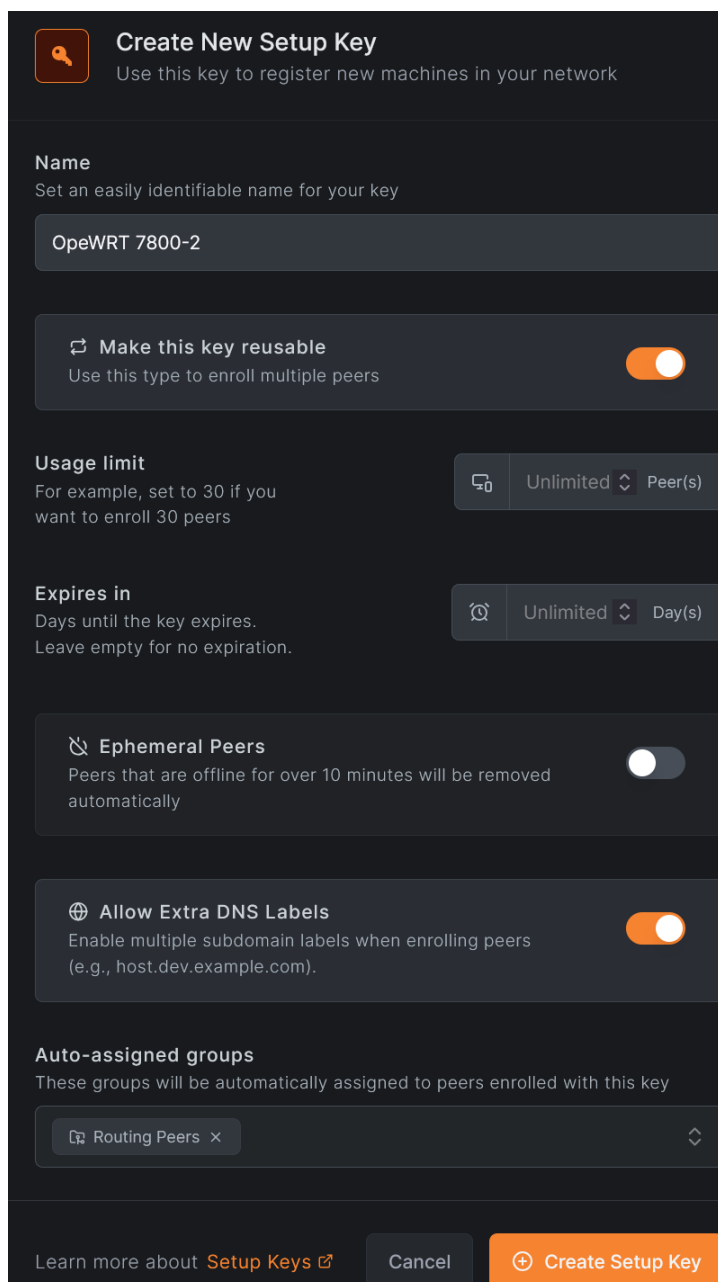


Next step is to create a setup key for your OpenWRT router

Create a setup key for your OpenWRT router, in your Netbird Dashboard click *Setup Keys*:



Fill in the name of your router and change the other items, shown are my settings, when done Click *Create Setup Key*.

A screenshot of the 'Create New Setup Key' form in the Netbird dashboard. The form has a title 'Create New Setup Key' and a subtitle 'Use this key to register new machines in your network'. It contains several sections: 'Name' with a text input field containing 'OpeWRT 7800-2'; 'Make this key reusable' with a toggle switch turned on; 'Usage limit' with a dropdown menu set to 'Unlimited' and 'Peer(s)'; 'Expires in' with a dropdown menu set to 'Unlimited' and 'Day(s)'; 'Ephemeral Peers' with a toggle switch turned off; 'Allow Extra DNS Labels' with a toggle switch turned on; and 'Auto-assigned groups' with a dropdown menu showing 'Routing Peers'. At the bottom, there is a link to 'Learn more about Setup Keys', a 'Cancel' button, and a 'Create Setup Key' button.

Copy and store the setup key

## Install Netbird on OpenWRT router

For opkg:  
opkg update  
opkg install netbird

or for apk:  
apk update  
apk add netbird

Netbird is a rather large package around 20 MB written in Go so make sure your storage is sufficient

The netbird executable is stored in /usr/share/netbird.  
The service is called from /etc/init.d/netbird

When installed you can setup with:

```
netbird up --setup-key <key from previous step>
```

After some time you will see:

```
root@R7800-2:~# netbird up --setup-key E20033F4-0C99-470E-A27A-5F066D8590EA
```

**Connected**

```
root@R7800-2:~#
```

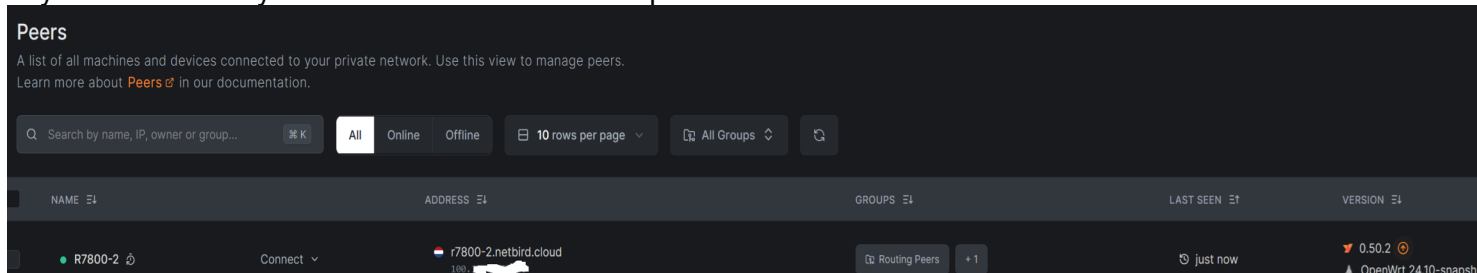
You can use **netbird help** to see the available commands e.g.:

netbird up/down/status etc

but using e.g.:

service netbird status/stop/start etc. will also work (for complete list: service netbird )

In your Dashboard you can now see the installed peer



with ifconfig or ip address show on the router, you should see the new interface (device) **wt0**

If not reboot the router and check netbird status: *netbird status*

## Netbird log:

Showing netbird log:

```
cat /tmp/log/netbird/client.log
```

## Firewall setup:

Luci > Network > scroll down and Add new interface:

Name e.g.: netbird1

Protocol: Unmanaged

Device: scroll down and choose wt0

## Interfaces » netbird1

General Settings Advanced Settings Firewall Settings DHCP Server

Status

Device: wt0  
Uptime: 0h 0m 9s  
RX: 0 B (0 Pkts.)  
TX: 0 B (0 Pkts.)

Protocol

Unmanaged ▼

Device

wt0 ▼

Disable this interface

☐

Bring up on boot

☒

/etc/config/network:

```
config interface 'netbird1'
    option proto 'none'
    option device 'wt0'
```

Head over to Firewall settings and add to LAN zone:

## Interfaces » Netbird1

General Settings Advanced Settings Firewall Settings DHCP Server

Create / Assign firewall-zone

lan lan: wg\_stos\_6: (empty) ▼

/etc/config/firewall:

```
config zone
    option name 'lan'
    option input 'ACCEPT'
    option output 'ACCEPT'
    option forward 'ACCEPT'
    list network 'lan'
    list network 'netbird1'
```

In the end reboot the router or do service network restart, service firewall restart and service netbird restart.

Check with ifconfig (ip a) and ip route that the interface (wt0) and route are present:

root@DL-WRX36:~# **ip address show wt0**

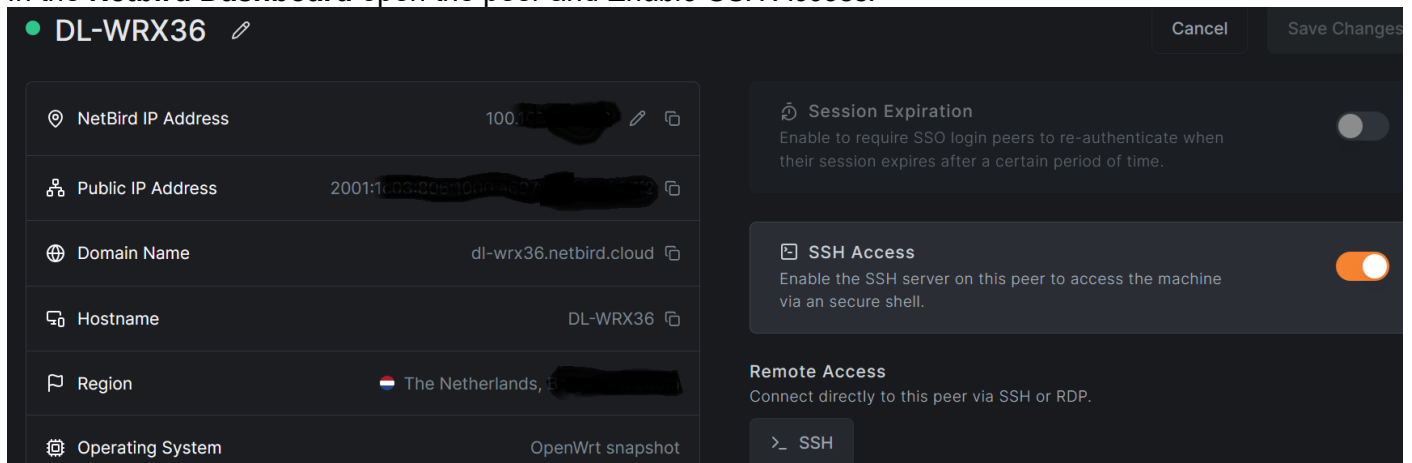
```
31: wt0: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1280 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/none
    inet 100.105.224.116/16 brd 100.105.255.255 scope global wt0
        valid_lft forever preferred_lft forever
```

root@DL-WRX36:~# **ip route**

```
default via 192.168.0.1 dev wan proto static src 192.168.0.9
100.105.0.0/16 dev wt0 proto kernel scope link src 100.105.224.116
```

## Allow SSH access from Dashboard

In the **Netbird Dashboard** open the peer and *Enable SSH Access*:



### On the router (client)

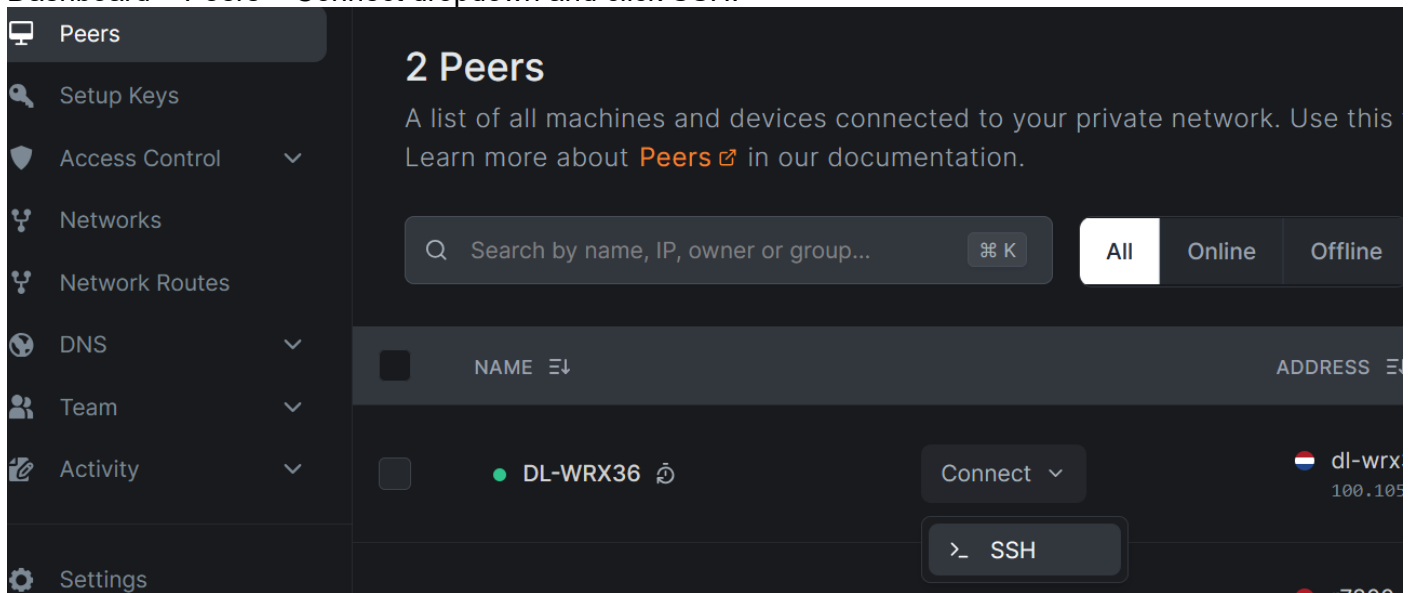
Make sure SSH is allowed (<https://github.com/netbirdio/netbird/issues/2632>):

```
netbird down
```


```
netbird up --allow-server-ssh
```

From your Netbird dashboard you should now be able to SSH into your router:

Dashboard > Peers > Connect dropdown and click SSH:



Connect with the default port 44338 to the in netbird included SSH server:




DL-WRX36

Connect to 100.105.224.116 via SSH


Username & Port

The username and port you will use to connect to the remote host.




root

@100.105.224.116



44338



Learn more about [SSH](#)

Cancel

Connect

## Create routing rules

See: <https://docs.netbird.io/how-to/routing-traffic-to-private-networks>

Note for routing between your peers it is imperative that all involved subnets are unique!

My DL-WRX36 has subnet 192.168.9.0/24.

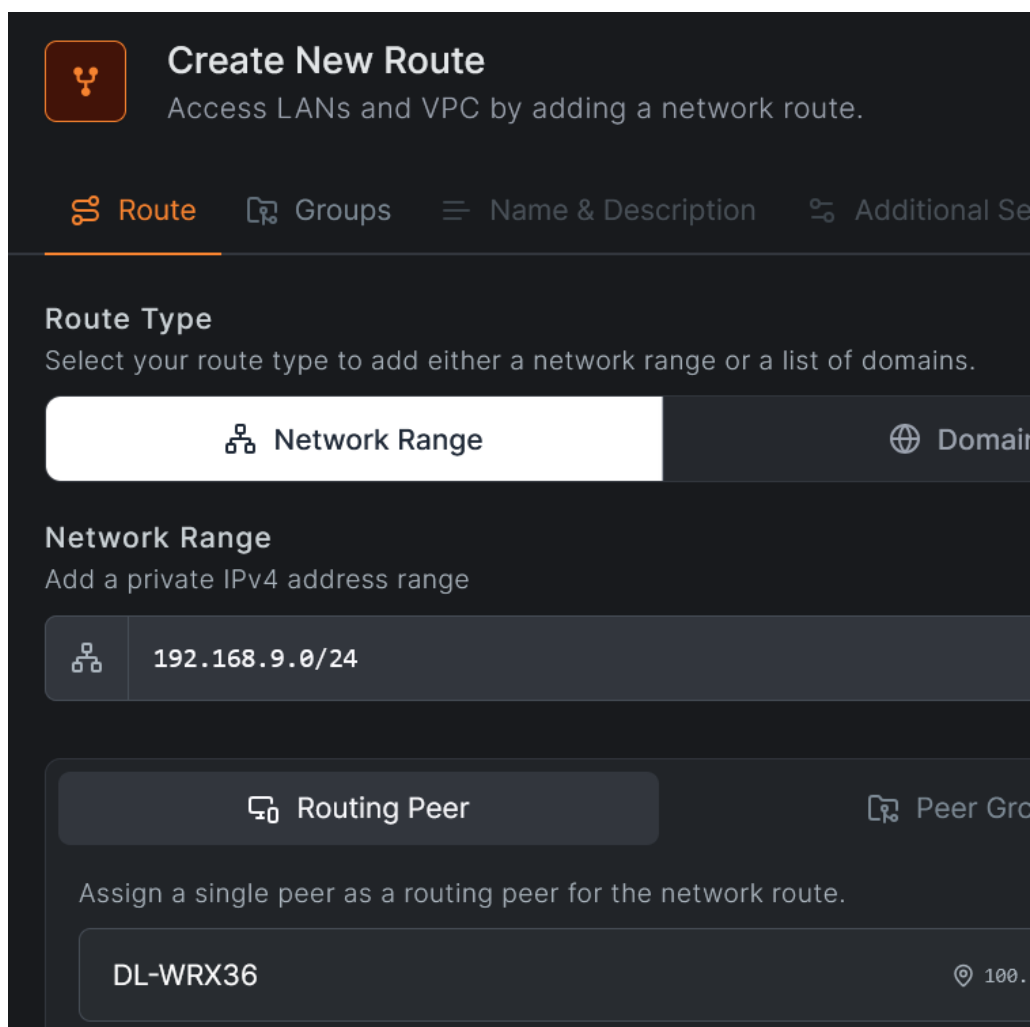
I will create a routing rule to create a route for this 192.168.9.0/24 subnet to my DL-WRX36 and push that route to all peers.

Those pushed routes are pushed to an alternate routing table on all peers, this table is usually called netbird.

Lets go:

Netbird Dashboard > Network Routes > Add Route

Add the network range to my DL-WRX36:



The screenshot shows the 'Create New Route' interface in the Netbird dashboard. At the top, there's a title 'Create New Route' with a subtitle 'Access LANs and VPC by adding a network route.' Below this is a navigation bar with tabs: 'Route' (selected), 'Groups', 'Name & Description', and 'Additional Settings'. The 'Route Type' section has two options: 'Network Range' (selected) and 'Domain'. Under 'Network Range', there's a text input field containing '192.168.9.0/24'. The 'Routing Peer' section has a dropdown menu with 'DL-WRX36' selected. The interface is dark-themed with orange accents.

**Create New Route**  
Access LANs and VPC by adding a network route.

**Route** Groups Name & Description Additional Settings

**Route Type**  
Select your route type to add either a network range or a list of domains.

Network Range Domain

**Network Range**  
Add a private IPv4 address range

192.168.9.0/24


**Routing Peer** Peer Group

Assign a single peer as a routing peer for the network route.

DL-WRX36 100.0.0.0/24







Advertise this route to all my peers:




## Create New Route

Access LANs and VPC by adding a network route.

 Route  **Groups**  Name & Description  Additional Settings

### Distribution Groups


Advertise this route to peers that belong to the following groups

 Routing Peers ×

### Access Control Groups (optional)

These groups allow you to limit access to this route. Simply use these groups as a destination when creating access policies.


Add or select group(s)...

[Learn more about Network Routes](#) 

Back





Continue

Name and description:



## Create New Route

Access LANs and VPC by adding a network route.

 Route  Groups  **Name & Description**  Additional Settings

### Network Identifier

Add a unique network identifier that is assigned to each device.


DL-WRX36

### Description (optional)

Write a short description to add more context to this route.

Route to DL-WRX36 192.168.9.0/24 subnet

Additional settings:



## Create New Route


Access LANs and VPC by adding a network route.

Route


Groups


Name & Description

**Additional Settings**


 **Enable Route**

Use this switch to enable or disable the route.




 **Masquerade**

Allow access to your private networks without configuring routes on your local routers or other devices.




**Metric**

A lower metric indicates higher



9999



You might need to restart netbird on all peers

On my Oracle VPS I can now see the rules and the alternate routing table created by netbird:

```
ubuntu@vps-egc:~$ ip rule show
0:    from all lookup local
105:  from all lookup main suppress_prefixlength 0
110:  not from all fwmark 0x1bd00 lookup netbird
32766: from all lookup main
32767: from all lookup default
ubuntu@vps-egc:~$
```

```
ubuntu@vps-egc:~$ ip route sho table netbird
192.168.9.0/24 dev wt0
ubuntu@vps-egc:~$
```

So from my oracle VPS there now is a route to my DL-WRX36 subnet

## Install on Oracle VPS with Ubuntu (24.04)

```
sudo apt-get update
sudo apt install ca-certificates curl gnupg -y
curl -sSL https://pkgs.netbird.io/debian/public.key | sudo gpg --dearmor --output /usr/share/keyrings/netbird-
archive-keyring.gpg
echo 'deb [signed-by=/usr/share/keyrings/netbird-archive-keyring.gpg] https://pkgs.netbird.io/debian stable
main' | sudo tee /etc/apt/sources.list.d/netbird.list
```

```
sudo apt-get update
sudo apt-get install netbird
# only for the GUI
#sudo apt-get install netbird-ui
```

```
netbird up --setup-key <setup-key made on dashboard> --allow-server-ssh
```

Log on Ubuntu: `cat /var/log/netbird/client.log`

SSH access note that the user name is usually: *ubuntu*

For (SSH) Access add thes firewall rules

```
sudo iptables -I INPUT 3 -p udp --dport 3478 -j ACCEPT # NetBird TURN
sudo iptables -I INPUT 4 -p tcp --dport 44338 -j ACCEPT # SSH service port from netbird

sudo iptables -I INPUT 5 -p udp --dport 51820 -j ACCEPT # NetBird WireGuard
#sudo iptables -t nat -I POSTROUTING -o ens3 -j MASQUERADE #To Masquerade traffic
```

Make persistent:  
`sudo netfilter-persistent save`

vcn-XXX > Security > Default Security List for vcn-XXX > Security rules:

<input type="checkbox"/>	No	0.0.0.0/0	UDP	All	3478
<input type="checkbox"/>	No	0.0.0.0/0	TCP	All	44338
<input type="checkbox"/>	No	0.0.0.0/0	UDP	All	51820

## Setup Oracle free OpenVPN cloud server

<https://www.youtube.com/watch?v=E-CLtExRzX8>

<https://mateo.cogeanu.com/2020/wireguard-vpn-pihole-on-free-oracle-cloud/>