

# OpenWRT Netbird

version 8

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, to connect multiple routers and other clients (phone/PC/Mac etc.) and when setup as exit node as a remote VPN 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, besides they know your clients and routes but the traffic of course is still encrypted via the WireGuard encryption.

Usually you can do the same by 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 also 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 favor Netbird because it is open source and has some [advantages](#) over Tailscale, but all things mentioned will get the job done, using Netbird is just my personal choice.

## Table of Contents

Introduction.....	1
Make a free account on Netbird.....	2
Install Netbird on OpenWRT router.....	4
Netbird log.....	4
Network setup.....	5
Firewall setup.....	5
Allow SSH access from Dashboard.....	7
Create routing rules.....	9
Create Exit node.....	12
Install on Oracle VPS with Ubuntu (24.04).....	15
Setup Oracle free OpenVPN cloud server.....	16

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

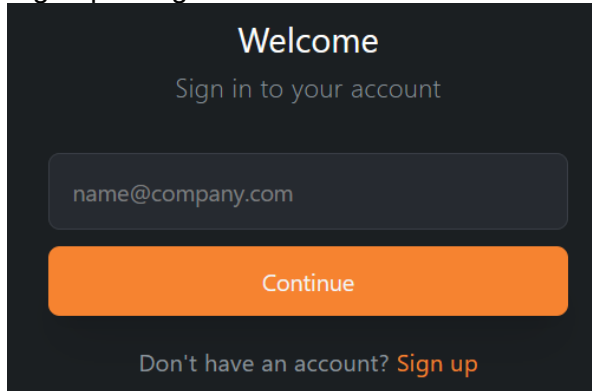
## Make a free account on Netbird

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

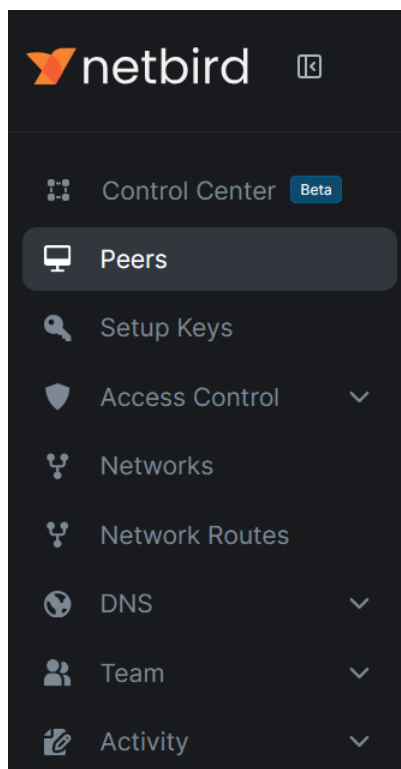
Click:



Sign up or login:

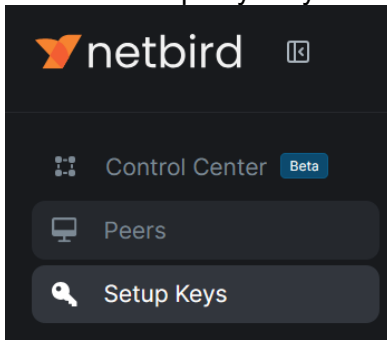
A dark-themed login screen. At the top, it says "Welcome" in white, followed by "Sign in to your account" in a lighter gray. Below this is a text input field containing the placeholder "name@company.com". Under the input field is a large orange button labeled "Continue". At the bottom, it says "Don't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

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*.

 **Create New Setup Key**  
Use this key to register new machines in your network


**Name**  
Set an easily identifiable name for your key


 **Make this key reusable**

Use this type to enroll multiple peers


☒


**Usage limit**  
For example, set to 30 if you want to enroll 30 peers




**Expires in**  
Days until the key expires.  
Leave empty for no expiration.




 **Ephemeral Peers**

Peers that are offline for over 10 minutes will be removed automatically


☐



 **Allow Extra DNS Labels**

Enable multiple subdomain labels when enrolling peers  
(e.g., host.dev.example.com).

☒

**Auto-assigned groups**  
These groups will be automatically assigned to peers enrolled with this key

 Routing Peers

Learn more about [Setup Keys](#)

Cancel

Create Setup Key

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:~#
```

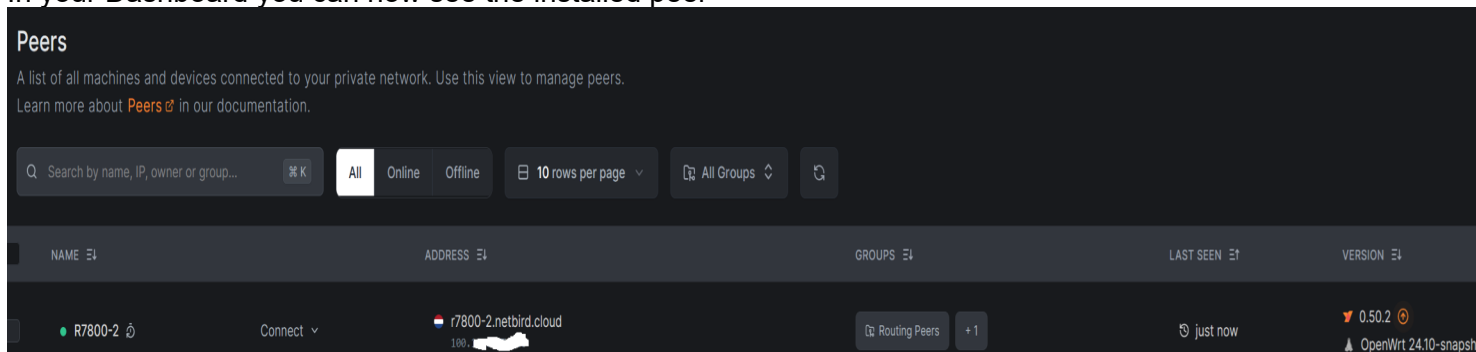
You can use **netbird help** to see the available commands e.g.:

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



NAME	ADDRESS	GROUPS	LAST SEEN	VERSION
R7800-2	r7800-2.netbird.cloud	Routing Peers	just now	0.50.2

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
```

## Network setup

Create a new unmanaged interface via LuCI: **Network > Interfaces > Add new interface**

- Name: **netbird1**
- Protocol: **Unmanaged**
- Device: **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'
```

## Firewall setup

Create a new firewall zone via LuCI: **Network → Firewall → Zones → Add**

- Name: **netbird**
- Input: **ACCEPT** (default)
- Output: **ACCEPT** (default)
- Forward: **ACCEPT**
- Masquerading: **on**
- MSS Clamping: **on**
- Covered networks: **netbird1**
- Allow forward to destination zones: Select your **LAN** (and/or other internal zones or WAN if you plan on using this device as an exit node), as this is na exit node **WAN** is slected
- Allow forward from source zones: Select your **LAN** (and/or other internal zones or leave it blank if you do not want to route LAN traffic to other tailscale hosts)

Click **Save & Apply**

## Firewall - Zone Settings

### General Settings Advanced Settings Conntrack Settings

This section defines common properties of "netbird". The *input* and *output* options set the default policies for traffic entering and leaving the zone. The *input* option describes the policy for forwarded traffic between different networks within the zone. *Covered networks* specifies which networks are covered by this zone.

Name	<input type="text" value="netbird"/>
Input	<input type="text" value="accept"/>
Output	<input type="text" value="accept"/>
Intra zone forward	<input type="text" value="accept"/>
IPv4 Masquerading	<input checked="" type="checkbox"/> Enable network address and port translation IPv4 (NAT4) or typically enabled on the wan zone.
MSS clamping	<input checked="" type="checkbox"/>
Covered networks	<input type="text" value="netbird1"/>

The options below control the forwarding policies between this zone (netbird) and other zones. *Destination zones* cover for which *Source zones* match forwarded traffic from other zones **targeted at netbird**. The forwarding rule is *unidirectional*, e.g. a rule allowing to forward from wan to lan as well.

Allow forward to *destination* zones:

lan	lan:	wg_stos_6: (empty)	wan	wan:	wan6:
-----	------	--------------------	-----	------	-------

Allow forward from *source* zones:

lan	lan:	wg_stos_6: (empty)
-----	------	--------------------

/etc/config/firewall:

config zone

```
option name 'netbird'
option input 'ACCEPT'
option output 'ACCEPT'
option forward 'ACCEPT'
option masq '1'
option mtu_fix '1'
list network 'netbird1'
```

config forwarding

```
option src 'netbird'
option dest 'lan'
```

config forwarding

```
option src 'lan'
option dest 'netbird'
```

# As this is an exit node traffic from netbird to wan is allowed

config forwarding

```
option src 'netbird'
option dest 'wan'
```

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:

```

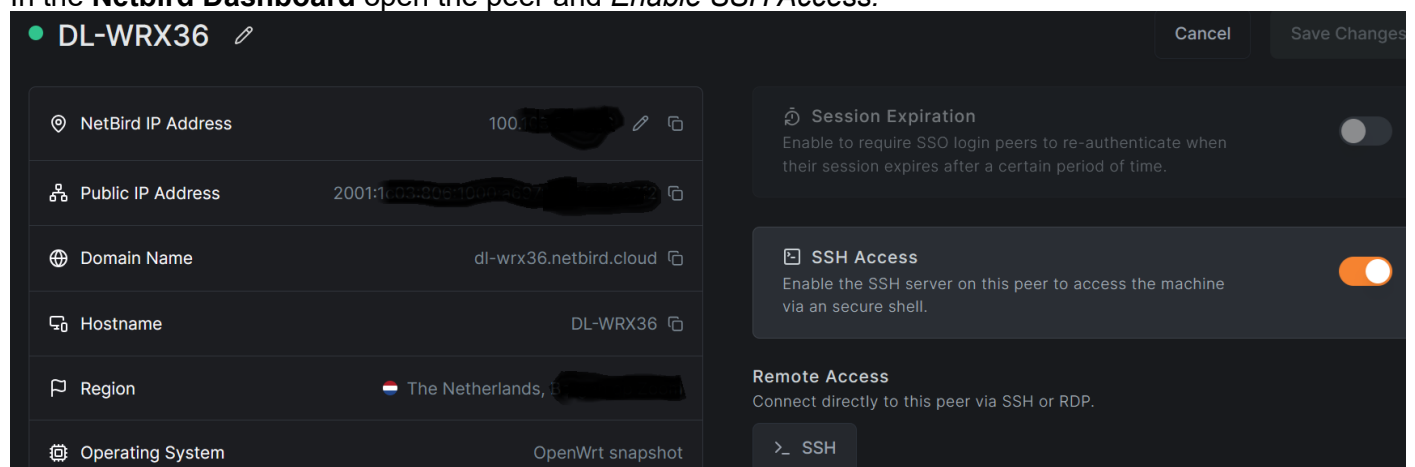
root@DL-WRX36:~# ip address show wt0
31: wt0: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 1280 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/none
    inet 100.105.224.116/16 brd 100.105.255.255 scope global wt0
        valid_lft forever preferred_lft forever

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

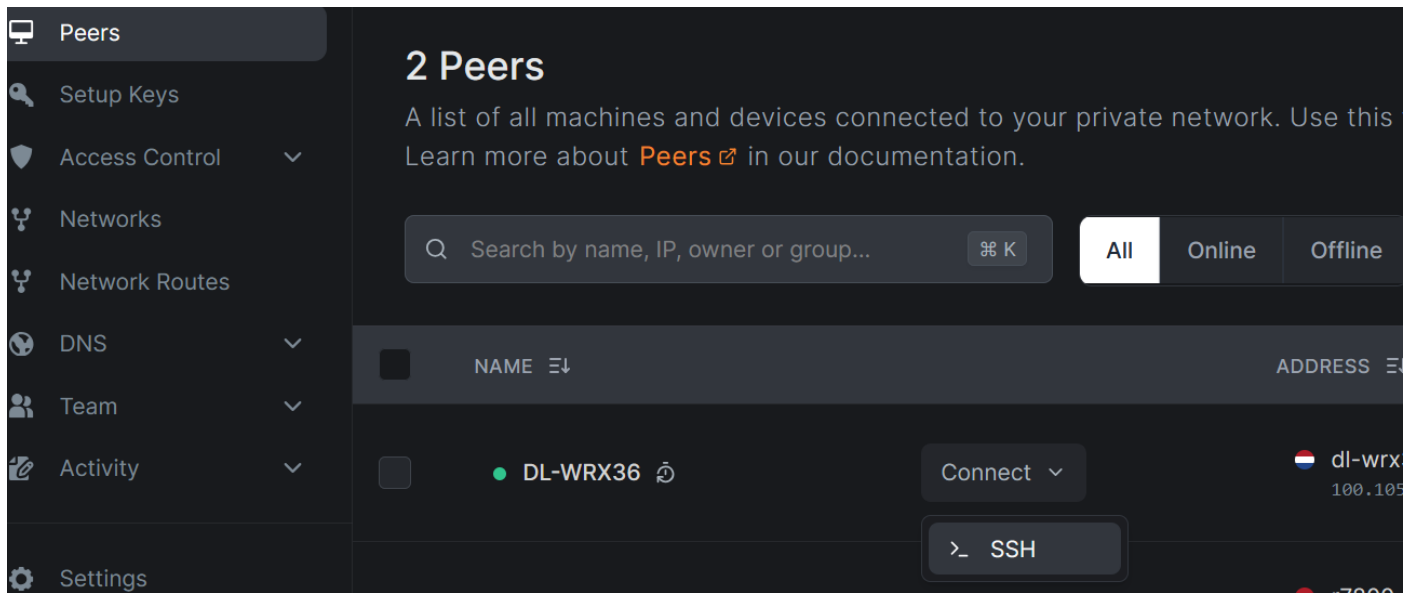
Make sure SSH is allowed (<https://github.com/netbirdio/netbird/issues/2632>):

```
netbird down
```

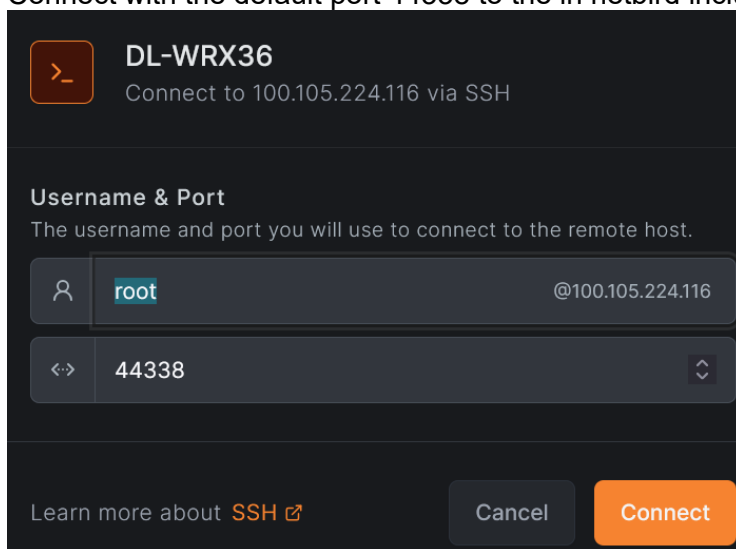
```
netbird up --allow-server-ssh
```

On 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:





## 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.

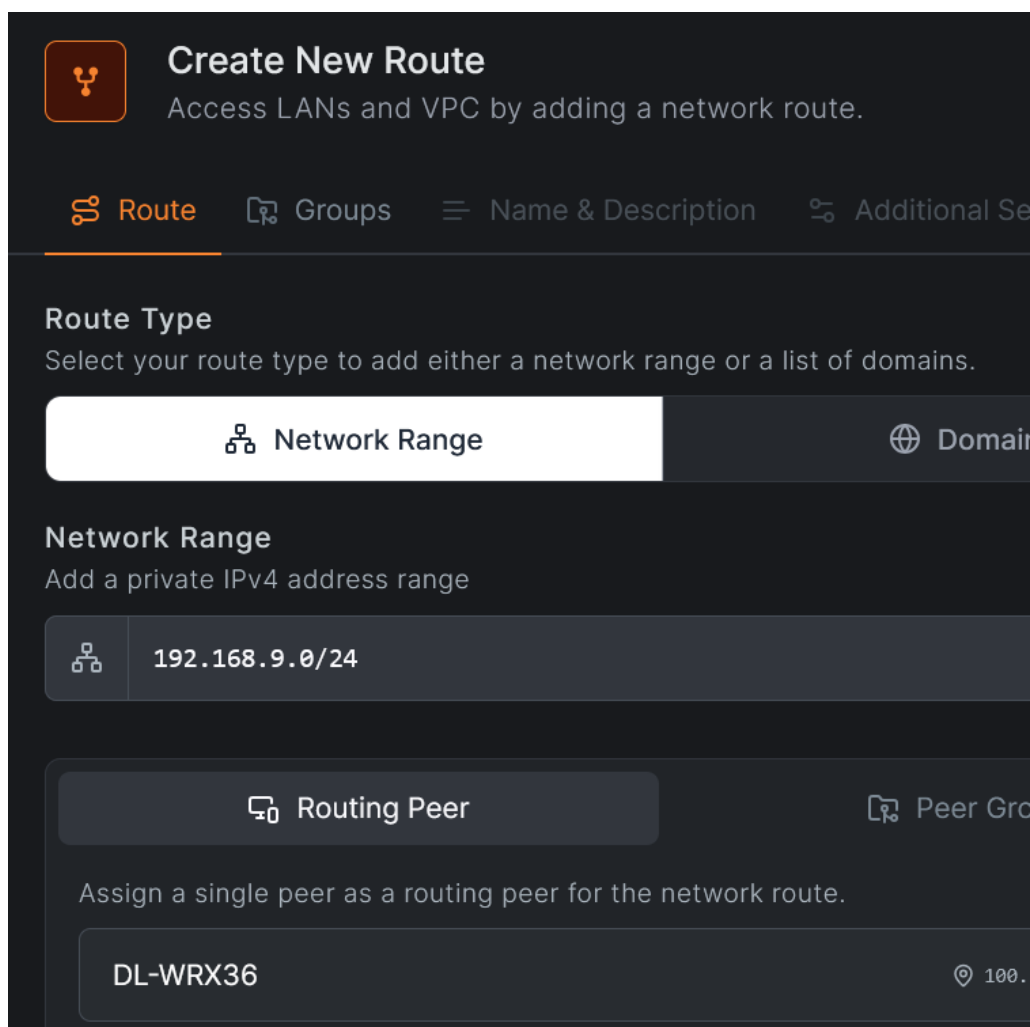
I will create a routing rule to create a route for this 192.168.9.0/24 subnet to my DL-WRX36 and push that route to all peers.

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 tabbed interface with 'Route' selected. The 'Route Type' section has two options: 'Network Range' (selected) and 'Domain'. The 'Network Range' section prompts to 'Add a private IPv4 address range' and shows a text input field containing '192.168.9.0/24'. The 'Routing Peer' section prompts to 'Assign a single peer as a routing peer for the network route.' and shows a dropdown menu with 'DL-WRX36' selected.



### 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

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

## Create Exit node

An exit node is a peer which acts as a VPN server other designated peers route all their traffic via the exit node.

On the exit node it is important that the firewall allows forwarding from **netbird** to **wan**, see paragraph about [firewall](#).

Netbird documentation: <https://netbird.io/knowledge-hub/netbird-network-routes>, scroll down to the bottom.

Login in the Netbird dashboard

**Peers** > Click on the peer you want to be the exit node > On the overview page scroll to the bottom and click **Setup Exit node**

DL-WRX36

NetBird IP Address100.105.224.116

Public IP Address2001:1c03:806:1000:a697:33ff:fedf:97f2

Domain Namedl-wrx36.netbird.cloud

Hostnamedl-wrx36

RegionThe Netherlands, Bergen op Zoom

Operating SystemOpenWrt snapshot

Registered on8 October, 2025 at 6:53 PM (2 days ago)

Last seenjust now

Agent Version0.58.2

Session Expiration

SSH Access

Remote Access

Assigned Groups

Network Routes

Accessible Peers

Traffic Events

Network Routes

Access other networks without installing NetBird on every resource.

NAME

NETWORK

DISTRIBUTION GROUPS

ACTIVE

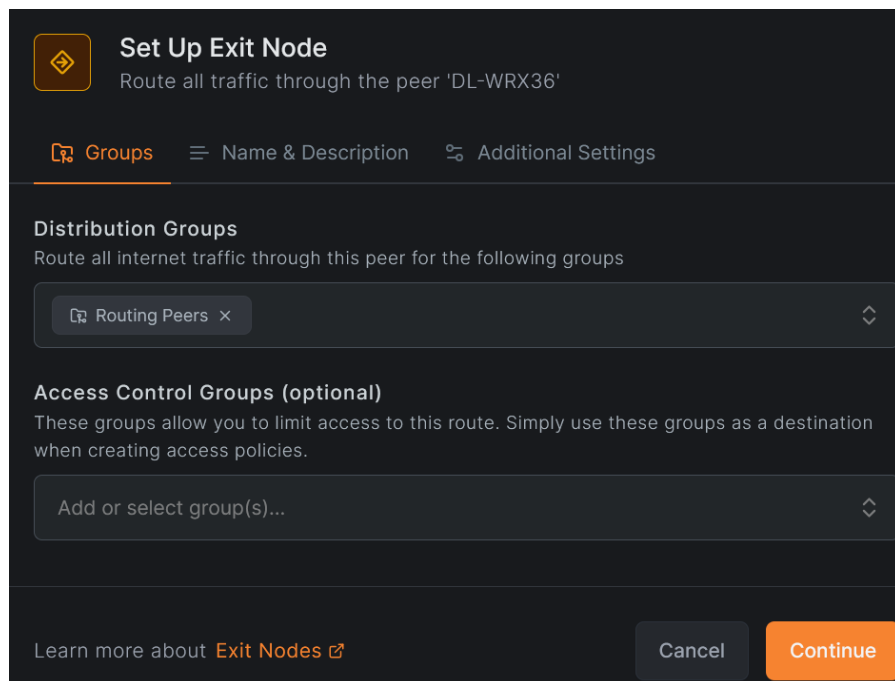
DL-WRX36

192.168.9.0/24

Routing Peers

Delete

Under **Groups** add the peers you want to use the exit node, I had created a group **Routing Peers** and I want all those peers to use this router as exit node



**Set Up Exit Node**  
Route all traffic through the peer 'DL-WRX36'

**Groups** | Name & Description | Additional Settings

**Distribution Groups**  
Route all internet traffic through this peer for the following groups

Routing Peers x

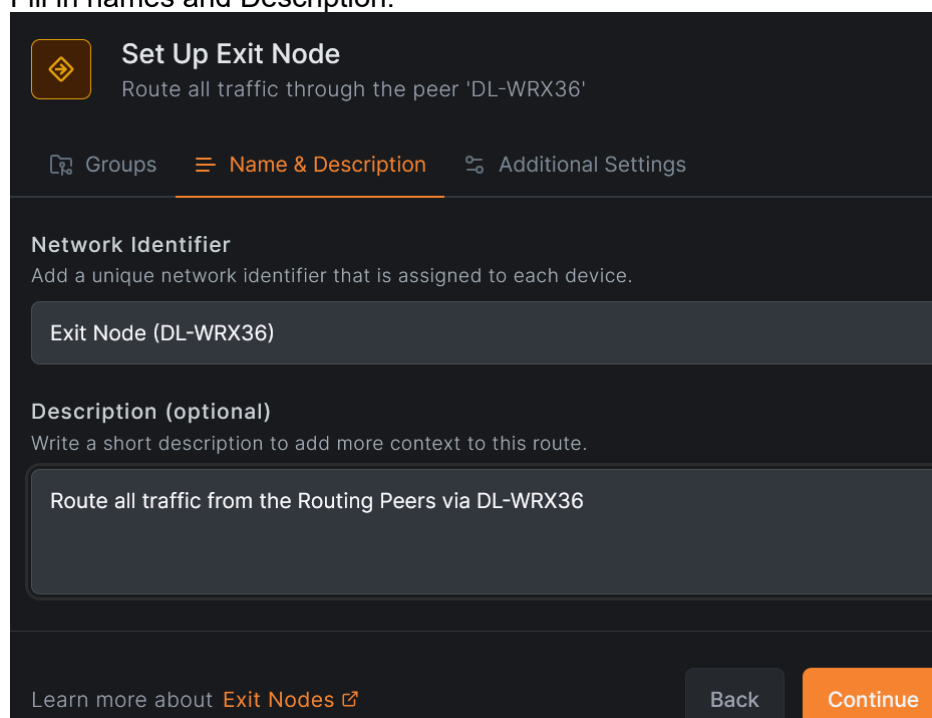
**Access Control Groups (optional)**  
These groups allow you to limit access to this route. Simply use these groups as a destination when creating access policies.

Add or select group(s)...

Learn more about [Exit Nodes](#) | Cancel | Continue

## Continue

Fill in names and Description:



**Set Up Exit Node**  
Route all traffic through the peer 'DL-WRX36'

Groups | **Name & Description** | Additional Settings

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

Exit Node (DL-WRX36)


**Description (optional)**  
Write a short description to add more context to this route.

Route all traffic from the Routing Peers via DL-WRX36

Learn more about [Exit Nodes](#) | Back | Continue

## Continue


Enable Route and Auto Apply Route




## Set Up Exit Node

Route all traffic through the peer 'DL-WRX36'

[Groups](#)
[Name & Description](#)
[Additional Settings](#)


**Enable Route**


Use this switch to enable or disable the route.


**Auto Apply Route**


Automatically apply this exit node to your distribution groups. This requires NetBird client v0.55.0 or higher.

**Metric**

A lower metric indicates higher priority routes.



9999



[Learn more about Exit Nodes](#)
[Back](#)
[Add Exit Node](#)

## Add Exit Node






My DL-WRX36 is running Snapshot with Netbird 0.58 (you can see it on the overview page if you click on the Peer) so all routes are applied automatically.

My DL-WRX36 now has set a route to its own subnet (which is 192.168.9.0/24), pushed to all the Routing peers en an Exit node which pushes a default route to all the routing peers.

[Network Routes](#)
[Accessible Peers](#)
[Traffic Events](#)

## 2 Network Routes

Access other networks without installing NetBird on every resource.

NAME	NETWORK	DISTRIBUTION GROUPS	ACTIVE
 Exit Node (DL-WRX36)	 Exit Node	 Routing Peers	<div></div>
 DL-WRX36	192.168.9.0/24	 Routing Peers	<div></div>

You can check on one of the other routing peers e.g. my R7800-2 where you can see the pushed default route and the pushed route to reach the DL-WRX36:

```
root@R7800-2:~# ip route show table netbird
default dev wt0
192.168.9.0/24 dev wt0
root@R7800-2:~#
```

Now all traffic from the R7800-2 (and all its clients are routed) via Netbird, Netbird internally routes this traffic to the exit node.

## 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/>