Installing the Baseic Building Blocks

- Install our IdP and VPN

Install our IdP and VPN

https://www.youtube.com/embed/Vi2UqOFqDGU

I have extensive articles on installing Authentik, Netbird, NGinX Proxy Manager, Docker, and Docker Compose. So this article is really dedicated to the configuration of Netbird to use Authentik as our IdP (Identity Provider).

What You'll Need

- A few VPS, or VMs on which to host these services.
- Docker and Docker Compose installed on each server (I have a script for that)
- A domain name for which you can create sub-domains, and set A Records in DNS
- About an hour of time.

Setup your VMs or VPS

It will be extremely important that you setup your VPS / VMs appropriately for the workload you'll be putting on them. Also, the ability to sscale those up over time will be important as well. In the video, I use Digital Ocean for this purpose, but you are welcome to use any provider you wish, or your own hardware of course.

> Keep in mind: Your clients will be depending on you to keep your, and their, services up and running at all times. If you use your own hardware in side your home / business, make sure you have redundancy in mind for power, connectivity, and so on.

Installation of Docker and Docker Compose via a Simple Script

You should setup your first VM / VPS for NGinX Proxy Manager (NPM from this point on) / (if you don't already have it). Make sure that you have ports 80 and 443 accessible. Initially, you'll also need port 81, as this is where the NPM admin user interface runs. We'll create a proxy to this port, then close this port afterward (see my video for this procedure). The server should have 1 vCPU and 512 MB RAM.

You can easily install Docker-CE, Docker-Compose, Portainer-CE, and NGinX Proxy manager by using this quick install script I created and maintain on Github. Just use the command:

`wget https://gitlab.com/bmcgonag/docker_installs/-/raw/main/install_docker_nproxyman.sh -O install-docker.sh`

To download the script to your desired host.

Change the permissions to make the script executable:

`chmod +x ./install-docker.sh`

and then run the script with the command:

`./install-docker.sh`

When run, the script will prompt you to select your host operating system, then will ask you which bits of software you want to install.

Simply enter 'y' for each thing you want to install.

At some point, you may be asked for your super user (sudo) password as well.

Allow the script to complete installation.

At this point, you might want to log out and back in, as this will allow you to use the `docker` and `docker-compose` commands without the need of sudo in front of them.

Install Authentik

For this server, I used a 1 vCPU and 2GB RAM VPS. This should give us decent performance, as well as some room to grow over time before needing to upsize.

For Authentik, we really just followed my original guide, and didn't have to make any specific changes.

Install Netbird with Authentik as the IdP

As always, you should really check the Netbird official documentation to ensure no major changes have occurred in their install and setup steps since this article was written. When in doubt, always go to the source.

Firs,t we'll be using another VPS with 1 vCPU and 2 GB RAM for this install. Additionally, we'll be using the "advanced" guide fot eh netbird install. This guide is pretty straight forward. First giving you instructions on how to get the proper / latest version of the Netbird server available.

>

> NOTE: Netbird, by default, ships with the Zitadel IdP server as part of their "Quick install" solution. This is shown in my original Netbird video, so feel free to use that, and Zitadel as your IdP if you prefer.

We again, need to start with install Docker and Docker Compose. Once that's done, we need to get Netbird ready. This will be heavily taken from the Netbird documentation, again, make sure to check that as the source of truth.

First, let's run these lines to grab the Repository and Version we need to download.

```
export REPO="https://github.com/netbirdio/netbird/"
```

```
export LATEST_TAG=$(basename $(curl -fs -o/dev/null -w %{redirect_url} ${REPO}releases/latest))
```

To test that our commands worked, we can echo out the values of the two variables we just set:

```
echo $REPO
```

```
echo $LATEST_TAG
```

Those should each give you some output below the command. Next, we'll clone the repository from Github.

```
git clone --depth 1 --branch $LATEST_TAG $REPO
```

Now we should have a new folder called 'netbird' in the current directory. We need to move into that folder, and then into the folder inside of of called 'infrastructure_files'.

```
cd netbird/infrastructure_files
```

Here, we want to copy the 'setup.env.example' file to a new file called 'setup.env'.

```
cp setup.env.example setup.env
```

Now we can edit the new 'setup.env' file, but still have the original example file to fall back on if needed.

```
nano setup.env
```

Here's my example file. You'll want to copy / paste this to your setup.env, and then change the values as stated below the file. Part of this is setting up a Provider and Application for Netbrid in Authentik.

```
## example file, you can copy this file to setup.env and update its values
##


# Image tags
```

```
# you can force specific tags for each component; will be set to latest if empty
NETBIRD_DASHBOARD_TAG=""
NETBIRD_SIGNAL_TAG=""
NETBIRD_MANAGEMENT_TAG=""
COTURN_TAG=""

# Dashboard domain. e.g. app.mydomain.com
NETBIRD_DOMAIN="vpn.mygreatdomain.com"

# TURN server domain. e.g. turn.mydomain.com
# if not specified it will assume NETBIRD_DOMAIN
NETBIRD_TURN_DOMAIN=""

# TURN server public IP address
# required for a connection involving peers in
# the same network as the server and external peers
# usually matches the IP for the domain set in NETBIRD_TURN_DOMAIN
NETBIRD_TURN_EXTERNAL_IP="210.3.42.55"

# ----------------------------------------
# OIDC
#  e.g., https://example.eu.auth0.com/.well-known/openid-configuration
# ----------------------------------------
NETBIRD_AUTH_OIDC_CONFIGURATION_ENDPOINT="https://auth.mygreatdomain.com/application/o/netbird
/.well-known/openid-configuration"
NETBIRD_USE_AUTH0=false
NETBIRD_AUTH_CLIENT_ID="copy-this-from-your-authentik-provider-information"
NETBIRD_AUTH_SUPPORTED_SCOPES="openid profile email offline_access api"
NETBIRD_AUTH_AUDIENCE="copy-this-from-your-authentik-provider-information"
NETBIRD_AUTH_DEVICE_AUTH_CLIENT_ID="copy-this-from-your-authentik-provider-information"
NETBIRD_AUTH_DEVICE_AUTH_AUDIENCE="copy-this-from-your-authentik-provider-information"

NETBIRD_MGMT_IDP="authentik"
NETBIRD_IDP_MGMT_CLIENT_ID="copy-this-from-your-authentik-provider-information"
NETBIRD_IDP_MGMT_EXTRA_USERNAME="Netbird"
NETBIRD_IDP_MGMT_EXTRA_PASSWORD="this-is-a-different-key-that-your-authentik-instance-will-
create-for-the-netbird-service-user"

# ----------------------------------------
# Letsencrypt
```

```
# ----------------------------------------
# Disable letsencrypt
#  if disabled, cannot use HTTPS anymore and requires setting up a reverse-proxy to do it instead
NETBIRD_DISABLE_LETSENCRYPT=false
# e.g. hello@mydomain.com
NETBIRD_LETSENCRYPT_EMAIL="myself@mygreatdomain.com"
# ----------------------------------------
# Extra settings
# ----------------------------------------
# Disable anonymous metrics collection, see more information at https://netbird.io/docs/FAQ/metrics-collection
NETBIRD_DISABLE_ANONYMOUS_METRICS=false
# DNS DOMAIN configures the domain name used for peer resolution. By default it is netbird.selfhosted
NETBIRD_MGMT_DNS_DOMAIN=netbird.selfhosted
```

In the setup.env file, make sure you fill out the NETBIRD_DOMAIN with your intended domain / sub-domain name.

Additionally, you need to put in a public IPv4 address for the server where Netbird is running if you'll be relying on their coturn server which is setup automatically (I suggest this). If you want to use your own coturn server, you can, but you need to modify the resulting 'management.json' file with the appropriate details for communicating with and through your coturn server (I don't recommend this).

If you are using the Netbird created coturn server, you can leave the TURN_SERVER_DOMAIN blank (just empty double quotes ""). If, however, you are using your own coturn server, you need to fill in the domain name for that server here.

Next, you need to create a provider and application entry in Authentik for Netbird. Start with the provider.

- In Authentik go to Administrator
- Click to expand the Applications section
- Click 'Providers', and then select to create a new Provider in the Main UI
- Select OpenID/OAuth Provider and click Next.
- Fill out the form with the values
  - Name: Netbird
  - Authentication Flow: Default Authentication Flow
  - Authorization Flow: Default Authorization Flow (explicit option).
  - Under Protocol Settings select:
    - 'Public'

        - Redirect URIs: String: `https://<domain>`,Regex: `https://<domain>.*`, Strict: `http://localhost:53000`
            - Make sure to enter each of those on a separate line, without commas.
        - Signing Key: Select the Signing Key, or feel free to use the Self-signed Key.
    - Advanced protocol settings:
        - Access code validity: `minutes=10`
        - Subject mode: `Based on the User's ID`

Click 'Finish'.

Next, we need to create an Application to go with this Provider. In the left menu, under Application, click the Applications sub-option.

In the main page interface, give the Application a name of 'Netbird'. Make sure to set the slug to 'netbird', then choose 'Netbird' from the Provider drop-down selection list. Additionally, under 'UI Settings', you can add the Netbird logo, as this helps make the Authentik application a little better for you and your end users (IMO).

Click 'Create'.

Now, we need to create a service user. Click on 'Directory' in the left menu, then select 'Users' underneath it.

In the main user interface, click the button at the top that says 'Create Service Account'.

Put in 'Netbird' for the username, and then disable the option for 'Create Group'.

Click 'Create'. Next, you'll see a window with your newly create service account user, and a password generated by Authentik. Copy this username information into a secure password manager. You'll need this value in your 'setup.env' file for the value of NETBIRD_IDP_MGMT_EXTRA_USERNAME.

Next, we need to create an application password for the Netbird system account. We don't want to use the Authentik create password for this setup (anymore).

Finally, this is what all I had to dod to make this work.

1. Add the 'API' scope to the provider under `Applications > Providers`, edit the Netbird provider, then expand `Advanced Protocol Settings`, and add the scope by selecting it on the left, and moving it to the right with the single '>' button.
2. Change the Redirect URL with the wildcard `*` symbol (in my case the second one, in `Applications > Providers`, edit the netbird provider, and change it from `Strict` to `Regex`.

The above will fix the URI Redirect issue people are running into.

Next, I had to tackle the 401 Access Error.

1. Bind the Netbird System account to the Netbird application.

Go to your Admin console in Authentik. Click on `Applications` in left Menu to expand it. Select `Applications`. Click the `Netbird` application in the list to open its properties. Select the `Policy / Group / User Bindings` tab.

Click the `Bind existing Policy / Group / User` button, and then select the `User` tab on the window that opens. Select the `Netbird` user from the drop-down, and click the `Create` button.

NOTE: While you're here, you can bind any other users or groups in the same way, and skip step 3 below.

2. Add an app password for the Netbird system account.

In the left menu select `Directory > Token and App Passwords`. Click the `Create` button. Name the object whatever makes sense to you, then select the `Netbird` user. Select the `App Password` option, give it a description that makes sense to you, and un-tick the `Expiring` option, or set the expiration date out a good ways (like a year or so). Click `Create`.

Copy the password by clicking the little copy icon out to the right on the new entry in the table.

Now, paste this copied application password into the setup.env file for the NETBIRD_IDP_MGMT_EXTRA_PASSWORD.

Next, we need to add the service account we just created to the admin group in Authentik.

On the left menu, under 'Directory', select the 'Groups' option. In the 'Groups' view on the main user interface, select the 'authentik Admins' group by clicking on it. At the top of the window that opens, select the 'Users' tab. Click 'Add existing user', and click the '+' icon, and pick the 'Netbird' service account you just created.

Disable the 'Hide Service-accounts' option at the top of the view, and verify that Netbird ahs been added as an admin.

Next, we'll bind the Netbird System account to the Netbird application.

Go to your Admin console in Authentik. Click on `Applications` in left Menu to expand it. Select `Applications` under the expanded section. Click the `Netbird` application in the list to open its properties. Select the `Policy / Group / User Bindings` tab.

Click the `Bind existing Policy / Group / User` button, and then select the `User` tab on the window that opens. Select the `Netbird` user from the drop-down, and click the `Create` button.

Make sure any users you want to access Netbird are part of a Group with a bind to the Netbird application, or that each user is set with a Bing to the application.

Using the same steps as above, bind any users or groups to the Netbird application who will need to access it using their Authentik credentials.

You will hopefully now be able to access the management system again.

Whew! Done with Authentik setup now.

Go back to the Application >> Providers option in the left menu, and click the 'Netbird' title for the Netbird provider we created earlier. In the window that opens, select the 'OpenID Configuration URL' and copy the entire URL. Paste this value into your setup.env for the environment variable labeled 'NETBIRD_AUTH_OIDC_CONFIGURATION_ENDPOINT'.

Next, copy the 'Client ID' from the Authentik provider view. Paste this value in the setup.env for the following environment variables:

- NETBIRD_AUTH_CLIENT_ID
- NETBIRD_AUTH_AUDIENCE
- NETBIRD_AUTH_DEVICE_AUTH_CLIENT_ID
- NETBIRD_AUTH_DEVICE_AUTH_AUDIENCE
- NETBIRD_IDP_MGMT_CLIENT_ID

Enter your email address for the LetsEncrypt certificate section, then save the file with CTRL + O, press Enter to confirm, and CTRL + X to exit the nano editor.

Build your files to run Netbird

Now that we have filled out our setup.env file, we need to run the script that Netbird provides for creating our docker-compose.yml and two other setup files we'll need. To do this, just run the following command in the terminal:

```
./configure.sh
```

The output should be a long list of environment varialbes, and where they are being set, as well as what they are being set to.

Now, let's move into the 'artifacts' folder. This is where those files are create by the 'configure.sh' script for us.

```
cd artifacts
```

Here, you can do an `ls` and see the files that are created for us. The 'management.json' and 'turnserver.conf' files are two important ones. You don't need to modify anything in these files, but if you are having trouble getting Netbird to start, and function properly, you may need to check the turn user and password in each, to ensure they match.

Before we try to bring up our Netbird server, let's check one more thing. We want to be sure that our hostname in the /etc/hosts file is set correctly.

```
sudo nano /etc/hosts
```

This will open the 'hosts' file, and you want to look for a line that may start with the ip address 127.0.1.1. If you have this line (not to be confused with the line that has 127.0.0.1), then you need to make sure that this IP is updated to be your public IP address, and that the hostname / domain are correct for your Netbird server. In my example on the video I had

```
127.0.1.1  nb.sysmainit.com nb
```

I had to change this value to be the public IP address of my server, so it looked more like

```
203.66.43.152  nb.sysmainit.com nb
```

If you need to make an edit, save your changes with CTRL + O, then press Enter to confirm, and next use CTRL + X to exit the nano editor.

Bring Up our Netbird Containers

Now we are ready to bring up our Netbird server with our Authentik IdP.

```
docker compose up -d && docker compose logs -f
```

The command(s) above tell docker to pull down the images and start the containers for Netbird, and to show us the log output as the containers begin to run.

You can look for any ERRO or ERROR messages in the logs, but they fly by pretty quickly. Give Netbird a couple fo minutes to finish it's initial startup and setup, then navigate to the domain you setup for Nebird in your browser. You should see the initial Netbird loading screen, then be redirected to your Authentik instance to log in, and after a successful authentication be re-directed back to your Netbird admin screen to start adding clients to your VPN.

Support My Channel and Content

Support my Channel and ongoing efforts through Patreon:

https://www.patreon.com/awesomeopensource