Navidrome

- Install Navidrome and Stream Your Own Library using the Airsonic API

Install Navidrome and Stream Your Own Library using the Airsonic API

https://www.youtube.com/embed/K3DtR3g2S8Y

Subsonic / Airsonic music streaming servers are an incredible way to self host your own Google Music, iTunes, Apple Music, Spotify, Pandora, and any other streaming service out there, by using your own music, and keeping it all local and under your control. The best thing is these are fairly straightforward to setup, and there are some really great mobile apps out there for free to use to stream once your server is setup.

What you'll need

- A machine you want to use as a music server
- Docker, Docker-Compose installed on that machine
- About 20 minutes of time.

Optional needs

If you want to access your music streaming server from outside your local network (over the internet) you'll need the following, otherwise these are optional:

- a domain name / subdomain with an A record pointing to your public IP address.
- NGinX Proxy Manager installed via Docker-compose (we'll install this with Docker and Docker-compose int eh next step.)
- The ability to setup port forwards on your network router for ports 80 and 443.

Docker, Docker-Compose, NGinX Proxy Manager Setup

I've got a script that will help you install all of these amazing bits of software at once if you have CentOS, Debian, or Ubuntu 18.04 or 20.04.

You just need to go to my github page and grab the right script for your OS.

https://github.com/bmcgonag/docker_installs

Find your OS and version, click the link to view the code, then copy the script code.

Create a new file called "docker-install.sh":

```
nano docker-install.sh
```

Now paste the copied code into the file you just created with CTRL + Shift + V.

Save the file wtih CTRL + O, then press Enter to confirm. Now use CTRL + X to exit the nano editor.

Next, make the file executable with:

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

Finally, run the script with:

```
./docker-install.sh
```

Allow the script to run, and enter your sudo password if / when prompted.

> Reverse Proxy Setup further down - for outside access back to your music server.

Navidrome Setup:

1. create a new folder on your server called "navidrome".

```
mkdir navidrome
```

1. change into that new directory:

```
cd navidrome
```

1. create a new file called "docker-compose.yml"

```
nano docker-compose.yml
```

1. Copy the following into that file: You can use CTRl + Shift + V to paste into the nano editor.

```
version: "3.3"
services:
```

```
  navidrome:
    image: deluan/navidrome:latest
    ports:
      - "4533:4533"
    environment:
      ND_SCANINTERVAL: 30m
      ND_LOGLEVEL: info
      ND_BASEURL: ""
    volumes:
      - "./data:/data"
      - "/path/to/your/music/here:/music:ro"
```

NOTES:

You can change the left side of the port mapping above, as this is the host port you'll access the Navidrome web server on. If port 4533 is already in use for another application you'll need to change it. **DO NOT**, however, change the right side, as this is the port on the container, and this is where the application is expected to run.

You also need to change the `path/to/your/music/here` portion to be the actual file / folder path to your music collection.

For instance, my music collection path is:

`/mnt/video/arr/data/media/music`

where /music is the directory in which all of my music resides.

Start the container with the command

`docker-compose up -d`

Allow the image to download, and the container to spin up, and you can go to your site at your server's IP address and port 4533. For instance, I went to `http://192.168.7.125:4533`

Once there, you'll see the option to create an administrator account.

If you want to have access to your collection over the internet, then we have a bit more to do.

Setup Internet Access with NGinX Proxy Manager

Login to NGinx Proxy Manager at the IP of your server with port 81.

For example, I go to `http://192.168.7.125:81`

NOTE: NGinX Proxy Manager must be allowed to use the host ports 80, 443, and 81.

NOTE 2: You should forward the ports 80, and 443 only through your router configuration to the local (LAN) IP address of your server running NGinX Proxy Manager. This makes all traffic on 80 and 443 go to NGinX Proxy Manager, then we let NGinX Proxy Manager handle directing traffic around your network to the applications you specify, vs. opening a bunch of ports to the internet which is arguably less secure.

Now, go to the IP address and port 81 of your NGinX Proxy Manager server, and use the login credentials:

username: admin@example.com
password: changeme

Upon logging in the first time, you'll be prompted to change the Admin account details, as well as the admin password. You should absolutely make these different from the defaults.

Now, log out, and back in with your new credentials to make sure it's all working.

Finally, let's create a proxy host to our Navidrome instance. This allows us to give our Navidrome install a proper sub-domain / domain name that we can access from outside of our network.

Inside NGinX Proxy Manager click the Proxy tab. Click to add a new proxy, and in the pop-up window enter the domain / sub-domain you setup with the A-record. Press tab, or Enter to accept the name and make it a "chip".

Now go to the IP address field. Here we need the IP of our Navidrome gateway in docker. (Alternatively, if you setup a docker network, and assign both NGinX Proxy Manager and Navidrome to the same network, you can simply use the Navidrome container name in this field.)

NOTE: If you are running Navidrome on a different physical server than your NGinX Proxy Manager, you'll simply use the LAN IP of the server host you are running Navidrome on.

To find the gateway IP in Docker, enter the following in the terminal:

```
docker network list
```

This will list all of the docker networks. You are likely just using the default network, but make sure.

If you don't see a network specifically created for Navidrome (either by you, or by docker), then simply do the following for the default network:

```
docker network inspect <network name>
```

Where you replace < network name > above with the actual name of your default docker network or the network created for navidrome in Docker.

You should see a bunch of JSON output in the terminal. Just scroll up, searching for "gateway". Get that IP (Usually 172.17.0.1) and put it in the field in the proxy host window for IP address.

Now, move to the port field and put in 4533, unless you changed this in the docker-compose file. If you did, then please put that port number here.

Enable the "Block Common Exploits", and "Webrtc" options, then save.

You should see your new proxy entry listed in a row. Click the url in the row to open it in a new tab, and make sure you are taken to your navidrome install.

Once you see the Navidrome login, close the browser tab, and go back to NGinX Proxy Manager.

Next, we want SSL.

Click on the 3 dot symbol at the right end of the row for your navidrome proxy entry, and select 'Edit' from the pop-up menu.

In the proxy entry form, select the SSL tab, and from the first drop down list, select "Request a New SSL Certificate". Next, enable the "Force SSL" option, and then enter your email address if it's not already filled in. Finally, enable the "Accept Terms of Service" option.

Click 'Save'. This will prompt NGinX Proxy Manager to ask letsEncrypt for a new SSL certificate. LetsEncrypt will in turn try to reach your site on Port 80 (that's why we wanted to make sure it opened and got to the Navidrome site using our domain / sub-domain). After a few seconds (maybe a minute), the pop-up window should close with no errors.

Now click on the name in the row you see, and it should open your site in a new tab, with SSL enabled. Now create your admin user (if you haven't already), and make sure your music is showing up.

You're done! Congratulations! You can now install a subsonic capable music player on your smart device, enter your SSL encrypted domain, your user credentials, and start streaming your own music to your mobile device with no third party involvement!

Support my Channel and Content

Support my Channel and ongoing efforts through Patreon:

https://www.patreon.com/bePatron?u=234177