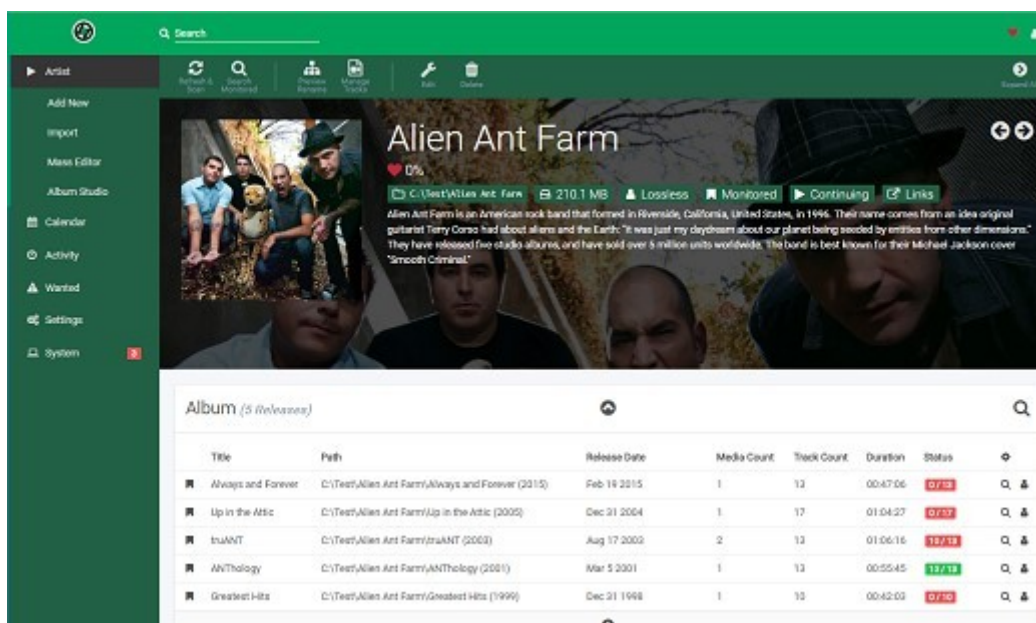


Lidarr



Lidarr

Lidarr is a music collection manager for Usenet and BitTorrent users. It can monitor multiple RSS feeds for new albums from your favorite artists and will interface with clients and indexers to grab, sort, and rename them. It can also be configured to automatically upgrade the quality of existing files in the library when a better quality format becomes available.



Installation

The Lidarr team does not offer an official Docker image. However, a number of third parties have created and maintain their own. These instructions provide generic guidance that should apply to any Lidarr Docker image.

I use a docker image from linuxserver.io to install Lidarr.

If you prefer to install Lidarr via another method go to [Lidarr's official downloads page](#).

Running Lidarr as a container

Basic examples for getting this image running as a container

Docker Compose

```
---
version: "2"
services:
  lidarr:
    image: linuxserver/lidarr:1.0.2
    container_name: lidarr
    restart: no
```

CLI

```
docker create \
  --name=lidarr \ --restart no \
  linuxserver/lidarr:1.0.2
```

Avoid common pitfalls

Volumes and Paths

There are two common problems with Docker volumes: Paths that differ between the Lidarr and download client container and paths that prevent fast moves and hard links.

The first is a problem because the download client will report a download's path as `/torrents/My.Music.2018/`, but in the Lidarr container that might be at `/downloads/My.Music.2018/`. The second is a performance issue and causes problems for seeding torrents. Both problems can be solved with well planned, consistent paths.

Most Docker images suggest paths like `/musics` and `/downloads`. This causes slow moves and doesn't allow hard links because they are considered two different file systems *inside* the container. Some also recommend paths for the download client container that are different from the Lidarr container, like `/torrents`.

The best solution is to use a single, common volume *inside* the containers, such as `/data`. Your Musics would be in `/data/Musics`, torrents in `/data/downloads/torrents` and/or usenet downloads in `/data/downloads/usenet`.

If this advice is not followed, you may have to configure a Remote Path Mapping in the Lidarr web UI (Settings > Download Clients).

Ownership and Permissions

Permissions and ownership of files is one of the most common problems for Lidarr users, both inside and outside Docker. Most images have environment variables that can be used to override the default user, group and umask, you should decide this before setting up all of your containers. The recommendation is to use a common group for all related containers so that each container can use the shared group permissions to read and write files on the mounted volumes. Keep in mind that Lidarr will need read and write to the download folders as well as the final folders.

Install Lidarr

To install and use these Docker images, you'll need to keep the above in mind while following their documentation. There are many ways to manage Docker images and containers too, so installation and maintenance of them will depend on the route you choose.

- [hotio/lidarr:release](#)

hotio doesn't specify any default volumes, besides `/config`. Images are automatically updated multiple times in an hour if upstream changes are found. Hotio also builds our Pull Requests which may be useful for testing. Read the [instructions](#) on how to install the image.

- [lscr.io/linuxserver/lidarr:latest](#)

[linuxserver.io](#) is one of the most prolific and popular Docker image maintainers. They also maintain images for most of the popular download clients as well. LinuxServer specifies a couple of optional default volumes such as `/music` and `/downloads`. The default volumes are not optimal nor recommended. Our recommendation is to use a single volume for the data, as mentioned above.

Revision #5

Created 2023-03-16 04:10:15 UTC by Tim

Updated 2023-04-17 16:14:19 UTC by Tim