

vladiiancu / **plex-reshare** Public

Re-share Plex's shared libraries with you further to other users on your server

MIT license

23 stars 11 forks Branches Tags Activity

Star

Notifications

&lt;&gt; Code Pull requests Actions Projects Security Insights

main

2 Branches 0 Tags

Go to file

Go to file

&lt;&gt; Code

...

peterbuga	fix: Empty ignore templates strings	6acb457 · 3 months ago
app	fix: Regression ignore marked files	3 months ago
nginx	feat: redis structure refactor	3 months ago
rq	fix: Empty ignore templates strings	3 months ago
.dockerignore	feat: PoC plex-reshare	4 months ago
.env.sample	feat: Configurable filtering options	3 months ago
.gitignore	feat: Ability to specify an ignore plex playlist to r...	3 months ago
LICENSE	Initial commit	4 months ago
Makefile	feat: redis structure refactor	3 months ago
README.md	fix: Empty ignore templates strings	3 months ago
docker-compose.yml	feat: redis structure refactor	3 months ago
pyproject.toml	chore: code cleanup	4 months ago
supervisord.conf	feat: redis structure refactor	3 months ago

README License

## PLEX-reshare

Combo of [openresty](#) + [starlette](#) + [rq](#) + [redis](#) to expose your Plex shares in a basic web-browsable `:8080` format a'la apache directory listing.

The reason behind this project it to make available your PLEX shares to other friends unrelated to the person who owns the original library.

For example, `plex-server-A` shares various libraries (Movies & TV Shows supported) to (your) `plex-server-owned-by-you-B` using `plex-reshare` create a new local library with the files from `plex-server-A` that you can later on can be shared directly to other friends `C`.

Basically `plex-reshare` will act as a plex-library-proxy and all the traffic will pass through it (downstream+upstream). It's ignoring self-libraries.

## Use scenario

You managed to get access to one or more shared libraries from other servers, with plex-reshare as a proxy you can host your own instance of plex and share it back to other close friends.

PS: it's not mandatory to use plex to share further the access, by same principle you can also use Jellyfin/Emby/any other media manager-indexer or even simply direct http access and open urls directly in VLC/IINA for example.

USE WITH CARE, **DO NOT HEAVILY REQUEST DATA FROM TARGET SERVERS**. BE NICE!

It'll create a http directory listing under the format

```

/
|-- movies
|  |-- c0e5a2.....
|  |  |-- movie.libraryA.movie1
|  |  |-- movie.libraryA.movie2
|  |  |-- movie.libraryB.movie1
|  |-- cb7d61.....
|  |-- e82c68.....
|  |-- f2423f.....
`-- shows
    |-- c0e5a2.....
    |  |-- tvshow.libraryA.show1
    |  |-- tvshow.libraryA.show2
    |  |-- tvshow.libraryB.show1
    |-- cb7d61.....
    |-- e82c68.....
    |-- f2423f.....

```

All the movie/shows libraries exposed by a specific plex server will be listed all in one place under a single served id uniquely identifiable.

As of now it's not made to recreate the structure defined by a specific plex(admin) but more like grouping all the data available and use external option like PMM (Plex Meta Manager) to create a more structured format out of (subject to change if needed/requested, please fill an issue!).

## Installation via Docker

Docker images available <https://hub.docker.com/r/peterbuga/plex-reshare>

### Sample command

```

docker run -d --name=plex-reshare \
-e PLEX_TOKEN='xxxxxxxxxxxxxxxx'
-p 8080:8080 \
peterbuga/plex-reshare:latest

```

### Docker compose

Copy `.env.sample` to `.env` and change the variable accordingly.

```
docker compose up -d
```

Browse to <http://your-host-ip:8080> to access the list of plex reshares.

### Environment variables

Variable	Description	Default
<code>PLEX_TOKEN</code>	(mandatory) find out <a href="#">how to get a plex token</a> .	(unset)
<code>REDIS_INTERNAL</code>	(optional) <code>true</code> or <code>false</code> use internal redis instance to store the files structure, using an internal refreshing system	<code>true</code>
<code>REDIS_HOST</code>	(optional) option to use an external redis instance if already available, set <code>REDIS_INTERNAL: false</code>	<code>127.0.0.1</code>
<code>REDIS_PORT</code>	(optional) to used when <code>REDIS_INTERNAL: false</code>	<code>11</code>
<code>REDIS_DB_RQ</code>	(optional) if python-rq should run on a separate redis db	<code>11</code>
<code>DATE_START</code>	(optional) needs to be under the format <code>YYYY-MM-DD</code> . limit the number of files exposed, increment by <code>FILES_DAY</code> daily.  this is to expose a subset of files to Plex initially and can scan them daily incremental. not	(unset)

Variable	Description	Default
	setting <code>DATE_START</code> will expose at once <b>ALL</b> the files it can find.  example: (today) 2024-03-05 - ( <code>DATE_START</code> ) 2024-02-01 * ( <code>FILES_DAY</code> ) 15 = 35 (days) * 15 => max 525 files will be exposed per library (either <code>movie</code> or <code>show</code> library type).	
<code>FILES_DAY</code>	(optional) how many files increment expose every day per library	25
<code>IGNORE_PLAYLIST</code>	(optional) generate a playlist, add items to it that you don't want to see in you library and they will slowly go away as the new refresh & trash process takes place	(unset)
<code>IGNORE_RESOLUTIONS</code>	(optional) list of resolutions comma separated that you'd like to ignore, ex: <code>sd</code>	(unset)
<code>IGNORE_EXTENSIONS</code>	(optional) list of file extension comma separated that you'd like to ignore, ex: <code>avi, mpeg</code>	(unset)
<code>MOVIE_MIN_SIZE</code>	(optional) minimal file size of a movie in Mb, everything below will be ignored	512
<code>EPISODE_MIN_SIZE</code>	(optional) minimal file size of an episode in Mb, everything below will be ignored	64
<code>IGNORE_MOVIE_TEMPLATES</code>	(optional) list of python regexes to ignore being added to the list, pipe (   ) separated, ex: <code>.*sample.*</code> will ignore all the sample file sometimes associated with movie files	(unset)
<code>IGNORE_EPISODE_TEMPLATES</code>	(optional) list of python regexes to ignore being added to the list, pipe (   ) separated	(unset)

## Local image build

The build image is a merge of multiple external dockerfiles (in order to kickstart the development) that's why there's no local Dockerfile defined

```
make build
```

## Rclone mount

More details here <https://rclone.org/http/> but I recommend using flags `--transfers 4 --low-level-retries 7 --retries 7 --tpslimit 0.7` to limit the access to API and files, otherwise plex scan will hammer the requests on target libraries.

## Development

### Linting:

Requires `pip install ruff==0.3.0`

```
make format-code
```

## Credits

- <https://github.com/openresty/docker-openresty>
- <https://github.com/tiangolo/uvicorn-gunicorn-docker>

## License

MIT license

### Releases

No releases published

### Packages

No packages published

### Languages

