

Useful Tools

companions to the *Arr Suite of Applications or media hoarding in general. They are not maintained, developed, nor supported by the *Arr Development Team.

Table of Contents

- ▶ [Table of Contents](#)
- ▶ [Recovering a Corrupt DB](#)
 - ▶ [Recovering a Corrupt DB \(UI\) \(Windows\)](#)
 - ▶ [Command Line DB Recovery](#)
- ▶ [Finding Cookies](#)
 - ▶ [General Instructions](#)
 - ▶ [Notes](#)
 - ▶ [Chrome](#)
 - ▶ [Firefox](#)
- ▶ [Clearing Cookies and Local Storage](#)
 - ▶ [Chrome](#)
 - ▶ [Firefox](#)
 - ▶ [Microsoft Edge \(Chromium\)](#)
- ▶ [Using Environment Variables for Config](#)
- ▶ [Other Projects and Programs - Request Apps *Arrs](#)
 - ▶ [Notifiarr \(fka Discord Notifier\)](#)
 - ▶ [Ombi](#)
 - ▶ [Overseerr](#)
 - ▶ [Jellyserr](#)
 - ▶ [Petio](#)
- ▶ [Other Projects and Programs - *Arr Related](#)
 - ▶ [Remote Control](#)
 - ▶ [LunaSea](#)
 - ▶ [Radarr & Sonarr Companion - Android App](#)
 - ▶ [nzb360 - Android App](#)
 - ▶ [Lidarr](#)

- ▶ [AMD](#)
- ▶ [AMVD](#)
- ▶ [Radarr](#)
- ▶ [AMTD](#)
- ▶ [Subtitles](#)
- ▶ [Bazarr](#)
- ▶ [Other Projects and Programs - Torrents/Downloading](#)
 - ▶ [Cross-Seed](#)
 - ▶ [Toolbarr](#)
 - ▶ [Unpackerr](#)
 - ▶ [qBit Management](#)
- ▶ [Other Projects and Programs](#)
 - ▶ [Filebot](#)
 - ▶ [JDupes](#)
 - ▶ [Drazzilb's UserScripts](#)
 - ▶ [Just A Bunch Of Starr Scripts](#)
 - ▶ [Just A Bunch Of Plex Scripts](#)
 - ▶ [Plex Meta Manager](#)
 - ▶ [Tautulli](#)
 - ▶ [Tdarr](#)
 - ▶ [tdarr_inform](#)
 - ▶ [Deleterr](#)
 - ▶ [Twitter Connect](#)

The following apps are companions to the *Arr Suite of Applications or media hoarding in general. They are not maintained, developed, nor supported by the *Arr Development Team. Please direct any specific support questions to the respective application development team.

Recovering a Corrupt DB

Note that the application's database can be found in the Application Data Directory which are linked below. The directory may also be passed as a datadir argument.

[Lidarr Appdata Directory](#)

Prowlarr Appdata Directory

Radarr Appdata Directory

Readarr Appdata Directory

Sonarr Appdata Directory



There are two options to recover the database which are listed below.

Using DB Browser for SQLite and using the UI

Using SQLite's `.recover` function

Recovering a Corrupt DB (UI) (Windows)



Note this effectively does the same as `.recover` which requires SQLite v3.29 | [Please refer to the SQLite docs for more details on the `.recover` command](#) . The steps to do so are linked below

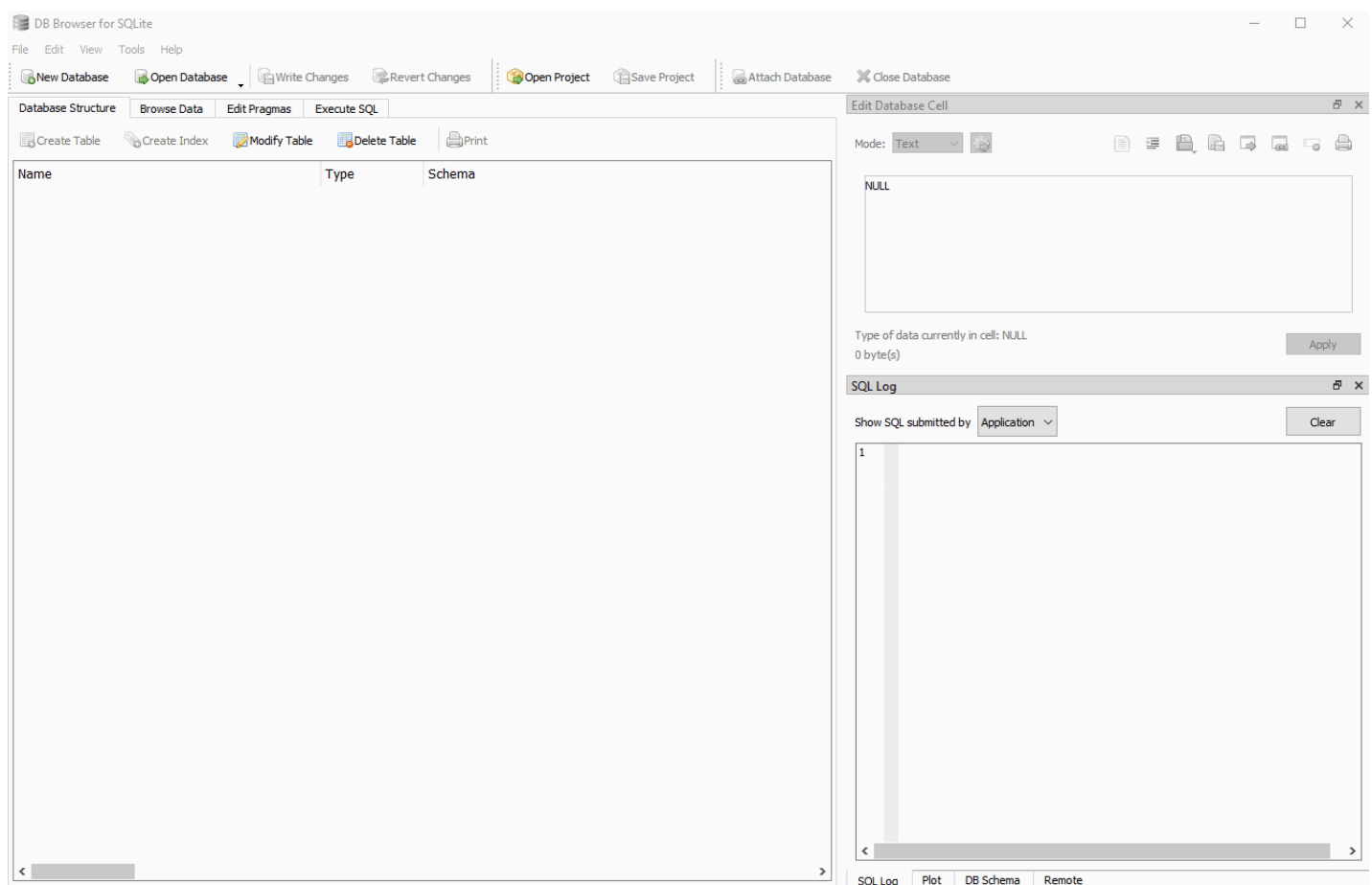


[DB Browser for SQLite \(DB4S\)](#) is a high quality, visual, open source tool to create, design, and edit database files compatible with SQLite. DB4S is for users and developers who want to create, search, and edit databases. DB4S uses a familiar spreadsheet-like interface, and complicated SQL commands do not have to be learned.

1. Stop the application
2. Make a copy of your corrupt database (`.db`) and copy any `.shm` and `.wal` files with it
3. Open your corrupt database in [DB Browser for SQLite \(DB4S\)](#)
4. File => Export => Export database to SQL file
5. Select all tables
6. Check/Enable "Keep column names in INSERT INTO"
7. Export Everything

8. Overwrite old schema
9. Save
10. Close the database
11. New Database => File => Import => import that file from the previous export step
12. Any import errors or constraint issues, clean up the problematic insert statement if possible or delete it
13. In the Execute SQL tab, execute the following SQL **VACUUM;**
14. Save the database when prompted.
15. Tools => Integrity Check; the result should say OK
16. Close the database
17. Remove all **wal**, **shm**, and **db** files from the config folder
18. Save (or copy, if *Arr is not on the same system as DB4S) the new database in the config folder and point the application at it. All *Arrs name their database as **<appname>.db** e.g. **radarr.db**
19. Correct permissions for the recovered database if needed. The owner should be the user and group *Arr is configured to run as.
20. Start the application

Please note that the gif does not cover the **VACUUM;** command



Command Line DB Recovery

The below instructions are for *Nix Operating Systems, but the concept will be similar on Windows Command Line.



This uses the [sqlite3](#) `.recover` [command](#) is ideal. Note that it requires Sqlite 3.29+



Given sqlite3 is required by *Arrs it is assumed you have sqlite3 installed on your system

1. Stop the application
2. SSH into your box or otherwise get a shell up
3. Enter `sqlite3 <path to bad database> ".recover" | sqlite3 <output path for recovered database>`
4. Correct permissions for the recovered database if needed. The owner should be the user and group *Arr is configured to run as.
5. Remove or move/rename the old corrupt database and any `wal` or `shm` in the folder
6. Rename the covered database. All *Arrs name their database as `<appname>.db` e.g. `radarr.db`
7. Start the application

Finding Cookies

- Some sites cannot be logged into automatically and require you to login manually then give the cookies to to work. The pages below describe how you do that.

General Instructions

1. Login to the Web Site with your browser using the Site Link address you have chosen to use from the Prowlarr indexer config
2. Open the DevTools panel by pressing F12
3. Select the Network tab
4. Click on the Doc button (Chrome Browser) or HTML button (Firefox Browser)
5. Refresh the page by pressing F5
6. Click on the first row entry
7. Select the Headers tab on the right panel
8. Find 'cookie:' in the Request Headers section. Refer to the helptext within the UI for your tracker for details
9. Select and Copy the whole cookie string (everything after the cookie:)

10. Delete anything already in the Prowlarr indexer config cookie box

11. Paste your copied cookie string to that box

12. Click Save



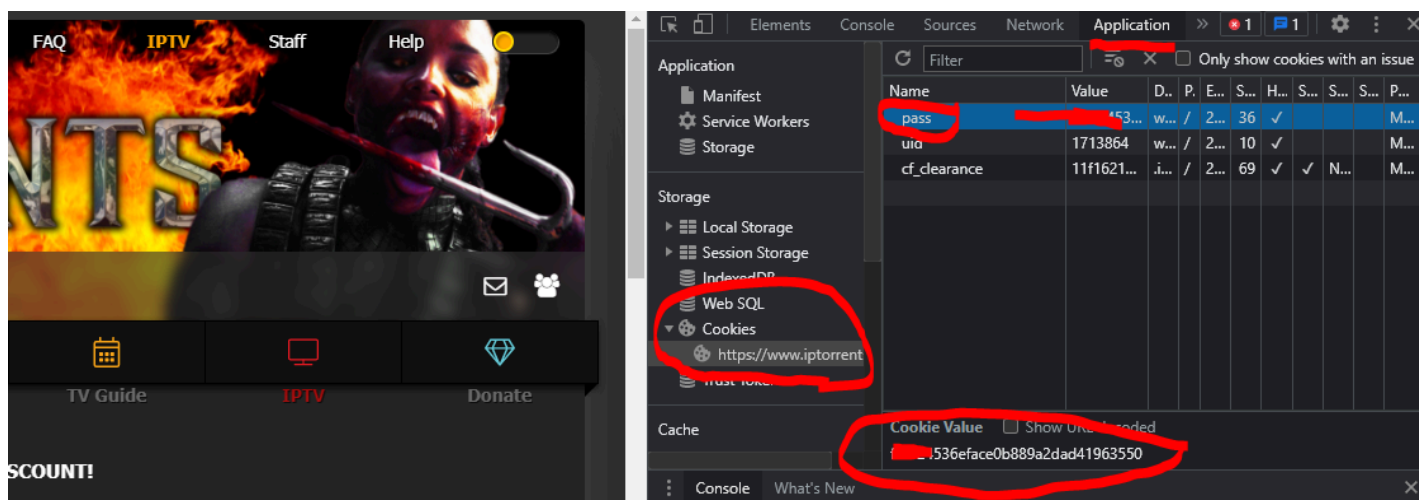
If user agent is required for your tracker, it can be found in the Request Headers

Notes

- ▶ Be sure to use the Browser from the same machine that is running Prowlarr, as cookies will rarely work from other platforms.
- ▶ On the login page for the Web site do not tick any options that restrict your session to one IP address, or logout after a short time. If there is a Remember me option, use it.
- ▶ Make sure that before fetching the cookie you can access the Site's torrent search page, as anything the site does to prevent this (alerts, unread notices or unread PM, or a low ratio warning) will stop Prowlarr from having a successful test after using the cookie.

Chrome

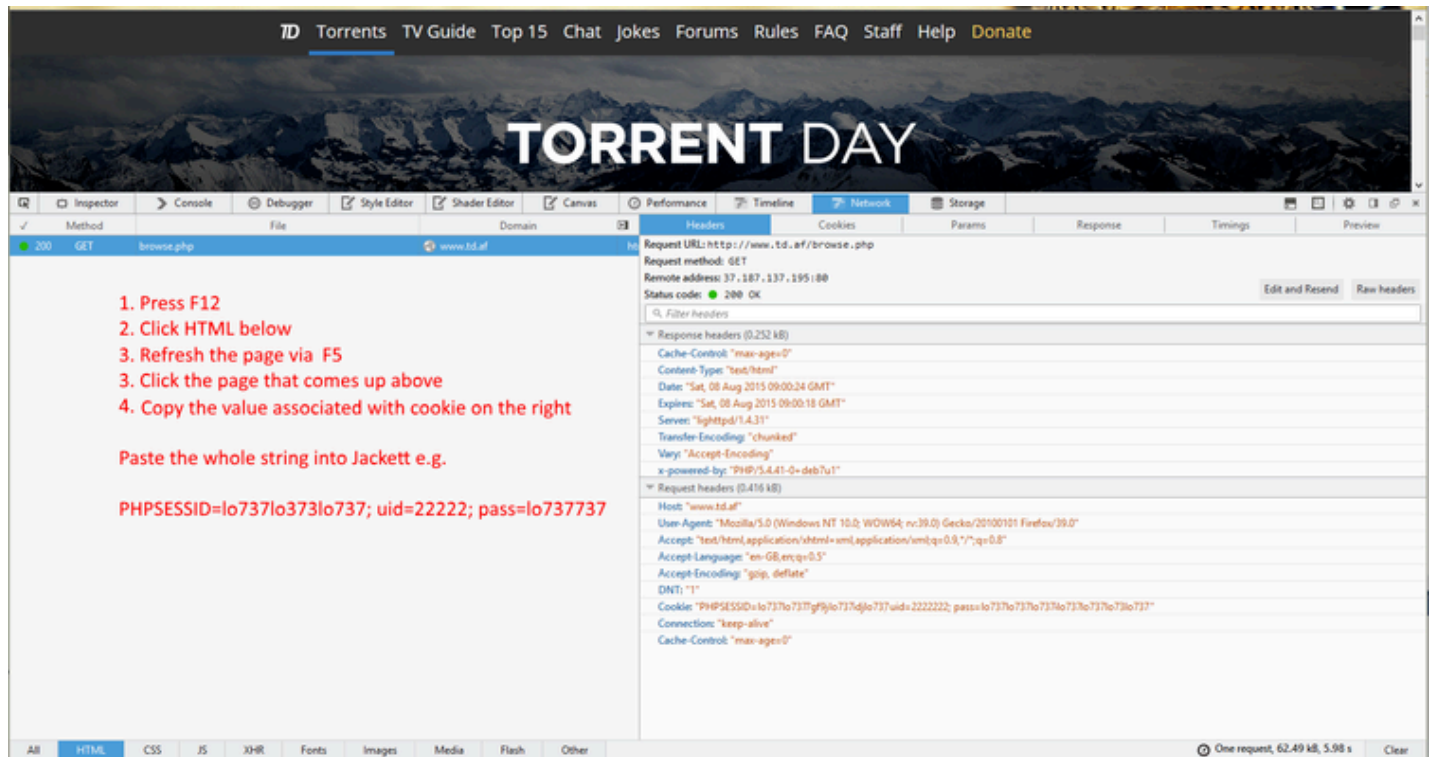
- ▶ Go to the torrent tracker website and log in.
- ▶ Hit F12
- ▶ Under the Application tab at the top, there will be "Storage" on the left side. You will see a "Cookies" subsection, and under that you will see your tracker's url. Click on that.
- ▶ Click on "Pass" on that tab or a similar entry, and it will pop up a box that says "Cookie Value" with a string about 25-30 chars long. Copy that and paste it into the application that needs it.
- ▶ If the string looks similar to `cid=cid-that-you-got-from-the-browser; sid=sid-that-you-got-from-the-browser` then the entire entry should be used.



- ▶ You may also reference Chrome's documents [Chrome cookies](#)

Firefox

► [Please see Mozilla's documentation](#) 



Clearing Cookies and Local Storage

Chrome

1. Navigate to `chrome://settings/siteData`
2. Enter the site (or app) name you wish to clear
3. Click the trash icon for the site

Firefox

► Please see [Mozilla's Help Article](#) 

Microsoft Edge (Chromium)

1. Navigate to `edge://settings/siteData`
2. Enter the site (or app) name you wish to clear
3. Click the arrow for the site
4. Click the trash icon for the site

Using Environment Variables for Config

All of the *arrs now have the ability to use environment variables to override entries in config.xml. The pattern for variable naming is predictable and can be used to set any config entry. In general the environment variables are comprised of 3 parts, delimited by two underscores.

APPNAME__CONFIGNAMESPACE__CONFIGITEM

The app name that should be used is simply the app name that the environment variable is passed to: **RADARR**, **SONARR**, **PROWLARR**, etc.

The config namespaces are currently predictable and shared between all apps. It is simply the option types for the options files in the project, located in the **src/NzbDrone.Common/Options** directory. The namespaces are currently:

- ▶ APP
- ▶ AUTH
- ▶ LOG
- ▶ POSTGRES
- ▶ SERVER
- ▶ UPDATE

Each of these options namespaces has several config items below it. Below you can find the entire directory for each. You must replace ***ARR** with whatever *arr app you are setting up. These have not all been tested and may be wrong.


Option Name in config.xml	Namespace	Option Name	Full Environment Variable Name
InstanceName	APP	INSTANCENAME	*ARR__APP__INSTANCENAME
Theme	APP	THEME	*ARR__APP__THEME
LaunchBrowser	APP	LAUNCHBROWSER	*ARR__APP__LAUNCHBROWSER
ApiKey	AUTH	APIKEY	*ARR__AUTH__APIKEY
AuthenticationEnabled	AUTH	ENABLED	*ARR__AUTH__ENABLED
AuthenticationMethod	AUTH	METHOD	*ARR__AUTH__METHOD
AuthenticationRequired	AUTH	REQUIRED	*ARR__AUTH__REQUIRED

Option Name in config.xml	Namespace	Option Name	Full Environment Variable Name
LogLevel	LOG	LEVEL	*ARR__LOG__LEVEL
FilterSentryEvents	LOG	FILTERSENTRYEVENTS	*ARR__LOG__FILTERSENTRYEVENTS
LogRotate	LOG	ROTATE	*ARR__LOG__ROTATE
LogSizeLimit	LOG	SIZELIMIT	*ARR__LOG__SIZELIMIT
LogSql	LOG	SQL	*ARR__LOG__SQL
ConsoleLogLevel	LOG	CONSOLELEVEL	*ARR__LOG__CONSOLELEVEL
ConsoleLogFormat	LOG	CONSOLEFORMAT	*ARR__LOG__CONSOLEFORMAT
AnalyticsEnabled	LOG	ANALYTICSENABLED	*ARR__LOG__ANALYTICSENABLED
SyslogServer	LOG	SYSLOGSERVER	*ARR__LOG__SYSLOGSERVER
SyslogPort	LOG	SYSLOGPORT	*ARR__LOG__SYSLOGPORT
SyslogLevel	LOG	SYSLOGLEVEL	*ARR__LOG__SYSLOGLEVEL
DbEnabled	LOG	DBENABLED	*ARR__LOG__DBENABLED
PostgresHost	POSTGRES	HOST	*ARR__POSTGRES__HOST
PostgresPort	POSTGRES	PORT	*ARR__POSTGRES__PORT
PostgresUser	POSTGRES	USER	*ARR__POSTGRES__USER
PostgresPassword	POSTGRES	PASSWORD	*ARR__POSTGRES__PASSWORD
PostgresMainDb	POSTGRES	MAINDB	*ARR__POSTGRES__MAINDB
PostgresLogDb	POSTGRES	LOGDB	*ARR__POSTGRES__LOGDB
UrlBase	SERVER	URLBASE	*ARR__SERVER__URLBASE
BindAddress	SERVER	BINDADDRESS	*ARR__SERVER__BINDADDRESS

Option Name in config.xml	Namespace	Option Name	Full Environment Variable Name
Port	SERVER	PORT	*ARR_SERVER_PORT
EnableSsl	SERVER	ENABLESSL	*ARR_SERVER_ENABLESSL
SslPort	SERVER	SSLPORT	*ARR_SERVER_SSLPORT
SslCertPath	SERVER	SSLCERTPATH	*ARR_SERVER_SSLCERTPATH
SslCertPassword	SERVER	SSLCERTPASSWORD	*ARR_SERVER_SSLCERTPASSWORD
UpdateMechanism	UPDATE	MECHANISM	*ARR_UPDATE_MECHANISM
UpdateAutomatically	UPDATE	AUTOMATICALLY	*ARR_UPDATE_AUTOMATICALLY
UpdateScriptPath	UPDATE	SCRIPTPATH	*ARR_UPDATE_SCRIPTPATH
Branch	UPDATE	BRANCH	*ARR_UPDATE_BRANCH

Other Projects and Programs - Request Apps *Arrs

Notifiarr (fka Discord Notifier)

[Notifiarr](#)  is a tool created to facilitate more in-depth detailed discord notifications by one of the *Arr Developers. It provides a configurable way to add notifications (including reactions) based on triggers you pick. Website provides a UI or picking what to show in the notification. Includes support for Grab, Import, Upgrade, Health, & Failed notifications in addition to much more.

[Wiki](#) 

Highlights

- ▶ Application Status
- ▶ Requests and Approvals (~Ombi)
- ▶ Customizable *ARR application notifications
- ▶ Request System with approvals
- ▶ Follow system for users to monitor a series or movie and be notified (via @mentions)
- ▶ Server Status
- ▶ Frequent New Features

Ombi

[Ombi](#)  gives users the ability to request movies, tv shows (series, seasons or single episodes) and music albums.


Overseerr

[Overseerr](#)  is a request management and media discovery tool built to work with your existing Plex ecosystem.

Jellyseerr

[Jellyseerr](#)  is a fork of Overseerr built to bring support for Jellyfin & Emby media servers.

Petio


[Petio](#)  is a third party companion app available to Plex server owners to allow their users to request, review and discover content.

The app is built to appear instantly familiar and intuitive to even the most tech-agnostic users. Petio will help you manage requests from your users, connect to other third party apps such as Sonarr and Radarr, notify users when content is available and track request progress. Petio also allows users to discover media both on and off your server, quickly and easily find related content and review to leave their opinion for other users.

Other Projects and Programs - *Arr Related

Remote Control



LunaSea

[LunaSea](#)  is a fully featured, open source self-hosted controller! Focused on giving you a seamless experience between all of your self-hosted media software. Manage new media content fetched via Lidarr, Radarr, and Sonarr.

Radarr & Sonarr Companion - Android App


Add new movies/shows to your system easily with your phone. App available at [Google Play](#) 

nzb360 - Android App


nzb360 provides management of Sonarr, Radarr, Lidarr, torrents, usenet, and other services. App available at [Google Play](#)  Check [official website](#)  for more info.

Lidarr

AMD


[Automated Music Downloader](#)  RandomNinjaAtk/amd is a Lidarr companion script to automatically download music for Lidarr

AMVD

[Automated Music Video Downloader](#)  RandomNinjaAtk/amvd is a Lidarr companion script to automatically download and tag Music Videos for use in other video applications (plex/kodi/jellyfin/emby)


Radarr

AMTD

[Automated Movie Trailer Downloader](#)  RandomNinjaAtk/amtd is a Radarr companion script to automatically download movie trailers and extras for use in other video applications (plex/kodi/jellyfin/emby)


Subtitles

Bazarr



[Bazarr](#)  is a companion application to Sonarr and Radarr that manages and downloads subtitles based on your requirements.

Other Projects and Programs - Torrents/Downloading


Cross-Seed

[Cross-Seed](#)  is an app designed to help you download torrents that you can cross seed based on your existing torrents. It is designed to match conservatively to minimize manual intervention. It supports Jackett and Qbittorrent/rTorrent at this time.

Toolbarr

[Toolbarr](#)  provides a suite of utilities to fix problems with Starr applications. Toolbarr allows you to perform various actions against your Starr apps and their SQLite3 databases. The most useful [feature is being able to convert paths from linux to windows and windows to linux](#) .

Unpackerr

[Unpackerr](#)  This application runs as a daemon on your download host. It checks for completed downloads and extracts them so *Arr may import them.

- There are a handful of options out there for extracting and deleting files after your client downloads them. Captain just didn't care for any of them, so he wrote his own. He wanted a small single-binary with reasonable logging that can extract downloaded archives and clean up the mess after they've been imported.

qBit Management


[qBit Management a.k.a. "qbit_manage"](#)  is a program used to manage your qBittorrent instance such as:

- Tag torrents based on tracker URL (only tag torrents that have no tags)


- ▶ Update categories based on save directory
- ▶ Remove unregistered torrents (delete data & torrent if it is not being cross-seeded, otherwise it will just remove the torrent)
- ▶ Automatically add cross-seed torrents in paused state (used in conjunction with the [cross-seed script](#))
- ▶ Recheck paused torrents sorted by lowest size and resume if completed
- ▶ Remove orphaned files from your root directory that are not referenced by qBittorrent
- ▶ Tag any torrents that have no hard links and utilize Share Limits to remove them
- ▶ Apply Share Limits based on tags and categories

Other Projects and Programs

Filebot

[FileBot](#)  is the ultimate tool for organizing and renaming your Movies, TV Shows and Anime as well as fetching subtitles and artwork. It's smart and just works.

JDupes





[Jdupes](#)  is a program for identifying and taking actions upon duplicate files.



TRaSH [has a guide](#)  as well


- ▶ `jdupes -M -r "/data/tv/" "/data/tv/.torrents/"` <= this would check for double files and print a summary of the results
- ▶ `jdupes -L -r "/data/tv/" "/data/tv/.torrents/"` <= this would recreate them as hard links thus reducing the used duplicate space

Drazzilb's UserScripts


- ▶ [Drazzilb's collection of unRAID \(and native compatible\) userscripts](#)  and other scripts around the internet that he finds neat/useful. The [list of scripts can be found on his wiki](#)  .
- ▶ [Drazzilb's Upgradinorr](#)  is a script is designed to keep your media in sync with your Radarr/Sonarr's Custom Formats/Scoring. It is a python script to manually search N items that are not tagged with a specific tag in your Radarr/Sonarr media library. N is the number of items this script will search for, this has the added benefit that you don't hammer your indexers and get banned 😊 This script is a python port of [Cuban's Powershell Upgradinorr](#)  ; thanks again Cuban.

Just A Bunch Of Starr Scripts


- ▶ [Just A Bunch Of Starr Scripts](#) 

- ▶ [Cuban's Upgradinator](#)  is a Powershell script to manually search N items that are not tagged with a specific tag in your Radarr/Sonarr media library. N is the number of items this script will search for, this has the added benefit that you don't hammer your indexers and get banned 😊

Just A Bunch Of Plex Scripts


- ▶ [Just A Bunch Of Plex Scripts \(JBOPS\)](#) 
- ▶ [TRaSH Guides JBOPS 4K Transcode Stopping with Tautulli](#) 

Kometa


[Kometa \(FKA Meta Manager \(PMM\)\)](#)  is a Python script to update metadata information for movies, shows, and collections as well as automatically build collections.

Kometa (formerly known as Plex Meta Manager) is a powerful tool designed to give you complete control over your media libraries. With Kometa, you can take your customization to the next level, with granular control over metadata, collections, overlays, and much more.


Tautulli

[Tautulli](#)  is a 3rd party application that you can run alongside your Plex Media Server to monitor activity and track various statistics. Most importantly, these statistics include what has been watched, who watched it, when and where they watched it, and how it was watched. The only thing missing is "why they watched it", but who am I to question your 42 plays of Frozen. All statistics are presented in a nice and clean interface with many tables and graphs, which makes it easy to brag about your server to everyone else.


Tdarr

[Tdarr](#)  is a closed-source self hosted web-app for automating media library transcode/remux management and making sure your files are exactly how you need them to be in terms of codecs/streams/containers etc. Designed to work alongside [Sonarr](#)/[Radarr](#) and built with the aim of modularization, parallelization and scalability, each library you add has its own transcode settings, filters and schedule. Workers can be fired up and closed down as necessary, and are split into 3 types - 'general', 'transcode' and 'health check'. Worker limits can be managed by the scheduler as well as manually.


tdarr_inform



[tdarr_inform](#)  is a custom script for Sonarr and Radarr to inform Tdarr of new/changed/deleted files without relying on filesystem events or frequent disk scanning.

Deleterr

[Deleterr](#)  is a tool to delete stale and inactive media from Plex/Sonarr/Radarr. It helps managing limited space when you allow users to request shows via Overseerr/Ombi but don't want to manually monitor available disk space. It's configurable to support only deleting media meeting your defined criteria.

Twitter Connect

Create a Twitter application (if you haven't already) at <https://apps.twitter.com/> 

Fill in the mandatory fields as well as the callback URL, set it to a publicly available URL (not localhost), it doesn't need to exist, but it does need to be set, using <https://sonarr.tv/twitter>  or <https://radarr.video>  is sufficient.

Powered by [Wiki.js](#)