Find Corrupted Torrents

Take care while deleting files within your torrent session state. We do not take any responsibility for accidental data deletion and cannot restore deleted data. The below procedures are provided for our clients to use at their own risk.

This guide will cover how to find and remove a corrupted torrent in your torrent client. Occasionally, a torrent can become corrupted and cause your torrent client to crash. This can be somewhat complicated to troubleshoot and diagnose, as there are many reasons for why a torrent client might crash.

Before we begin, let's introduce the available torrent clients, their strengths, and weaknesses. The information below will help you decide what torrent client to use. Please note that the below characteristics are highly subjective and may not align with your specific use case. The number of torrents listed is are soft limits.

- **Deluge** - Unstable and fast downloading, with a limit of **100** torrents for v1 and **2000** torrents for v2
- **qBittorrent** - Stable and fast seeding, with a limit of **10K** torrents
- **rTorrent** - Unstable and slow seeding, with a limit of **2000** torrents
- **Transmission** - Slow and featureless, with a limit of **25K** torrents

These torrent clients are running in a shared environment on Ultra.cc. We do not recommend nor officially support torrent clients with torrent sessions above these limits.

qBittorrent

If your qBittorrent instance is able to start up, but immediately becomes unreachable, it is possible that you have torrent files that are corrupted. To locate a corrupt torrent file in qBittorrent, you need to monitor the log output. Logging is enabled by default in qBittorrent and no further action is needed to get started.

If the qBittorrent log file throws torrent file errors without the hash string included, you need to locate the torrent hash from someplace else. A possible solution is to upgrade your qBittorrent instance to a newer version (if you aren't already on the newest version) and restore your torrent session. Then, restart qBittorrent and fetch the torrent hash from the webUI before qBittorrent crashes. This is not always possible, but will work in some cases. See our Torrent Session Migration guide for instructions on how to restore a torrent session.

- Connect to your Ultra.cc service via SSH
- Before you begin troubleshooting, you should backup your qBittorrent instance by following the instructions here.
- Alternatively, to manually backup qBittorrent, you can make copies of the below directory paths.
  - `~/.local/share/qBittorrent`
  - `~/.config/qBittorrent`
- Next, you can begin monitoring the qBittorrent log file in real-time by executing the following command:
  - To stop the real-time monitoring, press `CTRL+C`.

```
tail -f ~/.local/share/qBittorrent/logs/qbittorrent.log
```

- While you are monitoring the qBittorrent log file in real-time, proceed to restart your qBittorrent instance.
  - Restart qBittorrent from the **Apps** tab on the UCP, or by executing `app-qbittorrent restart`.
- During the restart of qBittorrent, keep an eye on the real-time monitoring of the log file and look for any errors.
  - If errors are found, take note of the error and exit the real-time monitoring by pressing `CTRL+C`.
- Next, if there were any corrupted torrents, you will see errors such as `Unable to resume torrent '<hash>'` and/or `File error alert`.
- Your next course of action would be to locate these torrents, which can be done by executing the following command:

```
cat ~/.local/qbittorrent/logs/qbittorrent* | grep -i unable
```

- The above command will list all the corrupted torrent files as `<hash>.torrent`. Take not of these files.
  - The format to note would look like this: `fe44aede8630g8dab52263da06e1e1lblc693136.torrent`
- Next, stop qBittorrent from the **Apps** tab on the UCP, or by executing `app-qbittorrent stop`.
- Once qBittorrent has been stopped, you can proceed to delete all the corrupted torrents.
- To delete a torrent, execute the following command for each of the corrupted torrents:
  - Be sure to replace the alphanumerical string with the filenames that you have noted previously.

```
rm ~/.local/share/qBittorrent/BT_BACKUP/fe44aede868a38dab52263da06e1e1bddc693136.torrent
```

- Once all the corrupted torrents have been deleted, you can proceed to restart qBittorrent.
  - Restart qBittorrent from the **Apps** tab on the UCP, or by executing `app-qbittorrent restart`.
- If all went well, qBittorrent should start up without issues. However, if qBittorrent still crashes upon start up, you need to restart the procedure and monitor the log file while restarting qBittorrent.

rTorrent

To locate a corrupt torrent file in rTorrent, you need to monitor the log output during the crash of rTorrent. By default, rTorrent logging is disabled and needs to be enabled.

> Enabling logging of rTorrent should only be done temporarily. Leaving logging enabled for a prolonged time will create very large log files, and could lead to your storage being filled up.

- Connect to your Ultra.cc service via SSH
- Stop rTorrent from the UCP, or by executing `app-rtorrent stop`
- Create a `log` directory and backup torrent session by executing the following command:

```
mkdir ~/.config/rtorrent/logs && cp -r ~/.config/rtorrent/session
~/.config/rtorrent/session.bak
```

- Next, to enable logging in rTorrent, open `.rtorrent.rc` with the nano editor:

```
nano ~/.rtorrent.rc
```

- At the end of the configuration file, paste the following:

```
log.open_file = "rtorrent", ~/.config/rtorrent/logs/rtorrent.log
log.open_file = "tracker", ~/.config/rtorrent/logs/tracker.log
log.open_file = "storage", ~/.config/rtorrent/logs/storage.log
log.open_file = "network", ~/.config/rtorrent/logs/network.log
log.add_output = "info", "rtorrent"
log.add_output = "critical", "rtorrent"
log.add_output = "error", "rtorrent"
log.add_output = "warn", "rtorrent"
log.add_output = "notice", "rtorrent"
log.add_output = "debug", "rtorrent"
log.add_output = "dht_debug", "tracker"
log.add_output = "tracker_debug", "tracker"
log.add_output = "storage_debug", "storage"
```

- Next, restart rTorrent from the UCP, or by executing `app-rtorrent restart`
- Now, wait until rTorrent crashes and then check the logs to see what torrent was last logged
- To output the log in your terminal, execute the following command:

```
tail ~/.config/rtorrent/logs/storage.log
```

- Find the last torrent logged and execute the command in the box below:
  - The `TORRENTID` will be in the format of `4CC1F979463A4852574E0DE8E111084C19342F83`
  - Make sure to include the `*` at the end of the command, as it will remove all related files.

```
rm ~/.config/rtorrent/session/TORRENTID*
```

- Then, restart rTorrent from the UCP, or by executing `app-rtorrent restart`
- Next, if rTorrent crashes again, repeat the process.
  - Stop rTorrent
  - Find the latest torrent ID in the log
  - Remove the files associated with the torrent ID
  - Restart rTorrent

- Once rTorrent no longer crashes, you can disable logging.
- Stop rTorrent from the UCP, or by executing `app-rtorrent restart`
- Remove the lines that you previously added to the configuration file `~/.rtorrent.rc`
- Restart rTorrent from the UCP, or by executing `app-rtorrent restart`
- Done!