
deluge-webapi Documentation

Release 0.4.0

Igor 'idle sign' Starikov

Sep 28, 2019

Contents

1	Description	3
2	Requirements	5
3	Installation	7
4	Table of Contents	9
4.1	Quick Reference	9
5	Get involved into deluge-webapi	13
6	More things	15

<https://github.com/idlesign/deluge-webapi>

CHAPTER 1

Description

Plugin for Deluge WebUI providing sane JSON API.

Exposes some sane JSON API through Deluge WebUI (missing out of the box or restyled) so that you can manipulate Deluge from your programs using HTTP requests.

CHAPTER 2

Requirements

1. Python 2.7+
2. Deluge 1.3.15+

CHAPTER 3

Installation

- Get plugin egg file, from `dist/` (<https://github.com/idlesign/deluge-webapi/tree/master/dist>) directory.
- Open Deluge Web UI, go to “Preferences -> Plugins -> Install plugin” and choose egg file.

Note: To build .egg file from source code yourself use `python setup.py bdist_egg` command in source code directory.

- Activate *WebAPI* plugin.
- You're ready.

4.1 Quick Reference

Note: First make sure you're using WebUI and WebAPI plugin is active under "Preferences -> Plugins".

4.1.1 Hints

1. WebAPI exists alongside with Deluge built-in JSON API, so all HTTP requests should proceed to `http://localhost:8112/json` (or what ever you've got instead of localhost)
2. Use POST method for HTTP requests.
3. Make requests with valid JSON (including `Accept: application/json` and `Content-Type: application/json` HTTP headers).
4. JSON must include the following fields:
 - `id` - message number for identity (could be any, increment it freely)
 - `method` - API method name. E.g.: `auth.login`, `webapi.get_torrents`
 - `params` - API method parameters (depend on method)
5. Login beforehand.
 - Send

```
{"id": 1, "method": "auth.login", "params": ["your_password_here"]}
```

 and receive cookies with auth information.
 - Use those cookies for every request.
6. API answers with an error with the following JSON:

```
{"error": {"message": "Some error description."}, "id": 1, "result": False}
```

Check responses for `error` field.

7. Make sure Deluge WebUI is connected to Deluge daemon that makes actual torrent processing.
 - Send `{"id": 1, "method": "auth.check_session", "params": []}` and verify no error.
8. WebAPI method names start with `webapi..` (E.g.: `webapi.add_torrent` to call `add_torrent` function).
9. See <https://github.com/idlesign/deluge-webapi/blob/master/webapi/test.py> for code sample. Run it for basic debug.

4.1.2 Debug

If you have problems with plugin activation, you can get log with useful information (look for string containing `webapi`).

For `deluged`:

```
$ deluged -d -L info
```

For `deluged-web`:

```
$ deluge-web -L info
```

You can run <https://github.com/idlesign/deluge-webapi/blob/master/webapi/test.py> for basic test.

4.1.3 API Methods

Note: WebAPI uses torrent hashes to identify torrents, so torrent ID is the same as hash.

Get torrents info

```
get_torrents(ids=None, params=None)
```

Returns information about all or a definite torrent. Returned information can be filtered by supplying wanted parameter names.

Add torrent

```
add_torrent(metainfo, options=None)
```

Adds a torrent with the given options. *metainfo* could either be base64 torrent data or a magnet link.

Remove torrent

```
remove_torrent(torrent_id, remove_data=False)
```

Removes a given torrent. Optionally can remove data.

Get API version

```
get_api_version()
```

Returns WebAPI plugin version.

Get involved into deluge-webapi

Submit issues. If you spotted something weird in application behavior or want to propose a feature you can do that at <https://github.com/idlesign/deluge-webapi/issues>

Write code. If you are eager to participate in application development, fork it at <https://github.com/idlesign/deluge-webapi>, write your code, whether it should be a bugfix or a feature implementation, and make a pull request right from the forked project page.

Spread the word. If you have some tips and tricks or any other words in mind that you think might be of interest for the others — publish it.

CHAPTER 6

More things

- You might be interested in considering other Deluge plugins at <http://dev.deluge-torrent.org/wiki/Plugins/>
- Or you might be interested in a PHP Deluge API wrapper: <https://github.com/kaysond/deluge-php>