

---

# **Read the Docs Template Documentation**

*Release 1.0*

**Read the Docs**

**Jul 19, 2019**



---

## Contents

---

<b>1</b>	<b>Authors</b>	<b>3</b>
<b>2</b>	<b>Installation</b>	<b>5</b>
<b>3</b>	<b>OTF - One Time File Sharing</b>	<b>7</b>
3.1	What is /otf? . . . . .	7
3.2	How does it differ from other EFS-services? . . . . .	7
3.3	Drawbacks? . . . . .	7
3.4	Why should I use /otf? . . . . .	8
<b>4</b>	<b>Usage</b>	<b>9</b>
<b>5</b>	<b>Indices and tables</b>	<b>13</b>



Contents:



# CHAPTER 1

---

Authors

---

- mad\_dev





## CHAPTER 2

---

### Installation

---

The source code will eventually be released to the public, until then:

<https://ki.tc/otf>



---

## OTF - One Time File Sharing

---

### 3.1 What is /otf?

- /otf is part of my CLI for the web project; dubbed as Keep It Tidy Charles(kite). As the name suggests; it's an Ephemeral File Sharing(EFS)-service that was designed initially as an api. The system has random readable urls/path through what3words's api; rationale: easier to read > easier to memorize > easier to verbally share. The process is actually fascinating as it's generating random longitudes and latitudes for any given upload, passes it to what3words api, and then, the 3 words are passed and stored. When the api fails, it falls back to 5 random characters.
- **There are two links for any given file:** [1] The download link [2] The download page link.

The download page link was created to circumvent file deletion due to services that request header info[many social media sites and IM services do this].

- File retention is 27 days.
- Max upload size is 400mb.

### 3.2 How does it differ from other EFS-services?

To be completely honest, it doesn't.

### 3.3 Drawbacks?

Aesthetics and an upload bar.

### 3.4 Why should I use /otf?

With the introduction of UID and IID, the system allows or a marginal trust between the uploader and the recipient.

>\_CLI:

```
curl -F "file=@59b5b" https://ki.tc/file/u/
{
  "file": {
    "_id": "perfecting.classics.postsurgical",
    "approved": "False",
    "download_page": "https://ki.tc/file/perfecting.classics.postsurgical",
    "file": [
      "File Name: 59b5b",
      "Content Type: application/octet-stream"
    ],
    "ip256": "2d54e0e74f8881713a7ea858fcdbe18f02d132ba640028f6b1c6a7b2a8d18446",
    "length": "3K",
    "link": "https://ki.tc/f/perfecting.classics.postsurgical",
    "sha256": "d81dd207b277be80af3698b0872da2a35ae733f19c07b17b2cc41ca8ae1cf39a",
    "time": "Fri, 19 Jul 2019 20:20:10 GMT"
  }
}
```

Random readable urls/path; rationale: easier to read > easier to memorize > easier to verbally share. When the w3w api fails, it falls back to 5 characters:

```
curl -F "file=@59b5b" https://ki.tc/file/u/
{
  "file": {
    "_id": "4fed2",
    "approved": "False",
    "download_page": "https://ki.tc/file/4fed2",
    "file": [
      "File Name: 59b5b",
      "Content Type: application/octet-stream"
    ],
    "ip256": "2d54e0e74f8881713a7ea858fcdbe18f02d132ba640028f6b1c6a7b2a8d18446",
```

(continues on next page)

(continued from previous page)

```
"length": "3K",
"link": "https://ki.tc/f/4fed2",
"sha256": "d81dd207b277be80af3698b0872da2a35ae733f19c07b17b2cc41ca8ae1cf39a",
"time": "Fri, 19 Jul 2019 20:20:10 GMT"
}
}
```

**DONT TRUST THE SERVICE...ALWAYS ENCRYPT:**

```
keybase encrypt -i Raspberry_Pi_Logo.svg.png -o r_keybase chris

openssl enc -aes-256-cbc -salt -in Raspberry_Pi_Logo.svg.png -out r_encrypted
```

**Using IID and UID:**

**Generate UID:**

```
curl -X POST -F 'keybase_id=YOUR_KEYBASE-USERNAME' https://ki.tc/otf/generate_key

{
  "generate_key": {
    "Attention": "Make sure it's public",
    "Instructions": "Create a text file under /otf so that --> https://keybase.pub/
↳YOUR_KEYBASE-USERNAME'/otf/otf_key.txt Use /check_key to validate your uid",
    "keybase_id_status": true,
    "uid": "9ff40341dc7c978839abd831595f65ef2b4aba9cda79d7c1057f335ac6dd66ec"
  }
}
```

Check UID in Keybase, make sure you have created [https://keybase.pub/YOUR\\_KEYBASE-USERNAME/otf/otf\\_key.txt](https://keybase.pub/YOUR_KEYBASE-USERNAME/otf/otf_key.txt):

```
curl -X POST -F 'uid=9ff40341dc7c978839abd831595f65ef2b4aba9cda79d7c1057f335ac6dd66ec
↳' https://ki.tc/otf/check_key

{
  "check": {
    "Attention": "Do not lose your iid",
    "iid": "ca98cc2b5cd7d1b8cabaa748a252623c54c65942a608b315bfe2dcfb44cd66e7",
    "match_condition": true
  }
}
```

**Upload using your new IID:**

```
curl i -H "iid:ca98cc2b5cd7d1b8cabaa748a252623c54c65942a608b315bfe2dcfb44cd66e7" -F
↳"file=@59b5b" https://ki.tc/file/u/

HTTP/1.1 100 Continue

HTTP/1.1 201 CREATED
```

(continues on next page)

(continued from previous page)

```
Date: Fri, 19 Jul 2019 19:02:28 GMT
Server: Apache/2.4.18 (Ubuntu)
Content-Length: 635
Content-Type: application/json
```

```
{
  "file": {
    "_id": "carnality.elderly.unreservedly",
    "approved": "YOUR_KEYBASE-USERNAME@ki.tc",
    "download_page": "https://ki.tc/file/carnality.elderly.unreservedly",
    "file": [
      "File Name: 59b5b",
      "Content Type: application/octet-stream"
    ],
    "ip256": "79f4fe7c35ed5cba2b5c74613e017a84680d7f96848303038de418b1fc23f71a",
    "length": "3K",
    "link": "https://ki.tc/f/carnality.elderly.unreservedly",
    "sha256": "d81dd207b277be80af3698b0872da2a35ae733f19c07b17b2cc41ca8ae1cf39a",
    "time": "Fri, 19 Jul 2019 19:02:28 GMT",
    "uidcert": "544a9894a7033d2dc8a1f822f7d9e78ed2baa56bfabffa9c5efc96a226180cb6"
  }
}
```





## CHAPTER 5

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`