

Datasafe REST Service

Table of Contents

1. Overview	3
2. Authentication API	3
3. User API	3
3.1. Objects	3
3.2. Methods	3
3.2.1. Create user	3
Request header	3
Request fields	3
Example request	3
Example success response	4
3.2.2. /user	4
Example request	4
Example success response	4
4. Private Document API	5
4.1. List Documents	5
4.1.1. Request header	5
4.1.2. Example request	5
4.1.3. Example success response	5
Response body	5
4.2. Read Document	5
4.2.1. Request header	6
4.2.2. Path fields	6
4.2.3. Example request	6
4.2.4. Example success response	6
Response body	6
4.3. Write Document	7
4.3.1. Request header	7
4.3.2. Path fields	7
4.3.3. Example request	7
4.3.4. Example success response	7
4.4. Remove Document	7
4.4.1. Request header	7
4.4.2. Path fields	8
4.4.3. Example request	8
4.4.4. Example success response	8
5. Inbox API	8

5.1. List Inbox	8
5.1.1. Request header	8
5.1.2. Example request	9
5.1.3. Example success response	9
Response body	9
5.2. Read from Inbox	9
5.2.1. Request header	9
5.2.2. Path fields	10
5.2.3. Example request	10
5.2.4. Example success response	10
Response body	10
5.3. Write to Inbox	10
5.3.1. Request header	10
5.3.2. Path fields	11
5.3.3. Example request	11
5.3.4. Example success response	11
5.4. Remove from Inbox	11
5.4.1. Request header	11
5.4.2. Path fields	11
5.4.3. Example request	12
5.4.4. Example success response	12
6. Versioned Document API	12
6.1. List Documents	12
6.1.1. Request header	12
6.1.2. Example request	12
6.1.3. Example success response	13
Response body	13
6.2. Read Document	13
6.2.1. Request header	13
6.2.2. Path fields	13
6.2.3. Example request	13
6.2.4. Example success response	14
Response body	14
6.3. Write Document	14
6.3.1. Request header	14
6.3.2. Path fields	14
6.3.3. Example request	15
6.3.4. Example success response	15
6.4. Remove Document	15
6.4.1. Request header	15
6.4.2. Path fields	15

6.4.3. Example request	15
6.4.4. Example success response	16

1. Overview

Datasafe REST Application exists for demonstration purposes of datasafe library. It consists endpoints for working with users, private documents, private documents with versioning support, public documents(documents exchange) and authentication.

2. Authentication API

3. User API

3.1. Objects

Table 1. UserDTO

Code	Description
Username	username.
Password	password.

3.2. Methods

3.2.1. Create user

This method allows to create new datasafe user. Http methods of request is PUT.

Request header

Name	Description
token	Bearer authentication token is required

Request fields

Path	Type	Description
userName	String	Name of user to create
password	String	Password of user

Example request

```
$ curl 'http://example.com/datasafe/user' -i -X PUT \
  -H 'Accept: application/json' \
  -H 'Content-Type: application/json;charset=UTF-8' \
  -H 'token: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiIjoic2VjdXJlLWFwc
CIsInN1YiI6InVzZXJuYV11IiwiaXhwIjoxNTYzNTU5MDAwLCJyb2wiOiUk9MRV9VU0VSIl19.3IRw989Hr8
e7o2qJyUB1VvIoZ3hmv_CgovG6dZX6fCWkrfzEdpaQEJGNTBgvH8ft02OWrpnNHPGeyYcKX1BqXA' \
  -d '{"userName":"testUser","password":"testPassword"}'
```

Example success response

```
HTTP/1.1 200 OK
Pragma: no-cache
X-XSS-Protection: 1; mode=block
Expires: 0
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

3.2.2. /user

This method deletes user and all his files from datasafe storage. Http method of request is DELETE.

Example request

```
$ curl 'http://example.com/datasafe/user' -i -X DELETE \
  -H 'Accept: application/json' \
  -H 'password: test' \
  -H 'user: test' \
  -H 'Content-Type: application/json;charset=UTF-8' \
  -H 'token: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiIjoic2VjdXJlLWFwc
CIsInN1YiI6InVzZXJuYV11IiwiaXhwIjoxNTYzNTU5MDAwLCJyb2wiOiUk9MRV9VU0VSIl19.3IRw989Hr8
e7o2qJyUB1VvIoZ3hmv_CgovG6dZX6fCWkrfzEdpaQEJGNTBgvH8ft02OWrpnNHPGeyYcKX1BqXA'
```

Example success response

```
HTTP/1.1 200 OK
Pragma: no-cache
X-XSS-Protection: 1; mode=block
Expires: 0
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

4. Private Document API

4.1. List Documents

4.1.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username
password	datasafe user's password

4.1.2. Example request

```
$ curl 'http://example.com/datasafe/documents/' -i -X GET \  
  -H 'token: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW11cmUtYXBpIiwiaXVkiIjoic2VjdXJLLWFwcCIsInN1YiI6InVzZXJuYW11IiwiaXhwIjoxNTYzNTU5MDExLCJyb2wiOiUk9MRV9VU0VSIl19.QmLO3ahgNtbxt0l02k7krXWLXD5Y4Ff3uDwxZv9vudisQjIVAQCxe65HqrR6MR7IsKU_4EgDx94PS6UDS0oyIw' \  
  -H 'password: test' \  
  -H 'user: test'
```

4.1.3. Example success response

```
HTTP/1.1 200 OK  
Pragma: no-cache  
Content-Length: 21  
X-XSS-Protection: 1; mode=block  
Expires: 0  
X-Frame-Options: DENY  
X-Content-Type-Options: nosniff  
Content-Type: application/json; charset=UTF-8  
Cache-Control: no-cache, no-store, max-age=0, must-revalidate  
  
["/path/to/file.txt"]
```

Response body

```
["/path/to/file.txt"]
```

4.2. Read Document

4.2.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username
password	datasafe user's password

4.2.2. Path fields

Table 2. /document/{path}

Parameter	Description
path	path to the file

4.2.3. Example request

```
$ curl 'http://example.com/datasafe/document/path/to/file' -i -X GET \  
  -H 'token: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiIjoic2VjdXJlLWFwcCIsInN1YiI6InVzZXJuYW1lIiwiaXhwIjoxNTYzNTU5MDExLCJyb2wiOiUk9MRV9VU0VSIl19.QmLO3ahgNtbxt0l02k7krXWLD5Y4Ff3uDwxZv9vudisQjIVAQCxe65HqrR6MR7IsKU_4EgDx94PS6UDS0oyIw' \  
  -H 'password: test' \  
  -H 'Accept: application/octet-stream' \  
  -H 'user: test'
```

4.2.4. Example success response

```
HTTP/1.1 200 OK  
Pragma: no-cache  
X-XSS-Protection: 1; mode=block  
Expires: 0  
X-Frame-Options: DENY  
X-Content-Type-Options: nosniff  
Cache-Control: no-cache, no-store, max-age=0, must-revalidate  
Content-Length: 12  
  
file content
```

Response body

```
file content
```

4.3. Write Document

4.3.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username
password	datasafe user's password

4.3.2. Path fields

Table 3. /document/{path}

Parameter	Description
path	path to the file

4.3.3. Example request

```
$ curl 'http://example.com/datasafe/document/path/to/file' -i -X PUT \  
  -H 'token: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZWN1cmUtYXBpIiwiaXVkiIjoic2VjdXJlLWFwcCIsInN1YiI6InVzZXJuYW1lIiwiaXhwIjojNTYzNTU5MDExLCJyb2wiOiJsiUk9MRV9VU0VSIl19.QmLO3ahgNtbxt0l02k7krXWLD5Y4Ff3uDwxZv9vudisQjIVAQCxe65HqrR6MR7IsKU_4EgDx94PS6UDS0oyIw' \  
  -H 'Content-Type: application/octet-stream; charset=UTF-8' \  
  -H 'password: test' \  
  -H 'user: test' \  
  -d 'file content'
```

4.3.4. Example success response

```
HTTP/1.1 200 OK  
Pragma: no-cache  
X-XSS-Protection: 1; mode=block  
Expires: 0  
X-Frame-Options: DENY  
X-Content-Type-Options: nosniff  
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

4.4. Remove Document

4.4.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username
password	datasafe user's password

4.4.2. Path fields

Table 4. /document/{path}

Parameter	Description
path	path to the file

4.4.3. Example request

```
$ curl 'http://example.com/datasafe/document/path/to/file' -i -X DELETE \
-H 'token: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW50cmUtYXBpIiwiaXVkiOiJoic2VjdXJlLWFWc
CIsInN1YiI6InVzZXJuYW1lIiwiaXhwIjoxNTYzNTU5MDExLCJyb2wiOiUk9MRV9VU0VSIl19.QmLO3ahgNt
bxt0l02k7krXWLD5Y4Ff3uDwxZv9vudisQjIVAQCxe65HqrR6MR7IsKU_4EgDx94PS6UDS0oyIw' \
-H 'password: test' \
-H 'user: test'
```

4.4.4. Example success response

```
HTTP/1.1 200 OK
Pragma: no-cache
X-XSS-Protection: 1; mode=block
Expires: 0
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

5. Inbox API

5.1. List Inbox

5.1.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username

Name	Description
password	datasafe user's password

5.1.2. Example request

```
$ curl 'http://example.com/datasafe/inbox/' -i -X GET \
  -H 'password: test' \
  -H 'token: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiIjoic2VjdXJlLWFwcCIiInN1YiI6InVzZXJyZW1lIiwiaXhwIjojNTYzNTU5MDE0LCJyb2wiOiUk9MRV9VU0VSIl19.WSf5MKNBL-
i6dt49nB_pc_JM7prL8jgSc2Ep8fRKQ2DetVR2jNqOP_wwVV-7YD9NC0R3LYIXgQlu-2g9LBaAlg' \
  -H 'user: test'
```

5.1.3. Example success response

```
HTTP/1.1 200 OK
Pragma: no-cache
Content-Length: 21
X-XSS-Protection: 1; mode=block
Expires: 0
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Content-Type: application/json; charset=UTF-8
Cache-Control: no-cache, no-store, max-age=0, must-revalidate

["/path/to/file.txt"]
```

Response body

```
["/path/to/file.txt"]
```

5.2. Read from Inbox

5.2.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username
password	datasafe user's password

5.2.2. Path fields

Table 5. /inbox/{path}

Parameter	Description
path	path to the file

5.2.3. Example request

```
$ curl 'http://example.com/datasafe/inbox/test.txt' -i -X GET \  
  -H 'password: test' \  
  -H 'Accept: application/octet-stream' \  
  -H 'token: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBPiwiYXVkJjoic2VjdXJLLWFwcCIsInN1YiI6InVzZXJuYW1lIiwiaXhwIjoxNTYzNTU5MDE0LCJyb2wiOiUk9MRV9VU0VSIl19.WSf5MKNBL-i6dt49nB_pc_JM7prL8jgSc2Ep8fRKQ2DetVR2jNqOP_wwVV-7YD9NC0R3LYIXgQlu-2g9LBaAlg' \  
  -H 'user: test'
```

5.2.4. Example success response

```
HTTP/1.1 200 OK  
Pragma: no-cache  
X-XSS-Protection: 1; mode=block  
Expires: 0  
X-Frame-Options: DENY  
X-Content-Type-Options: nosniff  
Content-Length: 5  
Cache-Control: no-cache, no-store, max-age=0, must-revalidate  
  
hello
```

Response body

```
hello
```

5.3. Write to Inbox

5.3.1. Request header

Name	Description
token	Bearer authentication token is required
users	recipients array

5.3.2. Path fields

Table 6. /inbox/{path}

Parameter	Description
path	path to the file

5.3.3. Example request

```
$ curl 'http://example.com/datasafe/inbox/test.txt' -i -X PUT \  
  -H 'Content-Type: application/octet-stream; charset=UTF-8' \  
  -H 'token: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW11cmUtYXBpIiwiaXVkiOiJoic2VjdXJlLWFwcCIsInN1YiOi6InVzZXJuYV11IiwiaXhwIjoxNTYzNTU5MDE0LCJyb2wiOiUk9MRV9VU0VSIl19.WSf5MKNBL-i6dt49nB_pc_JM7prL8jgSc2Ep8fRKQ2DetVR2jNqOP_wwVV-7YD9NC0R3LYIXgQlu-2g9LBaAlg' \  
  -H 'users: test' \  
  -d 'file content'
```

5.3.4. Example success response

```
HTTP/1.1 200 OK  
Pragma: no-cache  
X-XSS-Protection: 1; mode=block  
Expires: 0  
X-Frame-Options: DENY  
X-Content-Type-Options: nosniff  
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

5.4. Remove from Inbox

5.4.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username
password	datasafe user's password

5.4.2. Path fields

Table 7. /inbox/{path}

Parameter	Description
path	path to the file

5.4.3. Example request

```
$ curl 'http://example.com/datasafe/inbox/test.txt' -i -X DELETE \  
  -H 'password: test' \  
  -H 'token: Bearer  
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiIjoic2VjdXJLLWFwc  
CIsInN1YiI6InVzZXJuYW1lIiwiaXhwIjoxNTYzNTU5MDE0LCJyb2wiOiUk9MRV9VU0VSIl19.WSf5MKNBL-  
i6dt49nB_pc_JM7prL8jgSc2Ep8fRKQ2DetVR2jNqOP_wwVV-7YD9NC0R3LYIXgQlu-2g9LBaAlg' \  
  -H 'user: test'
```

5.4.4. Example success response

```
HTTP/1.1 200 OK  
Pragma: no-cache  
X-XSS-Protection: 1; mode=block  
Expires: 0  
X-Frame-Options: DENY  
X-Content-Type-Options: nosniff  
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

6. Versioned Document API

6.1. List Documents

6.1.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username
password	datasafe user's password

6.1.2. Example request

```
$ curl 'http://example.com/datasafe/versioned/' -i -X GET \  
  -H 'password: test' \  
  -H 'user: test' \  
  -H 'Content-Type: application/json;charset=UTF-8' \  
  -H 'token: Bearer  
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiIjoic2VjdXJLLWFwc  
CIsInN1YiI6InVzZXJuYW1lIiwiaXhwIjoxNTYzNTU5MDA0LCJyb2wiOiUk9MRV9VU0VSIl19.FBsYqnTlKE  
axj8DrldkBTU-ytHiPpor2T5v0L2HqLnt7b8L-v0M7I7srGPcEbA5mykcFmakiDqpQT_ww_2qwfa'
```

6.1.3. Example success response

```
HTTP/1.1 200 OK
Pragma: no-cache
Content-Length: 21
X-XSS-Protection: 1; mode=block
Expires: 0
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Content-Type: application/json;charset=UTF-8
Cache-Control: no-cache, no-store, max-age=0, must-revalidate

["/path/to/file.txt"]
```

Response body

```
["/path/to/file.txt"]
```

6.2. Read Document

6.2.1. Request header

Name	Description
<code>token</code>	Bearer authentication token is required
<code>user</code>	datasafe username
<code>password</code>	datasafe user's password

6.2.2. Path fields

Table 8. `/versioned/{path}`

Parameter	Description
<code>path</code>	path to the file

6.2.3. Example request

```
$ curl 'http://example.com/datasafe/versioned/path/to/file' -i -X GET \
-H 'password: test' \
-H 'Accept: application/octet-stream' \
-H 'user: test' \
-H 'token: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiIjoic2VjdXJlLWFwcCIsInN1YiI6InVzZXJuYW1lIiwiaXhwIjoxNTYzNTU5MDA0LCJyb2wiOiUk9MRV9VU0VSIl19.FBsYqnTlKEaxj8DrldkBTU-ytHiPpor2T5v0L2HqLnt7b8L-v0M7I7srGPcEbA5mykcFmakiDqpQT_wv_2qwfa'
```

6.2.4. Example success response

```
HTTP/1.1 200 OK
Pragma: no-cache
X-XSS-Protection: 1; mode=block
Expires: 0
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Content-Length: 5
Cache-Control: no-cache, no-store, max-age=0, must-revalidate

hello
```

Response body

```
hello
```

6.3. Write Document

6.3.1. Request header

Name	Description
<code>token</code>	Bearer authentication token is required
<code>user</code>	datasafe username
<code>password</code>	datasafe user's password

6.3.2. Path fields

Table 9. `/versioned/{path}`

Parameter	Description
<code>path</code>	path to the file

6.3.3. Example request

```
$ curl 'http://example.com/datasafe/versioned/path/to/file' -i -X PUT \
  -H 'Content-Type: application/octet-stream;charset=UTF-8' \
  -H 'password: test' \
  -H 'user: test' \
  -H 'token: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiJjoic2VjdXJlLWFwc
CIsInN1YiOi6InVzZXJuYW1lIiwiaXhwIjoxNTYzNTU5MDA0LCJyb2wiOiUk9MRV9VU0VSIl19.FBsYqnTlKE
axj8DrldkBTU-ytHiPpor2T5v0l2HqLnt7b8L-v0M7I7srGPcEbA5mykcFmakiDqpQT_wv_2qwfa' \
  -d 'file content'
```

6.3.4. Example success response

```
HTTP/1.1 200 OK
Pragma: no-cache
X-XSS-Protection: 1; mode=block
Expires: 0
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```

6.4. Remove Document

6.4.1. Request header

Name	Description
token	Bearer authentication token is required
user	datasafe username
password	datasafe user's password

6.4.2. Path fields

Table 10. */versioned/{path}*

Parameter	Description
path	path to the file

6.4.3. Example request

```
$ curl 'http://example.com/datasafe/versioned/path/to/file' -i -X DELETE \  
-H 'password: test' \  
-H 'user: test' \  
-H 'token: Bearer  
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW1cmUtYXBpIiwiaXVkiJoic2VjdXJLLWFwc  
CIsInN1YiI6InVzZXJyZW1lIiwiaXhwIjoxNTYzNTU5MDA0LCJyb2wiOiUk9MRV9VU0VSIl19.FBsYqnTlKE  
axj8DrldkBTU-ytHiPpor2T5v0l2HqLnt7b8L-v0M7I7srGPcEbA5mykcFmakiDqpQT_wv_2qwFA'
```

6.4.4. Example success response

```
HTTP/1.1 200 OK  
Pragma: no-cache  
X-XSS-Protection: 1; mode=block  
Expires: 0  
X-Frame-Options: DENY  
X-Content-Type-Options: nosniff  
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
```