

Privacy Crash Cam - API Documentation

Jan Wittler

August 23, 2019

Chapter 1

API Globals

1.1 Session information

Almost all API requests require a session token. This token must be sent as a HTTP header for key `token`. The token is received when performing the login request (c.f. section 2.1). Example:

`token : 155e05dc-f924-4d2e-8ce5-c0a085301fdf`

1.2 Status messages

The original implementation returned for most requests HTTP status code 200 with some status messages in the HTTP body to communicate the request's result, instead of using different HTTP status codes. To reduce overhead when adjusting Android client and web interface, the refactored API keeps this behaviour wherever needed. The used status codes are listed below:

| | |
|-------------------|---|
| SUCCESS | Request succeeded |
| WRONG_PASSWORD | Provided password was invalid |
| NOT_AUTHENTICATED | Provided authentication token was invalid |
| FAILURE | Request failed due to some other error |

Chapter 2

User Management

2.1 Login

| | |
|----------------|---|
| Synopsis: | POST <code>/webservice/account/login</code> Logs in the user with the given credentials |
| Precondition: | Accept: <code>application/x-www-form-urlencoded</code> |
| Parameters: | mail User's mail password User's password |
| Postcondition: | Content-Type: <code>text/plain</code> Cookie: <i>token</i> : <code>:session-token</code> |
| Status code: | 200 OK 401 Unauthorized |

2.2 Create User

| | |
|----------------|---|
| Synopsis: | POST <code>/webservice/account/create</code> Creates a new user from the specified credentials |
| Precondition: | Accept: <code>application/x-www-form-urlencoded</code> |
| Parameters: | mail User's mail. Must not be used for some user account already. password User's password |
| Postcondition: | Content-Type: <code>text/plain</code> |
| Status code: | SUCCESS FAILURE |

2.3 Delete User

| | |
|----------------|---|
| Synopsis: | POST /webservice/account/delete Deletes the user's account and all associated data |
| Precondition: | Token: :session-token |
| Parameters: | - |
| Postcondition: | Content-Type: text/plain |
| Status code: | SUCCESS NOT_AUTHENTICATED |

Chapter 3

Video Management

3.1 Get Video Ids

| | |
|----------------|---|
| Synopsis: | GET <code>/webservice/videos</code> Returns all video ids for the current user |
| Precondition: | Token: <code>:session-token</code> |
| Parameters: | - |
| Postcondition: | Content-Type: <code>application/json</code> HTTP Body: JSON-Array of video ids |
| Status code: | 200 OK NOT_AUTHENTICATED |

3.2 Get Video File

| | |
|----------------|--|
| Synopsis: | GET <code>/webservice/videos/:id</code> Returns the video file for the specified id |
| Precondition: | Token: <code>:session-token</code> |
| Parameters: | - |
| Postcondition: | Content-Type: <code>application/octet-stream</code> |
| Status code: | 200 OK 404 Not Found |

3.3 Get Metadata File

| | |
|----------------|--|
| Synopsis: | GET /webservice/videos/metadata/:id Returns the metadata file for the video with the specified id |
| Precondition: | Token: :session-token |
| Parameters: | - |
| Postcondition: | Content-Type: application/octet-stream |
| Status code: | 200 OK 404 Not Found |

3.4 Delete Video

| | |
|----------------|---|
| Synopsis: | DELETE /webservice/videos/:id Deletes all content associated with the specified video id |
| Precondition: | Token: :session-token |
| Parameters: | - |
| Postcondition: | Content-Type: text/plain |
| Status code: | SUCCESS |

3.5 Upload Video and Metadata

| | |
|----------------|--|
| Synopsis: | POST /webservice/videos Takes an encrypted video file, an encrypted metadata file, and an encrypted key file, decrypts all and stores them on the server. The video and metadata files must be encrypted with the uploaded key. The key itself must be encrypted with the server's public key (c.f. section 4.1). |
| Precondition: | Content-Type: multipart/form-data Token: :session-token |
| Parameters: | video Encrypted video file metadata Encrypted metadata file key Encrypted key file |
| Postcondition: | Content-Type: text/plain |
| Status code: | SUCCESS FAILURE |

Chapter 4

Public Key Provider

4.1 Get Public Key

| | |
|----------------|--|
| Synopsis: | GET /webservice/publicKey Returns the server's public key. This key must be used to encrypt keys uploaded to the server (c.f. section 3.5). |
| Precondition: | - |
| Parameters: | - |
| Postcondition: | Content-Type: text/plain HTTP Body: server's public key |
| Status code: | 200 OK |
