

DOJ CAPSS Bulk Upload API

Overview

The CAPSS system has expanded the ability to upload an XML file containing multiple Property Transactions via a JSON web service. This is an optional service to help with automating the upload process for those interested and is not mandatory. The existing web based upload bulk import and online form will still be available.

Uploading Property Transactions requires an account for the user to be established. Uploading transactions require limited access to the CAPSS system, which can be controlled by the account admin. Third party clients are registered for an account by the CAPSS admin. An authorized client will request an access token prior to interacting with the system. The token is used as credentials to upload and check status of Property Transactions.

OAuth2 provides a mechanism for resource owners to delegate secure access to resources for third-party client applications. OAuth2 is an open standard and thoroughly documented. OAuth2 manages client registration, token generation, and resource authentication.

OAuth2

The primary resource for OAuth2 documentation is the OAuth2 Specification: [The OAuth 2.0 Authorization Framework](#). CAPSS only implemented a subset of the specification needed to provide basic CAPSS requirements. Additional features may be added as required.

Registration

Registration Clients are registered in the CAPSS system by the CAPSS owner. Owners will grant clients access to the system in one of several methods. OAuth2 implements 4 methods to grant access to resources. Each method provides a different mechanism that is useful in different context.

Currently CAPSS supports the following grant:

Client Credentials Grant is used for machine-to-machine interfaces and is expected to be the preferred method by most clients. The CAPSS owner registers the client in CAPSS. A client id and client secret (username and password) is given to the third party clients. The client sends an http request to CAPSS with credentials and a response is returned with the token. The token is used in subsequent requests to access CAPSS resources (upload and status endpoints).

Resources

The resources are protected URI's within CAPSS. The CAPSS system allows registered clients limited access. Currently clients can Upload Property Transactions

and Check the status of an Upload. The resources are available as web services and require credentials in the form of OAuth2 tokens.

Upload & Status

Property Transactions are uploaded via an XML document. URL Parameters are used to dictate how processing should be accomplished in the case of errors or duplicate transactions.

A limited check is performed during upload and the response will return the status of the upload. A successful upload will contain the submission id used to identify the upload along with the URL for checking the status. Errors will contain the errors encountered.

A status check URL is provided to identify if uploaded transaction were successfully processed into CAPSS. HTTP status codes will identify if the transactions are being processed or are completed. If the processing code is presented, the application will try again after a period of time until the complete status code is returned, or until a timeout is reached in the case of an application error. When successful, the response will contain the status code and the URL to resubmit. Uploads with errors will additionally provide a list of errors for transactions.

The upload status response contains a list of errors for each store if present. To facilitate automated submissions, error handling is provided via a duplicate and error parameter that defines the actions to take under those conditions. Please see documentation for details.

The nature of web services is that it allows the client to be implemented in any manner the client deems appropriate. Requests can be handcrafted with tools such as curl or with browser plug-ins (Postman and RESTClient), which can facilitate the testing process. However, a more automated process is expected to be implemented.

The requests and responses are implemented with JSON web services and utilize HATEOAS to provide additional URLs to the client to keep the API flexible. HATEOAS is a architectural style that presents the user with valid urls for subsequent requests. When the user submits a valid Bulk Upload, the response will provide the link to view the status of that file. When checking the status of an upload, the response will continue to contain the link of the status while it is being processed. Once completed, the save link will be presented.

Creating a Client ID

The CAPSS admin will log into CAPSS Main application (<https://capss.doj.ca.gov>), navigate to API Client and select New Client. On the Create Client page, they will enter the values in all the fields and press Create button. They will then provide the user with the credentials.

For Sandbox environment, CAPSS admin will need to access (<https://capss-sb.doj.ca.gov>)

Sample:

```
clientId = apitest  
clientSecret = Password1
```

Client Credentials Grant

Request Grant by using credentials to retrieve a token. The user will use the credentials provided by the CAPSS Admin for a token.

Client Request

```
curl -k -X POST -u clientId:clientSecret -d "scope=api" -d  
"grant_type=client_credentials"  
https://capss.doj.ca.gov/oauth/token
```

Client Request <for Sandbox environment>

```
curl -k -X POST -u clientId:clientSecret -d "scope=api" -  
d "grant_type=client_credentials" https://capss-  
sb.doj.ca.gov/oauth/token
```

Example: User can run the following command in their Windows Command Prompt

```
curl -k -X POST -u apitest:Password1 -d "scope=api" -d  
"grant_type=client_credentials"  
https://capss.doj.ca.gov/oauth/token
```

CAPSS Response

```
{"access_token": "4e36b4e8-81ae-4d2d-bfac-  
cd24dfa24ba8", "token_type": "bearer", "expires_in": 99999998, "scope": "a  
pi"}
```

Bulk Upload

The bulk upload process uploads a file containing Property Transaction files. The CAPSS application provides a “save” resource to submit bulk uploads to. The response contains the URL to check the status.

Client Request

```
curl -k -X POST -H "Authorization: Bearer token" -F
"bulkUploadFile=pathToYourXmlFile.xml"
https://capss.doj.ca.gov/api/bulkupload/save
```

Client Request <for Sandbox environment>

```
curl -k -X POST -H "Authorization: Bearer token" -F
"bulkUploadFile=pathToYourXmlFile.xml" https://capss-
sb.doj.ca.gov/api/bulkupload/save
```

Example: User can run the following command in their Windows Command Prompt

```
curl -k -X POST -H "Authorization: Bearer 4e36b4e8-81ae-4d2d-bfac-
cd24dfa24ba8" -F
"bulkUploadFile=@C:/Users/Desktop/api/APIBulkUpload.xml"
https://capss.doj.ca.gov/api/bulkupload/save
```

CAPSS Response

HTTP Status code 202

```
{"licenseNumber":"000000000","status":"new","submission":{"submissio
nId":"c663ec64-29f7-4a31-b246-
654a792c35dc"},"links":{"href":"https://capss.doj.ca.gov/api/bulkupl
oad/show/53820960","rel":"show"}}
```

Bulk Upload Status Check

The status of the Upload can be checked by requesting the show resource endpoint with the given id for the upload. The URL should not be stored and created by the client as it is possible for it to change over time. The responses provides the URLs to navigate to get the status and should be retrieved from the links property in the response.

Client Request

```
curl -k -X GET -H "Authorization: Bearer token" hrefURL
```

Example: User can run the following command in their Windows Command Prompt

```
curl -k -X GET -H "Authorization: Bearer 4e36b4e8-81ae-4d2d-bfac-cd24dfa24ba8" https://capss.doj.ca.gov/api/bulkupload/show/53820960
```

CAPSS Response

HTTP Status Code 200

```
{"licenseNumber":"000000000","status":"complete","statistics":{"totalBulks":1,"totalTransactions":1,"totalItems":2,"savedTransactions":1,"savedItems":2,"errorItems":0},"stores":[],"links":{"href":"https://capss.doj.ca.gov/api/bulkupload/save","rel":"new"},"statusMessage":"Successfully uploaded file. File APIBulkUpload.xml uploaded 1 record(s) at Wed Mar 02 09:36:08 PST 2022."}
```

Bulk Update

The bulk update process uploads a file containing Serial Number(s) and/or Owner Applied Number(s) for updates on articles categorized as an electronic handheld device. The CAPSS application provides a “save” resource to submit item updates to. The response contains the URL to check the status.

Client Request

```
curl -k -X POST -H "Authorization: Bearer token" -F "itemUpdateFile=pathToYourXmlFile.xml" https://capss.doj.ca.gov/api/itemupdate/updateItem
```

Client Request <for Sandbox environment>

```
curl -k -X POST -H "Authorization: Bearer token" -F "itemUpdateFile=pathToYourXmlFile.xml" https://capss-sb.doj.ca.gov/api/itemupdate/updateItem
```

Example: User can run the following command in their Windows Command Prompt

```
curl -k -X POST -H "Authorization: Bearer 4e36b4e8-81ae-4d2d-bfac-cd24dfa24ba8" -F "itemUpdateFile=@C:/Users/enduser/Desktop/APIItemUpdate.xml" https://capss.doj.ca.gov/api/itemupdate/updateItem
```

CAPSS Response

HTTP Status code 202

```
{"licenseNumber":"000000000","status":"new","submission":{"submissionId":"cc08e195-d839-40a9-89ee-d8b0c3ed9b08"},"links":{"href":"https://capss.doj.ca.gov/api/itemupdate/show/53820964","rel":"show"}}
```

Bulk Update Status Check

The status of the update can be checked by requesting the show resource endpoint with the given id for the update. The URL should not be stored and created by the client as it is possible for it to change over time. The responses provides the URLs to navigate to get the status and should be retrieved from the links property in the response.

Client Request

```
curl -k -X GET -H "Authorization: Bearer token" hrefURL
```

Example: User can run the following command in their Windows Command Prompt

```
curl -k -X GET -H "Authorization: Bearer 4e36b4e8-81ae-4d2d-bfac-cd24dfa24ba8" https://capss.doj.ca.gov/api/itemupdate/show/53820964
```

CAPSS Response

HTTP Status Code 200

```
{"licenseNumber":"000000000","status":"complete","statistics":{"totalItems":1,"savedItems":1,"errorItems":0,"duplicateItems":0},"links":{"href":"https://capss.doj.ca.gov/api/itemupdate/updateitem","rel":"new"},"statusMessage":"Successfully updated Property Transaction Items. File APIItemUpdate.xml uploaded 1 record(s) at Wed Mar 02 09:40:03 PST 2022."}
```

URL End Points

CAPSS URL	Description	Response	Method
oauth/token	OAuth2 Token Request. Used to get a new token from Client Credentials.	A JSON object containing the Token.	POST

api/bulkupload/save	CAPSS endpoint for submitting Bulk Uploads	A JSON response representing the result of the upload. Can contain errors, and the link to the status check url if successfully processing.	POST
api/itemupdate/updateItem	CAPSS endpoint for submitting Bulk Updates	A JSON response representing the result of the update. Can contain errors, and the link to the status check url if successfully processing.	POST

Request Parameters

Parameters	Where Used	Values	Notes
grant_type	OAuth2 Grant/Token requests	client_credentials	One or the other is used during code grants
<u>clientId</u>	OAuth2 Grant/Token requests	Client Username	The username supplied via CAPSS authorized user
clientSecret	OAuth2 Grant/Token requests	Client password	The password supplied via CAPSS authorized user
scope	OAuth2 Grant/Token requests	api	Used to identify what resources the client can access.
onDup	CAPSS BulkUpload Submission	cancel submitDups submitWithoutDup	The action to take when encountering dups in a transaction
onError	CAPSS BulkUpload Submission	cancel submit	The action to take when encountering errors in a transaction

Status Codes

Codes	Bulk Upload Save	Bulk Upload Status
200 OK	N/A	Request complete successfully. Processing is complete
202 Accepted	Bulk Import was accepted for processing	Bulk Import was accepted for processing. Processing is still in progress
400 Bad Request	A problem with the request. Check parameters and file being uploads	N/A
404 Not Found	N/A	Requested Bulk Import not in system
500 Internal Server Error	Error submitting file.	Error Processing file.

Glossary of Terms

Below are some terms used in this document and OAUTH2 spec as they apply to CAPSS. These definitions are summaries, more detailed and abstract explanations can be found in the OAuth2 spec.

Access Token: A code representing user credentials used to access protected resources.

Authorization Server: The server that controls OAuth2 token management and client registration.

Bulk Upload: An abstraction resource representing the uploaded file containing Property Transactions.

Bulk Update: An abstraction resource representing the uploaded file containing Serial Number(s) and/or Owner Applied Numbers for updates on certain property transaction items.

Client: The third-party granted access from the Resource Owner. The client may be a third-party organization submitting on behalf of the Resource Owner, or may be an application the Resource Owner uses to upload transactions to system.

Curl: a tool/library for transferring data via HTTP and various other protocols.

Grant: The OAuth2 mechanism for giving (granting) authorization to a third party client.

Grant Process: The set of 4 methods defined by OAuth2 to grant authorization to a third party.

HATEOAS (Hypermedia as the Engine of Application State): A constraint of the REST application architecture. A hypermedia-driven site provides information to navigate the site's REST interfaces dynamically by including hypermedia links with the responses

OAuth2: A protocol for providing specific authorization flows for web-applications. Used in CAPSS to delegate authorization to third party clients for uploading transactions

Resource: A key abstraction of information that can be a document or image, a service, an object, or a collection of other resources. Resources are exposed as HTTP endpoints and comprise of the Bulk Upload in CAPSS

Resource Owner: The CAPSS User with permissions to grant third party applications access to system via OAuth2.

Resource Server: The CAPSS server containing the delegated resource. Includes the endpoints used to upload and check the status of an upload.

ReST (Representation State Transfer): A stateless communication protocol implemented over HTTPS for the CAPSS Application.

Status Codes: Meta-information about the status of an HTTP request.