

CKAN

Overview

The purpose of this document is to describe a simple set of test cases for CKAN in order to determine viability for integration with Clowder.

- [Overview](#)
- [Orientation](#)
- [Retrieve Your API Key](#)
- [Create a Dataset](#)
- [Updating or Modifying a Dataset](#)
- [Creating or Uploading a New Resource to a Dataset](#)

Orientation

Familiarize yourself with the CKAN API by playing around with a few examples, like these: <https://docs.ckan.org/en/ckan-1.7.4/api-tutorial.html>

NOTE: You can use **demo.ckan.org** instead of **test.ckan.org** in the examples above.

Retrieve Your API Key

First, we'll need to add an API Key. To do this, you'll need to follow a few simple steps:

1. Navigate to <https://demo.ckan.org>
2. Register and/or login to CKAN
3. At the top-right, click on the button containing **Your Name**
4. At the bottom-left, your current API key should be listed

This should regenerate your existing API key, allowing you to interact with the CKAN API in an authorized fashion.

NOTE: This is currently causing an error on the CKAN Demo instance - Internal Server Error

Create a Dataset

Using the API key you just created above, create a dataset in CKAN:

```

$ curl https://demo.ckan.org/api/rest/dataset --header 'Content-Type: application/json' -d '{"name":"dataset-has-a-tail", "title":"OMG its a dataset"}' -H "Authorization:<API_KEY>" -vvv
* Trying 104.24.114.210...
* TCP_NODELAY set
* Connected to demo.ckan.org (104.24.114.210) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/ssl/cert.pem
*   CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-ECDSA-CHACHA20-POLY1305
* ALPN, server accepted to use h2
* Server certificate:
*   subject: OU=Domain Control Validated; OU=PositiveSSL Multi-Domain; CN=sn1113313.cloudflaressl.com
*   start date: Aug 11 00:00:00 2018 GMT
*   expire date: Feb 17 23:59:59 2019 GMT
*   subjectAltName: host "demo.ckan.org" matched cert's "*.ckan.org"
*   issuer: C=GB; ST=Greater Manchester; L=Salford; O=COMODO CA Limited; CN=COMODO ECC Domain Validation Secure Server CA 2
*   SSL certificate verify ok.
* Using HTTP2, server supports multi-use
* Connection state changed (HTTP/2 confirmed)
* Copying HTTP/2 data in stream buffer to connection buffer after upgrade: len=0
* Using Stream ID: 1 (easy handle 0x7f97b9005800)
> POST /api/rest/dataset HTTP/2
> Host: demo.ckan.org
> User-Agent: curl/7.54.0
> Accept: /*/*
> Content-Type: application/json
> Authorization:<API_KEY>
> Content-Length: 58
>
* Connection state changed (MAX_CONCURRENT_STREAMS updated)!
* We are completely uploaded and fine
< HTTP/2 201
< date: Mon, 13 Aug 2018 18:00:49 GMT
< content-type: application/json; charset=utf-8
< set-cookie: __cfduid=<COOKIE>; expires=Tue, 13-Aug-19 18:00:48 GMT; path=/; domain=.ckan.org; HttpOnly
< cache-control: no-cache
< location: https://demo.ckan.org/api/rest/dataset/a09f9319-bf4c-48a0-8d98-a8d066d6636d
< pragma: no-cache
< vary: Accept-Encoding
< expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
< server: cloudflare
< cf-ray: 449d1555af3055be-ORD
<
* Connection #0 to host demo.ckan.org left intact
{"license_title": null, "maintainer": null, "private": false, "maintainer_email": null, "num_tags": 0, "id": "a09f9319-bf4c-48a0-8d98-a8d066d6636d", "metadata_created": "2018-08-13T18:00:48.853260", "relationships": [], "license": null, "metadata_modified": "2018-08-13T18:00:48.853275", "author": null, "author_email": null, "state": "active", "version": null, "creator_user_id": "28fb4558-3686-4b24-96d9-8d935190d962", "type": "dataset", "resources": [], "num_resources": 0, "tags": [], "groups": [], "license_id": null, "organization": null, "name": "dataset-has-a-tail", "isopen": false, "notes_rendered": "", "url": null, "ckan_url": "https://demo.ckan.org/dataset/dataset-has-a-tail", "notes": null, "owner_org": null, "ratings_average": null, "extras": {}, "ratings_count": 0, "title": "OMG its a dataset", "revision_id": "eaf48c90-b219-40a0-912a-00e7a5313697"}

```

Hooray! Your dataset has been created!

Updating or Modifying a Dataset

Simple modifications can occur with a PUT:

```

$ curl https://demo.ckan.org/api/rest/dataset/dataset-has-a-tail --header 'Content-Type: application/json' -
d'{"name":"dataset-has-a-tail", "title":"OMG its a dataset with a tail"}' -H "Authorization:<API_KEY>" -vvv
* Trying 104.24.115.210...
* TCP_NODELAY set
* Connected to demo.ckan.org (104.24.115.210) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/ssl/cert.pem
*   CAspace: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-ECDSA-CHACHA20-POLY1305
* ALPN, server accepted to use h2
* Server certificate:
*   subject: OU=Domain Control Validated; OU=PositiveSSL Multi-Domain; CN=sn1113313.cloudflaressl.com
*   start date: Aug 11 00:00:00 2018 GMT
*   expire date: Feb 17 23:59:59 2019 GMT
*   subjectAltName: host "demo.ckan.org" matched cert's "*.ckan.org"
*   issuer: C=GB; ST=Greater Manchester; L=Salford; O=COMODO CA Limited; CN=COMODO ECC Domain Validation Secure
Server CA 2
* SSL certificate verify ok.
* Using HTTP2, server supports multi-use
* Connection state changed (HTTP/2 confirmed)
* Copying HTTP/2 data in stream buffer to connection buffer after upgrade: len=0
* Using Stream ID: 1 (easy handle 0x7fca76800400)
> POST /api/rest/dataset/dataset-has-a-tail HTTP/2
> Host: demo.ckan.org
> User-Agent: curl/7.54.0
> Accept: */*
> Content-Type: application/json
> Authorization:<API_KEY>
> Content-Length: 70
>
* Connection state changed (MAX_CONCURRENT_STREAMS updated)!
* We are completely uploaded and fine
< HTTP/2 200
< date: Mon, 13 Aug 2018 22:03:32 GMT
< content-type: application/json; charset=utf-8
< set-cookie: __cfduid=<COOKIE>; expires=Tue, 13-Aug-19 22:03:31 GMT; path=/; domain=.ckan.org; HttpOnly
< cache-control: no-cache
< pragma: no-cache
< vary: Accept-Encoding
< expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
< server: cloudflare
< cf-ray: 449e78e3af13560c-ORD
<
* Connection #0 to host demo.ckan.org left intact
{"license_title": null, "maintainer": null, "private": false, "maintainer_email": null, "num_tags": 0, "id":
"a09f9319-bf4c-48a0-8d98-a8d066d6636d", "metadata_created": "2018-08-13T18:00:48.853260", "relationships": [],
"license": null, "metadata_modified": "2018-08-13T22:03:32.287239", "author": null, "author_email": null,
"state": "active", "version": null, "creator_user_id": "28fb4558-3686-4b24-96d9-8d935190d962", "type":
"dataset", "resources": [], "num_resources": 0, "tags": [], "groups": [], "license_id": null, "organization":
null, "name": "dataset-has-a-tail", "isopen": false, "notes_rendered": "", "url": null, "ckan_url":
"https://demo.ckan.org/dataset/dataset-has-a-tail", "notes": null, "owner_org": null, "ratings_average": null,
"extras": {}, "ratings_count": 0, "title": "OMG its a dataset with a tail", "revision_id": "1f36361a-a9cf-498d-
94e6-5af2431c98cd"}

```

We can do the same thing to add arbitrary key-value pairs into the **"extras"** field. For example, if we wanted to stringify our extracted JSON metadata and attach it to a CKAN dataset:

[illegible]

```

"metadata_modified": "2018-08-13T22:19:02.606453",
"author": null,
"author_email": null,
"state": "active",
"version": null,
"creator_user_id": "28fb4558-3686-4b24-96d9-8d935190d962",
"type": "dataset",
"resources": [],
"num_resources": 0,
"tags": [],
"groups": [],
"license_id": null,
"organization": null,
"name": "dataset-has-a-tail",
"isopen": false,
"notes_rendered": "",
"url": null,
"ckan_url": "https://demo.ckan.org/dataset/dataset-has-a-tail",
"notes": null,
"owner_org": null,
"ratings_average": null,
"extras": {
  "Metadata": "{\n  \"apiVersion\": \"1.0.0\",\n  \"swaggerVersion\": \"1.2\",\n  \"basePath\": \"http://petstore.swagger.io/api\",\n  \"resourcePath\": \"\\\"/store\\\"\",\n  \"produces\": [\n    \"application/json\",\n    \"application/xml\"\n  ],\n  \"apis\": {\n    \"path\": \"\\\"/store/order/{orderId}\\\"\",\n    \"operations\": {\n      \"method\": \"GET\",\n      \"summary\": \"Find purchase order by ID\",\n      \"notes\": \"For valid response try integer IDs with value <= 5. Anything above 5 or nonintegers will generate API errors\",\n      \"type\": \"Order\",\n      \"nickname\": \"getOrderById\",\n      \"authorizations\": {\n        \"name\": \"orderId\"\n      },\n      \"description\": \"ID of pet that needs to be fetched\",\n      \"required\": true,\n      \"type\": \"string\",\n      \"paramType\": \"path\",\n      \"responseMessages\": {\n        \"code\": 400,\n        \"message\": \"Invalid ID supplied\"\n      },\n      \"code\": 404\n    },\n    \"code\": 404\n  },\n  \"message\": \"Order not found\",\n  \"method\": \"DELETE\",\n  \"summary\": \"Delete purchase order by ID\",\n  \"notes\": \"For valid response try integer IDs with value < 1000. Anything above 1000 or nonintegers will generate API errors\",\n  \"type\": \"void\",\n  \"nickname\": \"deleteOrder\",\n  \"authorizations\": {\n    \"scope\": \"test:anything\"\n  },\n  \"description\": \"anything\",\n  \"parameters\": {\n    \"name\": \"orderId\"\n  },\n  \"description\": \"ID of the order that needs to be deleted\",\n  \"required\": true,\n  \"type\": \"string\",\n  \"paramType\": \"path\",\n  \"responseMessages\": {\n    \"code\": 400,\n    \"message\": \"Invalid ID supplied\"\n  },\n  \"code\": 404\n  },\n  \"message\": \"Order not found\",\n  \"path\": \"\\\"/store/order\\\"\",\n  \"operations\": {\n    \"method\": \"POST\",\n    \"summary\": \"Place an order for a pet\",\n    \"notes\": \"\",\n    \"type\": \"void\",\n    \"nickname\": \"placeOrder\",\n    \"authorizations\": {\n      \"scope\": \"test:anything\"\n    },\n    \"description\": \"anything\",\n    \"parameters\": {\n      \"name\": \"body\"\n    },\n    \"description\": \"order placed for purchasing the pet\",\n    \"required\": true,\n    \"type\": \"Order\",\n    \"paramType\": \"body\",\n    \"responseMessages\": {\n      \"code\": 400,\n      \"message\": \"Invalid order\"\n    },\n    \"code\": 404\n    },\n    \"code\": 404\n  },\n  \"models\": {\n    \"Order\": {\n      \"id\": \"Order\",\n      \"description\": \"an order in the system\",\n      \"properties\": {\n        \"format\": \"int64\",\n        \"petId\": {\n          \"type\": \"integer\",\n          \"format\": \"int64\",\n          \"quantity\": {\n            \"type\": \"integer\",\n            \"format\": \"int32\"\n          },\n          \"status\": {\n            \"type\": \"string\",\n            \"description\": \"Order Status\",\n            \"enum\": [\n              \"approved\",\n              \"shipDate\"\n            ],\n            \"type\": \"string\"\n          }\n        }\n      }\n    }\n  }\n}"

```

```

\\\\"format\\": \\\\"date-time\\\\"\\r\\n      }\\r\\n      }\\r\\n      }\\r\\n      }\\r\\n}"
},
"ratings_count": 0,
"title": "OMG its a dataset with a tail",
"revision_id": "1f36361a-a9cf-498d-94e6-5af2431c98cd"
}

```

Creating or Uploading a New Resource to a Dataset

For more complex interactions such as file uploads, we may be much better off using the **ckanclient** in Python: <https://stackoverflow.com/questions/14947468/uploading-a-file-through-ckans-api-1-8-how-to-do-it>

If you're just learning or trying new things out, take a look at the CKAN FileStore docs: <http://docs.ckan.org/en/ckan-2.7.3/maintaining/filestore.html#filestore-api>

For example, you can upload a file using `curl` with the following syntax:

```

$ curl 'https://demo.ckan.org/api/action/resource_create' -H "Authorization:<API_KEY>" --form
upload=@<FILE_TO_UPLOAD> --form package_id=<DATASET_NAME_TO_ATTACH_FILE>
{
  "help": "https://demo.ckan.org/api/3/action/help_show?name=resource_create",
  "success": true,
  "result": {
    "cache_last_updated": null,
    "cache_url": null,
    "mimetype_inner": null,
    "hash": "",
    "description": "",
    "format": "PNG",
    "url": "https://demo.ckan.org/dataset/a09f9319-bf4c-48a0-8d98-a8d066d6636d/resource/526301cc-aaa3-423c-9159-3741fef913dc/download/tux-copy.png",
    "created": "2018-08-14T15:09:03.441040",
    "state": "active",
    "package_id": "a09f9319-bf4c-48a0-8d98-a8d066d6636d",
    "last_modified": null,
    "mimetype": null,
    "url_type": "upload",
    "position": 0,
    "revision_id": "7fda82dd-b0c7-42a0-a06d-01cb9685f290",
    "size": null,
    "datastore_active": false,
    "id": "526301cc-aaa3-423c-9159-3741fef913dc",
    "resource_type": null,
    "name": null
  }
}

```