Command Line

Table of Contents:

- Overview
  - Correlation Between REST and CLI
  - CLI Session Management

Overview

opencga.sh is the officially recommended command line tool for users. It implements most of the functionality with many different *commands* and *subcommands.* These *commands* are a one-to-one mapping of *Resources* from REST web services and *subcommands* are mapping to end-points. All the operations that can be performed using the command line internally creates one or several REST calls, so access to REST machine/cluster is required.

Correlation Between REST and CLI

In the following URL, "*samples"* is the resource and "*search"* is the endpoint:

http://bioinfo.hpc.cam.ac.uk/opencga-demo/webservices/rest/v1/**samples**/**search**

the corresponding command in the command line is :

```
./opencga.sh samples
```

and the corresponding subcommand is :

```
./opencga.sh samples search
```

Executing ./opencga.sh will return the list of all available commands with a description for each of them as shown below:

```
Program:     OpenCGA (OpenCB)
Version:     2.0.0-rc1
Git commit:  01cbe42598defa2ef5a192bad1f456166487aee4
Description: Big Data platform for processing and analysing NGS data

Usage:       opencga.sh [-h|--help] [--version] <command> [options]

Catalog commands:
         users  User commands
      projects  Project commands
       studies  Study commands
         files  File commands
          jobs  Jobs commands
   individuals  Individual commands
      families  Family commands
        panels  Panel commands
       samples  Samples commands
       cohorts  Cohorts commands

Analysis commands:
    alignments  Implement several tools for the genomic alignment analysis
       variant  Variant commands
      clinical  Clinical analysis commands

Operation commands:
    operations  Operations commands
```

The list of sample subcommands can be retrieved by simply executing the "samples" command without any argument as show below:

```
./opencga.sh samples

Usage:   opencga.sh samples <subcommand> [options]

Subcommands:
                create  Create a sample
                  load  Load samples from a pedigree file
                  info  Get samples information
                search  Search samples
                update  Update sample
                delete  Delete a sample
                 stats  Sample stats
                   acl  Return the acl of the resource
            acl-update  Update the permissions set for a member
annotation-sets-update  Update the value of some annotations
```

CLI Session Management

Generally, unless we are pointing to a public OpenCGA installation, users will first need to log in using the "users login" command line. Once the user has successfully logged in, a session file will be generated in their home folder:

```
~/.opencga/session.json
```

This session file contains the following information:

This makes easier for users to login only once and execute any number of commands till the session token is expired. Please note down, session expiration is set by OpenCGA server independently from client. Once token is expired, user have to login again and can perform desired operations as normal.

```
{
  "host" : "http://localhost:8080/opencga",
  "version" : "v2",
  "user" : "user1",
  "token" : "eyJhbGciOiJIUzI1NiJ9.
eyJzdWIiOiJ1c2VyMSIsImF1ZCI6Ik9wZW5DR0EgdXNlcnMiLCJpYXQiOjE1NzQxNTcyODIsImV
4cCI6MTU3NDE2MDg4Mn0.aMvD28oR_W5ZjWXuVpBXBGVHOEby-lAl7pb8oxOSXxE",
  "refreshToken" : "eyJhbGciOiJIUzI1NiJ9.
eyJzdWIiOiJ1c2VyMSIsImF1ZCI6Ik9wZW5DR0EgdXNlcnMiLCJpYXQiOjE1NzQxNTcyODIsImV
4cCI6MTU3NDE2MDg4Mn0.aMvD28oR_W5ZjWXuVpBXBGVHOEby-lAl7pb8oxOSXxE",
  "login" : "20191119095437",
  "expirationTime" : "20191119105436",
  "studies" : [ "user1@default:study1", "user1@default:study2" ]
}
```

where:

- **Line 2:** OpenCGA host against which the user has been authenticated.
- **Line 3**: API version of the OpenCGA host
- **Line 4**: Authenticated user.
- **Line 5**: Token generated when the user last logged in.
- **Line 6**: Token generated when the user last logged in to refresh token without the need to provide user credentials again.
- **Line 7**: Date when the user last logged in.
- **Line 8**: Date when the token will expire.
- **Line 9**: Studies that are accessible by the user

Authenticating is only necessary the first time. Users will have time to execute any other command line without the need to provide any more credentials until the stored token expires. Please note down that the token expiration time is set by the main OpenCGA installation. Once this token has expired, users will need to log back in again to keep working with the command line.