

Pre-requisites

GA4GH Connect - DRS test-a-thon

@April 21, 2023 at 13:30–15:00 BST

To test a DRS implementation during the DRS test-a-thon connect session, you'll need the following:

1. A [Config File](#) that includes the following details
 - a. Authorization details for the service-info endpoint
 - b. A handful of DRS IDs present in the DRS server
 - c. Authorization details for each of these DRS objects
 - d. Indicate if the DRS object is a bundle or a single blob
2. Server URL of the DRS implementation
(Eg: <https://locate.be-md.ncbi.nlm.nih.gov/ga4gh/drs/v1>)

Installations:

A computer that has at least one of the following

- a. [Python3.x](#)
- b. [Docker Desktop](#)

CONFIG FILE:

Here's a sample `config.json` file:

```
{
  "service_info": {
    "auth_type": "basic",
    "auth_token": "dXN1cm5hbWU6cGFzc3dvcmQ="
  },
  "drs_objects" : [
    {
      "drs_id": "697907bf-d5bd-433e-aac2-1747f1faf366",
      "auth_type": "none",
      "auth_token": "",
      "is_bundle": false
    },
  ],
}
```

```

{
  "drs_id": "0bb9d297-2710-48f6-ab4d-80d5eb0c9eaa",
  "auth_type": "basic",
  "auth_token": "dXNlcm5hbWU6cGFzc3dvcmQ=",
  "is_bundle": false
},
{
  "drs_id" : "41898242-62a9-4129-9a2c-5a4e8f5f0afb",
  "auth_type": "bearer",
  "auth_token": "secret-bearer-token-1",
  "is_bundle": true
},
{
  "drs_id" : "a1dd4ae2-8d26-43b0-a199-342b64c7dff6",
  "auth_type": "passport",
  "auth_token": "43b-passport-a1d",
  "is_bundle": true
}
]
}

```

- "auth_type" refers to the type of authorization. It can be one of these - ["basic", "bearer", "passport", "none"]
- "auth_token" is the corresponding authorization token value.
 - If "auth_type": "basic", then "auth_token" can be created by using base64 encoding. Here is an [online base64 encoder](#).
 - If "auth_type": "none", then "auth_token": ""
- "is_bundle" flag if true indicates that the DRS object is a bundle, else, it is a single blob.

Find more examples of config files [here](#).

INSTALLATIONS:

Install Python 3.x - (optionally use virtual environment)

- Install instructions: <https://python.land/installing-python>
- Official website: <https://www.python.org/downloads/>

- Make sure “Add Python 3.x to PATH” option is selected during installation (refer to image below)



- Create a python virtual environment

```
# Running this command should return the python version installed on your
# machine, which should be 3.x
python3 --version

# create a new directory 'venvs'
mkdir venvs

# create a new virtual env 'drscs'
python3 -m venv venvs/drscs

# Windows: activate virtual env
.\venvs\drscs\Scripts\activate

# Mac: activate virtual env
source drscs/bin/activate
```

Install Docker Desktop

- Install **Docker** (and Docker Compose) on your local machine. Verify installation by running **docker** and **docker-compose** in your terminal, you should see a dialog message explaining how to use the command in each case.
- Ensure the Docker daemon is running on your local machine at the start of the tutorial.

```
# Running this command should output a message that
# starts with “Hello from Docker!”
docker run hello-world
```