


How to create and access an S3 bucket in a local-s3 instance in EZUA

Table of contents

Issue

Environment

Cause

Resolution

Issue

Starting with EZUA 1.3, a default internal object data store named local-s3 is created. Unlike our external Data Fabric S3 Object Store, where we have an MCS portal to create buckets, access keys, etc the internal local-s3 isn't exposed to anyone outside the platform.

If one wanted to use the internal local-s3 object store for testing purposes, we can use a Jupyter notebook or any other equivalent tool to create the bucket and upload files or download files for our testing purposes.

Environment

EZUA 1.3


Cause


Unable to create s3 bucket in EZUA platform local-s3 object store.

Resolution


Below is an example of how to perform this operation.

1. Obtain the endpoint of the local-s3 object store by visiting the EZUA home page --> Data Sources --> Object Data Store --> local-s3:




local-s3 

Ezmeral Data Fabric S3



Ready



HPE Ezmeral Data Fabric Software is SaaS-based data foundation for analytics and AI across hybrid cloud to seamlessly access, analyze, and govern data globally.

<http://local-s3-service.ezdata-system.svc.cluster.local:30000>

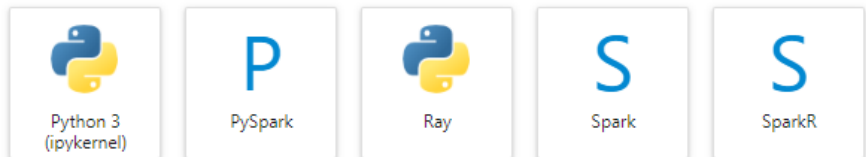
Copy the above endpoint (<http://local-s3-service.ezdata-system.svc.cluster.local:30000>) and return to the Jupyter notebook for next steps. For this exercise, the steps to create a Jupyter notebook are not shown, as EZUA ships with a pre-equipped notebook which we can use for this test.

2. Click on Connect to Notebook from the Notebooks tab and once you see the below widget click on the 1st option python3 (ipykernel)

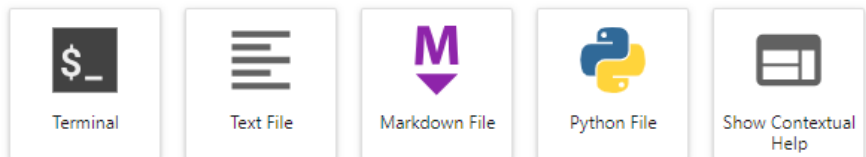
Notebook



Console



Other



3. Connect to the local-s3 using below Python boto client code snippet:

```
import boto3
client = boto3.client(
    service_name="s3",
    aws_access_key_id=os.environ["AUTH_TOKEN"],
    aws_secret_access_key='s3',
    endpoint_url="http://local-s3-service.ezdata-system.svc.cluster.local:30000",
    verify=False)
#Print Buckets
buckets = client.list_buckets()
print(buckets)
```

The above code creates a connection with the local-s3 endpoint and prints the available buckets in our local-s3. Then use the below code to create a test bucket named 'ezua-demo' and upload files available in our datasources into the local-s3 bucket:

```
# Create Bucket
client.create_bucket(Bucket="ezua-demo")
client.upload_file(Filename="./shared/ezua-tutorials/Data-Analytics/Spark/DataProcessTransfer/financial.c",
                  Bucket="ezua-demo",
                  Key="data/financial.csv")
obj=client.list_objects(Bucket="ezua-demo")
print(obj)
```

NOTE: we are using the concept of **auth_token** to connect to local-s3 object store. It is a function for notebooks which came available in EZUA 1.3. For more info on those we can refer to the EZUA documentation.

