# How to share read only access to a swift container and read data from Python

Tutorial assumes object container **testPrivateContainer** has already been created and already populated with data. Installed OpenStack swift cli tool and valid openrc file are also required.

1) Use swift cli tool to get stats on our container

```
swift stat testPrivateContainer
                    Account: AUTH_a5610a26a08a4e5fbc9dcf0dc506f03c
                  Container: testPrivateContainer
                    Objects: 1
                      Bytes: 11
                   Read ACL:
                  Write ACL:
                    Sync To:
                   Sync Key:
Strict-Transport-Security: max-age=31536000;includeSubDomains
    X-Content-Type-Options: nosniff
   Content-Security-Policy: frame-ancestors yyc.cloud.cybera.ca cloud.cybera.ca yeg.cloud.cybera.ca
              Accept-Ranges: bytes
                 X-Trans-Id: tx34d531891d434f0dafa7e-0062952219
            X-Storage-Policy: Policy-0
              Last-Modified: Mon, 30 May 2022 19:51:27 GMT
                X-Timestamp: 1653417455.66060
               Content-Type: application/json; charset=utf-8
   X-Openstack-Request-Id: tx34d531891d434f0dafa7e-0062952219
```

2) Find the OpenStack ID for users who need read access. The easiest way to do this is for the user that needs read access to run the following openstack cli command after sourcing their openrc file.

```
openstack user show "${OS_USERNAME}" -c id -f value
cc424f1b7230a8f1aeb573a899abd97d
```

3) As the owner of the container, run the following command to allow the second user read access to the container

```
swift post --read-acl "cc424f1b7230a8f1aeb573a899abd97d:*" testPrivateContainer
```

You can verify the Read ACL has been updated with the stat command. It should show the value you provided inside the quotes.

- You are able to add multiple users by using a , to separate the values, here is an example with two users in the read list

```
swift post --read-acl "cc424f1b7230a8f1aeb573a899abd97d:*,a16b42b61e9a489ca747e3b1cec6e62e:*" testPrivateContainer
```

4) If everything looks good, the below is a basic example of how to read an object in that shared container in Python.

```python
import json
import requests
from requests.structures import CaseInsensitiveDict

url = "https://keystone-yyc.cloud.cybera.ca:5000/v3/auth/tokens"

headerData = """
{
  "auth": {
    "identity": {
        "methods": ["password"],
          "password": {
             "user": {
                "domain": {"name": "Default"},
                  "name": "USERNAME@EMAIL.COM",
                  "password": "PASSWORD"
             }
          }
       },
       "scope": {
          "project": {
             "domain": {"name": "Default"},
                "name": "USERNAME@EMAIL.COM"
          }
       }
    }
}
"""

headers = CaseInsensitiveDict()
headers["Content-Type"] = "application/json"

resp = requests.post(url, headers=headers, data=headerData)

if resp.status_code != 201:
  print(resp.status_code)
else:
  authHead = resp.headers
  token = authHead["X-Subject-Token"]

  dataUrl = "https://swift-yyc.cloud.cybera.ca:8080/v1/AUTH_a5610a26a08a4e5fbc9dcf0dc506f03c
/testPrivateContainer/someDataFile"

  dataHeaders = CaseInsensitiveDict()
  dataHeaders["Content-Type"] = "application/json"
  dataHeaders["X-Auth-Token"] = token

  dataResp = requests.get(dataUrl, headers=dataHeaders)
  print(dataResp.text)
```