Linux Academy

Live! Lab

Set Expire Times on Objects

Contents

Introduction 1

Create Container 1

View Object Stats 1

Set Object to Expire 2

Lab Connection Information

- Labs may take up to five minutes to build
- Access to the Horizon Dashboard is provided on the Live! Lab page, along with your login credentials
- SSH information is provided on the Live! Lab page
- Labs will automatically end once the alloted amount of time finishes

Related Courses

OpenStack Foundation Certified OpenStack Administrator

Related Videos

Swift - Object Storage Overview

Account, Proxy, Object and Container Services

Manage Expiring Objects

Need Help?

Linux Academy Community

... and you can always send in a support ticket on our website to talk to an instructor!

Introduction

Objects contained in Swift containers can be configured to delete automatically after a set amount of time. This lab demonstrates how to create a container, add an object to that container and set an expire time for that object.

Log in to the Horizon Dashboard using the *demo* tenant, and ensure you have the *demo* project selected. Navigate to **Access & Security** page, then click the **API Access** tab. **Download OpenStack RC File**.

SSH into the server through your terminal using the SSH Details provided on the Live! Lab page.

Copy the content of the file to the terminal instance, either by copying and pasting it into a new file called *demo.sh*, or using `scp` to copy it up to your server. Ensure the file is named *demo.sh*.

Source the file, entering your *demo* user's password when prompted:

```
root@ubuntu-openstack:~# source demo.sh
```

Create Container

Check to see if there are currently any containers available:

```
root@ubuntu-openstack:~# swift list
```

There are none. Return to the Horizon Dashboard to create a container. Select the **Object Store** heading on the left menu, then click **Containers**. Press **+Create Container**.

We named ours *container1* and set the **Container Access** to *public*. Press **Create Container**.

Run `swift list` again to view the available container.

List the files in the root directory: Your *demo.sh* file should be accessible. We want to upload this to our container. To do this, use:

```
root@ubuntu-openstack:~# swift upload container1 demo.sh
```

View Object Stats

Before we can set the expire time on an object, we need to find the auth token for the container and the public URL for the object. To see details about *container1*'s *demo.sh* file, run:

```
root@ubuntu-openstack:~# swift stat -v container1 demo.sh
           URL: http://172.99.75.133:8080/v1/AUTH_51be0a3ab19443eeb2367f04144936dd/
container1/demo.sh
    Auth Token: 87279525d73d4be59dcc2df9961a1dc2
       Account: AUTH_51be0a3ab19443eeb2367f04144936dd
     Container: container1
        Object: demo.sh
  Content Type: text/x-sh
Content Length: 1572
 Last Modified: Mon, 09 May 2016 15:13:07 GMT
          ETag: 59755c6d1bba1227ff84b9b8e832b927
    Meta Mtime: 1462806616.360438
 Accept-Ranges: bytes
   X-Timestamp: 1462806786.67947
    X-Trans-Id: txbeec52b884db4f2ea146a-005730a91d
```

Should you copy the endpoint URL into your web browser, you are asked to download the file: This proves that the container is public.

Set Object to Expire

Object deletion is set in epoch time. In this lab, we want to set the object to delete twenty-four hours from now. To determine this length of time, we need to do some simple math:

```
24 * 60 * 60 = 86400
```

To set the object to delete at the designated time, we need to use a combination of `curl` and `POST` commands:

```
root@ubuntu-openstack:~# curl -X POST -H 'X-Auth-Token: 87279525d73d4be59dcc2df9961a1dc2'
-H 'X-Delete-After: 86400' http://172.99.75.133:8080/v1/
AUTH_51be0a3ab19443eeb2367f04144936dd/container1/demo.sh
```

The information used to run the above command properly can be found in the results from the `swift stat` command, used earlier. Substitute in your auth token and public endpoint URL.

Run `swift stat -v container1 demo.sh`; you can now see it has an expire time listed in epoch time.