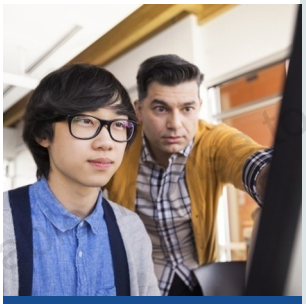
15

Oracle Storage Cloud Service Containers and Objects

Objectives

After completing this lesson, you should be able to:

- Create containers
- List containers
- Set container metadata
- Delete containers
- List objects in a container
- Create objects
- Download objects
- Delete objects
- Update object metadata



Two Sections

SECTION I: CONTAINERS

- Create containers
- List containers
- Set container metadata
- Delete containers

SECTION II: OBJECTS

- List objects in a container
- Create objects
- Download objects
- Delete objects
- Update object metadata

This lesson has two sections.

- **Section I:** Focuses on managing (object storage) *containers* in Oracle Storage Cloud Service
- **Section II:** Focuses on managing (object storage) *objects* in Oracle Storage Cloud Service

Containers

SECTION I: Creating Containers

- cURL command syntax:

```
curl -v -X PUT \
     -H "X-Auth-Token: token" \
     accountURL/containerName
```

- cURL command example:

```
curl -v -X PUT \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/FirstContainer
```


You will need to substitute the parameters highlighted in the slide based on your account's information.

Where:

- `token` is the authentication token obtained earlier from Oracle Storage Cloud Service.
- `accountURL` is the URL that you see in My Services when signed in to the Oracle Cloud My Services application—This is the Global namespace URL for all customers.
  - Other type of URLs assigned to customers include: Data center–specific URL, URLs based on metered subscription and nonmetered subscription. (The latter two are URLs in the earlier release of Oracle Storage Cloud Service.)
- `containerName` is the name of the container to be created.
  - The container name can be from 1 to 256 characters long and must consist of alphanumeric characters.

HTTP Response Codes

- `Success: 201 Created`

cURL Command Example:

```
curl -v -X PUT \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```

- Where:
  - `Token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `containerName` **is** `FirstContainer`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> PUT /v1/Storage-myIdentity3/FirstContainer HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
>
< HTTP/1.1 201 Created
< Date: Fri, 06 Mar 2015 10:34:20 GMT
< Content-Length: 0
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: tx23a1084b8c674fdeae8d4-0054f982ac
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION I: Listing Containers

- All containers within an account can be listed.
- Any user in the identity domain can perform this task.
- Containers are sorted lexicographically using `memcmp()`.
- Query parameters include:
  - `token`
  - `marker`
  - `end_marker`
  - `format`
    - Return `Xml or json` format (REST API only)

Containers are sorted lexicographically using **`memcmp()`**. All containers, up to 10000 by default, will be returned in the list, unless you filter the list by using any of the following query parameters:

- **`limit:`** Limit the number of containers listed to the specified value. The default and maximum value is 10000.
- **`marker:`** Return containers with names greater than the specified string.
- end_marker: Return containers with names less than the specified string.
- **`format:`** Return extended information about each returned container in either *xml* or *json* format (REST API only).

SECTION I: Listing Containers

- cURL command syntax:

```
curl -v -X GET \
     -H "X-Auth-Token: token" \
     accountURL[?query_parameter=value]
```

- cURL command example:

```
curl -v -X GET \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3?limit=15
```


- You will need to substitute the parameters highlighted in the slide based on your account's information.
- Parameter not yet introduced:
  - `query_parameter=value` is the optional filtering parameter

HTTP Response Codes

- `Success: 200 OK`

cURL Command Example:

```
curl -v -X GET \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3?limit=15
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `query_parameter` **is** `limit=15`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> GET /v1/Storage-myIdentity3 HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tk6403794c218a709d1c6c5a76444d01f6
>
< HTTP/1.1 200 OK
< Date: Fri, 06 Mar 2015 10:38:15 GMT
< Content-Length: 109
< X-Account-Container-Count: 3
< Accept-Ranges: bytes
< X-Account-Object-Count: 843
< X-Account-Bytes-Used: 10304761355
< X-Timestamp: 1412823447.62495
< X-Account-Meta-Test5: test1
< X-Account-Meta-Quota-Bytes: 107374182400
< Content-Type: text/plain; charset=utf-8
< X-Account-Meta-Test: test
< X-Account-Meta-Test1: test1
< X-Trans-Id: tx29052c64fe384fc690ccc-0054f98397
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
FirstContainer
hello
lab
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION I: Setting Container Metadata

- Only the following metadata can be changed in a container:
  - Container ACLs: `X-Container-Read` and `X-Container-Write`
  - Quotas: `X-Container-Meta-Quota-Bytes` and `X-Container-Meta-Quota-Count`
  - Custom metadata: `X-Container-Meta-Name`

SECTION I: Setting Container Metadata: ACLs

- Governed by Access Control Lists (ACLs)
  - Ability to read and write objects in assigned container
  - User must have the Service Administrator role
- A container has two ACLs:
  - `X-Container-Read`
  - `X-Container-Write`
- Roles can be built-in roles or custom roles
- Referrer designation indicates the host (or hosts)
  - Read access to the container should be allowed or denied.
  - The syntax of the referrer designation is `.r:value`.

The ability to read and write objects in a container is governed by the Access Control Lists (ACLs) assigned to the container. Any user with the Service Administrator role can perform this task.

A container has two ACLs: `X-Container-Read` and `X-Container-Write`.

The `X-Container-Read` ACL consists of a comma-separated list of roles or referrer designations. The `X-Container-Write` ACL consists of a comma-separated list of roles.

- The roles can be built-in roles or custom roles. Custom roles are defined on the My Services Security page.
  - For a role that was provisioned as part of another service instance, the format is `domainName.serviceName.roleName`.
  - For a custom role, the format is `domainName.roleName`.
- A referrer designation indicates the host (or hosts) for which read access to the container should be allowed or denied. When the server receives a request for the container, it compares the referrer designations specified in the `X-Container-Read` ACL with the value of the `Referrer` header in the request, and determines whether access should be allowed or denied. The syntax of the referrer designation is `.r:value`.

SECTION I: Setting Container Metadata: ACLs

- cURL command syntax: Grant access `X-Container-Write`

```
curl -v -X POST \
     -H "X-Auth-Token: token" \
     -H "X-Container-Read: item[,item...]" \
     -H "X-Container-Write: item[,item...]" accountURL/containerName
```

- cURL command example: Grant access `X-Container-Write`

```
curl -v -X POST \
        -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
        -H "X-Container-Write:
myDomain.Storage.Storage_ReadWriteGroup,myDomain.myCustomRole" \
        https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```

Provide write access for any user with the predefined role, `Storage_ReadWriteGroup` or the custom role, `myCustomRole`.

HTTP Response Codes:

- `Success: 204 No Content`

cURL Command Example:

```
curl -v -X POST \
        -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
        -H "X-Container-Write:
myDomain.Storage.Storage_ReadWriteGroup,myDomain.myCustomRole" \
        https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `X-Container-Write` **is** `myDomain.Storage.Storage_ReadWriteGroup,myDomain.myCustomRole`
  - `accountURL` is `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `containerName` **is** `FirstContainer`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> POST /v1/Storage-myIdentity3/FirstContainer HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
> X-Container-Write:Storage-myIdentity3.Storage.Storage_ReadWriteGroup,Storage-
myIdentity3.myCustomRole
>
< HTTP/1.1 204 No Content
< Date: Fri, 06 Mar 2015 11:19:21 GMT
< Content-Length: 0
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: txbf2c736d57494bf88e76a-0054f98d39
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION I: Setting Container Metadata: ACLs

- cURL Command Syntax: Grant access `X-Container-Read`

```
curl -v -X POST \
     -H "X-Auth-Token: token" \
     -H "X-Container-Read: item[,item...]" \
     -H "X-Container-Write: item[,item...]" accountURL/containerName
```

- cURL Command Example: Grant access `X-Container-Read`

```
curl -v -X POST \
        -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
        -H "X-Container-Read: .r:*,.rlistings" \
  https://foo.storage.oraclecloud.com/v1/Storage-myDomain/FirstContainer
```


- A referrer designation indicates the host (or hosts) for which read access to the container should be allowed or denied. When the server receives a request for the container, it compares the referrer designations specified in the `X-Container-Read` ACL with the value of the `Referrer` header in the request, and determines whether access should be allowed or denied. The syntax of the referrer designation is: `.r:value`
  - value indicates the host for which access to the container should be allowed. It can be a specific host name (example: `.r:www.example.com`), a domain (example: `.r:.example.com`), or an asterisk (`.r:*`) to indicate all hosts. Note that if `.r:*` is specified, objects in the container will be publicly readable without authentication.
  - A minus sign (–) before value (example: `.r:-temp.example.com`) indicates that the host specified in the value field must be denied access to the container.
  - By default, read access to a container does not include permission to list the objects in the container. To allow listing of objects as well, include the `.rlistings` directive in the ACL (example: `.r:*,.rlistings`).
- Parameter not yet introduced:
  - item can be either a role or a referrer designation.

cURL Command Example:

```
curl -v -X POST \
        -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
        -H "X-Container-Read: .r:*,.rlistings" \
        https://foo.storage.oraclecloud.com/v1/Storage-myDomain/FirstContainer
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `X-Container-Read` **is** `.r:*,.rlistings`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myDomain`
  - `containerName` **is** `FirstContainer`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> POST /v1/Storage-myIdentity3/FirstContainer HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
> X-Container-Read: .r:*,.rlistings
>
< HTTP/1.1 204 No Content
< Date: Fri, 06 Mar 2015 11:23:16 GMT
< Content-Length: 0
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: tx9127a70f18144c17afce5-0054f98e24
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION I: Setting Container Quotas

- Setting quotas to each container:
  - Maximum number of bytes
    - `X-Container-Meta-Quota-Bytes`
  - Maximum number of objects
    - `X-Container-Meta-Quota-Count`
  - User must have the Service Administrator role.
- Any user with the Service Administrator role can perform this task.

For each container, you can set quotas for the maximum number of bytes the container can contain (`X-Container-Meta-Quota-Bytes`) and the maximum number of objects the container can contain (`X-Container-Meta-Quota-Count`).

SECTION I: Setting Container Quotas

- cURL command syntax:

```
curl -v -X POST \
     -H "X-Auth-Token: token" \
     -H "X-Container-Meta-Quota-Bytes: maxBytes" \
     -H "X-Container-Meta-Quota-Count: maxObjects" accountURL/containerName
```

- cURL command example:

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Container-Meta-Quota-Bytes: 10737418240" \
     -H "X-Container-Meta-Quota-Count: 100" \
     https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```


- Parameters not yet introduced:
  - `maxBytes` is the maximum number of bytes of data that can be stored in the container.
  - `maxObjects` is the maximum number of objects that can be created in the container.

HTTP Response Codes

- `Success: 204 No Content`

cURL Command Example:

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Container-Meta-Quota-Bytes: 10737418240" \
     -H "X-Container-Meta-Quota-Count: 100" \
     https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `maxBytes` **is** `10737418240`
  - `maxObjects` **is** `100`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/`
  - `containerName` **is** `FirstContainer`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> POST /v1/Storage-myIdentity3/FirstContainer HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
> X-Container-Meta-Quota-Bytes: 10737418240
> X-Container-Meta-Quota-Count: 100
>
< HTTP/1.1 204 No Content
< Date: Fri, 06 Mar 2015 11:32:19 GMT
< Content-Length: 0
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: txe8869b3edea348e5b49eb-0054f99043
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION I: Setting Custom Metadata

- cURL command syntax:

```
curl -v -X POST \
     -H "X-Auth-Token: token" \
     -H "X-Container-Meta-Name: value" \
     accountURL/containerName
```

- cURL command example:

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Container-Meta-Category: Books" \
     https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```


- Custom metadata are arbitrary key-value pairs associated with a container. You may create any custom or arbitrary metadata you need.
- Any user with the Service Administrator role can perform this task.
- Parameters not yet introduced:
  - `Name` and `value` are the metadata key and value to be created.

HTTP Response Codes

- `Success: 204 No Content`

cURL Command Example:

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Container-Meta-Category: Books" \
     https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `X-Container-Meta-Category` **is** `Books`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `containerName` **is** `FirstContainer`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> POST /v1/Storage-myIdentity3/FirstContainer HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
> X-Container-Meta-Category: Books
>
< HTTP/1.1 204 No Content
< Date: Fri, 06 Mar 2015 11:35:35 GMT
< Content-Length: 0
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: tx3e77b77de39f4097a5a49-0054f99107
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION I: Deleting Containers

- To delete a container, all of its objects must be deleted first.
- Any user with the Service Administrator role can perform this task.

- **All** objects within a container **must** first be deleted before the container can be deleted. To find out whether a container contains any objects, send a HEAD request to the container URL.
- Any user with the Service Administrator role can perform this task.

HTTP Response Codes

- `Success: 204 Content`

SECTION I: Deleting Containers

- cURL command syntax:

```
curl -v -X DELETE \
     -H "X-Auth-Token: token" \
     accountURL/containerName
```

- cURL command example:

```
curl -v -X DELETE \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```


cURL Command Example:

```
curl -v -X DELETE \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3/FirstContainer
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `containerName` **is** `FirstContainer`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> DELETE /v1/Storage-myIdentity3/FirstContainer HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
>
< HTTP/1.1 204 No Content
< Date: Fri, 06 Mar 2015 10:43:38 GMT
< Content-Length: 0
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: txc100a7408d564f82916fb-0054f984da
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

Objects

SECTION II: Listing Objects in a Container

- All objects in a container can be listed.
- Who can perform this task?
  - Any user with the Service Administrator role
  - Any user with a role specified in the `X-Container-Read` ACL of the container
- Objects are sorted by their names lexicographically, using `memcmp()`.
- Available parameters:
  - `limit`
  - `marker`
  - `end_marker`
  - `format`
  - `prefix`
  - `Delimiter`

Objects are sorted by their names lexicographically, using `memcmp()`.

All objects, up to 10000 by default, will be returned in the list, unless you filter the list by using any of the following parameters:

- **`limit:`** Limits the number of objects listed to the specified value. The default and maximum value is 10000.
- **`marker:`** Returns objects with names greater than the specified string
- **`end_marker:`** Returns objects with names less than the specified string
- format: Returns extended information about each returned object in either `xml` or `json` format (REST API only)
- **`prefix:`** Returns objects with names that start with the specified string
- **`delimiter:`** Returns objects with names that include the specified character. Only the substring of object names before the specified character are returned; only unique substrings are returned.
  - If the `prefix` parameter is also used, any matches of the specified delimiter character are ignored.
  - The prefix parameter is used to emulate directory structures in a container (that is, with a forward slash (/) as the delimiter)

SECTION II: Listing Objects in a Container

- cURL command syntax:

```
curl -v -X GET \
     -H "X-Auth-Token: token" \
     accountURL/containerName[?query_parameter=value]
```

- cURL command example:

```
curl -v -X GET \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentityDomainID/myContainer?limit=15
```


- Parameters not yet introduced:
  - `containerName` is the name of the container for which objects should be listed.
  - `query_parameter=value` is the optional filtering parameter.

HTTP Response Codes

- `Success: 200 OK`
- *If there are no objects, the HTTP response code would be `204 No Content`.*

cURL Command Example:

```
curl -v -X GET \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentityDomainID/myContainer?limit=15
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentityDomainID`
  - `containerName` **is** `myContainer`
  - `limit is 15`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> GET /v1/Storage-myIdentityDomainID/FirstContainer HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tk4af5123f84d2e3ffb9e77ba657ac8edf
>
< HTTP/1.1 200 OK
< Date: Mon, 09 Mar 2015 11:15:50 GMT
< Content-Length: 63
< X-Container-Object-Count: 4
< X-Container-Write: myIdentityDomainID.Storage.Storage_ReadWriteGroup
< Accept-Ranges: bytes
< X-Timestamp: 1425033529.95392
< X-Container-Read:
myIdentityDomainID.Storage.Storage_ReadOnlyGroup,myIdentityDomainID.Storage.Stora
ge_ReadWriteGroup
< X-Container-Bytes-Used: 92095
< Content-Type: text/plain; charset=utf-8
< X-Trans-Id: tx23ba568df8864b45bc443-0054fd80e6
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
Backup-2-0_24680
Backup-3-0_32872
MetadataLog-0_32872
test.key
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION II: Creating Objects

- Objects must be created in a container.
- There are two way to create objects:
  - Uploading files
  - Specifying metadata
- Who can perform this task?
  - Any user with the Service Administrator role
  - Any user with a role that is specified in the `X-Container-Write` ACL of the container
- Objects can be created in a `standard` or an `archive` container
- The following are the object parameters:
  - `containerName, objectName, file`

Objects are sorted by their names lexicographically using `memcmp()`. All objects, up to 10000 by default, will be returned in the list, unless you filter the list by using any of the following parameters:

- `containerName` is the name of the container in which the object should be created.
- `objectName` is the name of the object to be created.
- `file` is the full path and name of the file to be uploaded.

Note: If you have a large file, greater than 5GB in size, split it into segments using the `split` command on Linux or a utility such as WinZip on a Windows computer.

SECTION II: Creating Objects

- cURL command syntax:

```
curl -v -X PUT \
     -H "X-Auth-Token: token" \
     -T file \
     accountURL/containerName/objectName
```

- cURL command example:

```
curl -v -X PUT \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -T myFile.txt \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/FirstContainer/myObject
```

- Parameters not yet introduced:
  - `containerName` is the name of the container in which the object should be created.

HTTP Response Codes

- `Success: 201 Created`

cURL Command Example:

```
curl -v -X PUT \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -T myFile.txt \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/FirstContainer/myObject
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `containerName` **is** `FirstContainer`
  - `objectName` **is** `myObject`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> PUT /v1/Storage-myIdentity3/FirstContainer/myObject HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
> Content-Length: 23
> Expect: 100-continue
>
* Done waiting for 100-continue
< HTTP/1.1 201 Created
< Date: Mon, 09 Mar 2015 11:26:57 GMT
< Last-Modified: Mon, 09 Mar 2015 11:26:58 GMT
< Content-Length: 0
< Etag: 846fa9d298be05e5f598703f0c3d6f51
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: tx2a97f34acb7048679ae3b-0054fd8381
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION II: Downloading Objects

- cURL command syntax:

```
curl -v -X GET \
     -H "X-Auth-Token: token" \
     -o file \
     accountURL/containerName/objectName
```

- cURL command example:

```
curl -v -X GET \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -o myFile.txt \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/myContainer/myObject
```


When you download an object, the object's metadata and data are downloaded.

Any user with the Service Administrator role or a role that is specified in the X-Container-Read ACL of the container can perform this task.

Parameters:

- `file` is the full path and name of the file to which the object should be downloaded.
- `containerName` is the name of the container that contains the object to be downloaded.
- `objectName` is the name of the object to be downloaded.

HTTP Response Codes

- `Success: 200 OK`

cURL Command Example

```
curl -v -X GET \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -o myFile.txt \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/myContainer/myObject
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `containerName` **is** `myContainer`
  - `objectName` **is** `myObject`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> GET /v1/Storage-myIdentity3/FirstContainer/myObject HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
>
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
<
< HTTP/1.1 200 OK
< Date: Mon, 09 Mar 2015 11:34:33 GMT
< Content-Length: 23
< Accept-Ranges: bytes
< Last-Modified: Mon, 09 Mar 2015 11:26:58 GMT
< Etag: 846fa9d298be05e5f598703f0c3d6f51
< X-Timestamp: 1425900417.95553
< Content-Type: application/octet-stream
< X-Trans-Id: txf0b592c7e49b4475944f8-0054fd8549
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
{ [data not shown]
  0    23    0    23    0     0     53      0 --:--:-- --:--:-- --:--:--   234*
Connection #0 to host foo.storage.oraclecloud.com left intact


* Closing connection #0
```

SECTION II: Deleting Objects

- cURL command syntax:

```
curl -v -X DELETE \
     -H "X-Auth-Token: token" \
     accountURL/containerName/objectName
```

- cURL command example:

```
curl -v -X DELETE \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/FirstContainer/myObject2
```


Any user with the Service Administrator role or a role that is specified in the `X-Container-Read` ACL of the container can perform this task.

Parameters:

- `containerName` is the name of the container that contains the object to be deleted.
- `objectName` is the name of the object to be deleted.

HTTP Response Codes

- `Success: 204 Content`

cURL Command Example:

```
curl -v -X DELETE \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/FirstContainer/myObject2
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `containerName` **is** `FirstContainer`
  - `objectName` **is** `myObject2`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> DELETE /v1/Storage-myIdentity3/FirstContainer/myObject2 HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
>
< HTTP/1.1 204 No Content
< Date: Mon, 09 Mar 2015 11:40:23 GMT
< Content-Length: 0
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: tx47aef42f16c44bd9a72cb-0054fd86a7
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
```

SECTION II: Updating Object Metadata

- Updating Custom Metadata for Objects
  - Arbitrary key-value pairs
  - Metadata keys:
    - Underscore
    - First letter capitalized, the rest lowercase
- Schedule deletion
  - At a specified time in the future
  - After a specified period of time has elapsed
- Any user with the Service Administrator role can perform this task:
  - `X-Container-Write`
- Scheduling Automatic Deletion of Objects

- Custom metadata are arbitrary key-value pairs. You may define and update any custom or arbitrary metadata that you need.
- Any user with the Service Administrator role or a role that is specified in the `X-Container-Write` ACL of the container can perform this task.
- The service transforms custom metadata keys as follows:
  - Underscores are converted to hyphens.
  - The first character after a hyphen is capitalized. All other letters are converted to lowercase.
- You can schedule deletion of objects at a specified time in the future or after a specified period of time has elapsed, by using the `X-Delete-After` or `X-Delete-At header`, respectively.

SECTION II: Updating Object Metadata

- cURL command syntax:

```
curl -v -X POST \
     -H "X-Auth-Token: token" \
     -H "X-Object-Meta-Name: value" \
     accountURL/containerName/objectName
```

- cURL command example:

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Object-Meta-Language: english" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/FirstContainer/myObject
```


Remember: The service transforms custom metadata keys as follows:

- Underscores are converted to hyphens.
- The first character after a hyphen is capitalized. All other letters are converted to lowercase.

Parameters not yet introduced:

- `name` and `value` are the metadata key and value to be created (using `X-Object-Meta-Name`).
- `containerName` is the name of the container that contains the object for which custom metadata should be created
- `objectName` is the name of the object for which custom metadata should be created.

HTTP Response Codes

- `Success: 202 Accepted`

cURL Command Example:

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Object-Meta-Language: english" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/FirstContainer/myObject
```

- Where:
  - Token **is** AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
  - accountURL **is** https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3
  - containerName **is** FirstContainer
  - objectName **is** myObject
  - X-Object-Meta-Name **is** X-Object-Meta-Language
  - value **is** english

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> POST /v1/Storage-myIdentity3/FirstContainer/myObject HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
> X-Object-Meta-Language: english
>
< HTTP/1.1 202 Accepted
< Date: Mon, 09 Mar 2015 11:46:34 GMT
< Content-Length: 76
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: txd54813b92dcd46849b009-0054fd8819
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
Accepted. The request is accepted for processing.
```

SECTION II: Scheduling Automatic Deletion of Objects

- cURL command syntax: After elapsed time

```
curl -v -X POST \
     -H "X-Auth-Token: token" \
     -H "X-Delete-After: period" \
     accountURL/containerName/objectName
```

- cURL command example: After elapsed time

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Delete-After: 86400" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/myContainer/myObject
```


You can schedule deletion of objects after a specified period of time has elapsed, by using the `X-Delete-After or X-Delete-At` header, respectively.

Note: You cannot schedule automatic deletion of objects for an `Archive` container by using the `X-Delete-After` and `X-Delete-At` headers.

The following command sets the object named `myObject` to be deleted automatically after 86400 seconds:

Parameters not yet introduced:

- `period` is the duration, in seconds, after which the object should be deleted (`X-Delete-After`).

HTTP Response Codes

- `Success: 202 Accepted`

cURL Command Example:

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Delete-After: 86400" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/myContainer/myObject
```

- Where:
  - `token` **is** `AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b`
  - `accountURL` **is** `https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3`
  - `containerName` **is** `myContainer`
  - `objectName` **is** `myObject`
  - `period` **is** `86400`

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> POST /v1/Storage-myIdentity3/myContainer/myObject HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
> X-Delete-After: 86400
>
< HTTP/1.1 202 Accepted
< Date: Mon, 23 Mar 2015 12:32:39 GMT
< Content-Length: 76
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: txbb5a2f22164e47aa8116f-00551007e7
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
The request is accepted for processing.
```

SECTION II: Scheduling Automatic Deletion of Objects

- cURL command syntax: Specified Time

```
url -v -X POST \
     -H "X-Auth-Token: token" \
     -H "X-Delete-At: time" \
     accountURL/containerName/objectName
```

- cURL command example: Specified Time

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Delete-At: 1417341600" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/myContainer/myObject
```


You can schedule deletion of objects at a specified time in the future, by using the `X-Delete-After` or `X-Delete-At` header, respectively.

Note: You cannot schedule automatic deletion of objects for an `Archive` container by using the `X-Delete-After` and `X-Delete-At` headers.

The following command sets the object named myObject to be deleted automatically on November 30, 2014 at 10:00:00 GMT, represented by the UNIX Epoch timestamp, 1417341600:

- Parameters not yet introduced:
  - `time` is the UNIX Epoch timestamp representing the date and time at which the object should be deleted. For example, 1416218400 represents November 17, 2014 10:00:00 GMT (`X-Delete-At`).

HTTP Response Codes

- `Success: 202 Accepted`

cURL Command Example:

```
curl -v -X POST \
     -H "X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b" \
     -H "X-Delete-At: 1417341600" \
     https://foo.storage.oraclecloud.com/v1/Storage-
myIdentity3/myContainer/myObject
```

- Where:
  - Token **is** AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
  - accountURL **is** https://foo.storage.oraclecloud.com/v1/Storage-myIdentity3
  - containerName **is** myContainer
  - objectName **is** myObject
  - time **is** 1417341600

The following is an example of the output of this command:

```
* About to connect() to foo.storage.oraclecloud.com port 443 (#0)
*   Trying 160.34.0.51... connected
* Connected to foo.storage.oraclecloud.com (160.34.0.51) port 443 (#0)
* Initializing NSS with certpath: sql:/etc/pki/nssdb
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* SSL connection using TLS_RSA_WITH_AES_128_CBC_SHA
* Server certificate:
*       subject: CN=*.us2.oraclecloud.com,O=Oracle Corporation,L=Redwood
Shores,ST=California,C=US
*       start date: Oct 22 00:00:00 2014 GMT
*       expire date: Dec 21 23:59:59 2015 GMT
*       common name: *.us2.oraclecloud.com
*       issuer: CN=Symantec Class 3 Secure Server CA - G4,OU=Symantec Trust
Network,O=Symantec Corporation,C=US
> POST /v1/Storage-myIdentity3/myContainer/myObject HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.14.0.0
zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: foo.storage.oraclecloud.com
> Accept: */*
> X-Auth-Token: AUTH_tkb4fdf39c92e9f62cca9b7c196f8b6e6b
> X-Delete-At: 1417341600
>
< HTTP/1.1 202 Accepted
< Date: Mon, 23 Mar 2015 12:32:39 GMT
< Content-Length: 76
< Content-Type: text/html; charset=UTF-8
< X-Trans-Id: txbb5a2f22164e47aa8116f-00551007e7
< Cache-Control: no-cache
< Pragma: no-cache
< Content-Language: en
<
* Connection #0 to host foo.storage.oraclecloud.com left intact
* Closing connection #0
The request is accepted for processing.
```

Quiz

Which of the following was NOT covered in this lesson?

a. Creating containers
b. Downloading containers
c. Updating object metadata
d. Listing objects in a container
e. Deleting containers

Quiz

What are the two ACLs for containers?

a. X-Container-Read and X-Container-Write
b. X-Container-Read and Y-Container-Write
c. Y-Container-Read and Y-Container-Write
d. Y-Container-Read and X-Container-Write

Quiz

When you successfully retrieve a list of objects in a container, what is the HTTP Response Code?

a. 200 OK
b. 201 Created
c. 202 Accepted
d. 204 No Content

Quiz

Which of the following is NOT a parameter when downloading an object?

a. file
b. containerName
c. objectName
d. format

Quiz

You can schedule automatic deletion of objects:

a. By specifying a time
b. After a specified period of time has elapsed
c. On a recurring basis
d. A and B
e. A and C

Summary

In this lesson, you should have learned how to:

- Create containers
- List containers
- Set container metadata
- Delete containers
- List objects in a container
- Create objects
- Download objects
- Delete objects
- Update object metadata