Object expiry

> Object expiry requires that you configure your environment with working S3-compatible credentials.

You can set a bucket's lifecycle configuration such that it automatically deletes objects after a certain number of days.

Enabling object expiry

First, you need to create a JSON file, `lifecycle.json`, that contains the lifecycle configuration rule. Be sure to set `Days` to your desired value:

```
{
  "Rules": [{
    "ID": "cleanup",
    "Status": "Enabled",
    "Prefix": "",
    "Expiration": {
       "Days": 5
    }
  }]
}
```

Then, apply this lifecycle configuration to your bucket using one of the following commands:

aws **mc** **s3cmd**

```
aws --profile <region> \
  --endpoint-url=https://
s3-
<region>.citycloud.com:
8080 \
  s3api put-bucket-
lifecycle-configuration \
  --lifecycle-
configuration file://
lifecycle.json \
  --bucket <bucket-
name>

mc ilm import
<region>/<bucket-
name> < lifecycle.json

s3cmd -c ~/.s3cfg-
<region> setlifecycle
```

```
lifecycle.json s3://
<bucket-name>
```

Removing object expiry

At some point, you might want to remove the object expiry functionality configuration from a bucket, so that objects in it no longer auto-delete after a period.

aws **mc** **s3cmd**

With the `aws s3api` command, you can remove the lifecycle configuration from a bucket:

```
aws --profile <region> \
  --endpoint-url=https://
s3-
<region>.citycloud.com:
8080 \
  s3api delete-bucket-
lifecycle \
  --bucket <bucket-
name>
```

With `mc`, you are able to remove just an individual bucket lifecycle rule. Assuming your rule uses the ID `cleanup`, here is how you remove it:

Last update: 2022-12-19
Created: 2022-05-11
Authors: Florian Haas, Toad Lind

```
mc ilm rm --id "cleanup"
<region>/<bucket-
name>
```

With `s3cmd`, you can remove the lifecycle configuration from a bucket:

```
s3cmd -c ~/.s3cfg-
<region> dellifecycle
s3://<bucket-name>
```