Public buckets

You can use the S3 API to configure a bucket with public read access, so that anyone can download its objects with a web browser. Making a bucket globally readable entails setting a *bucket policy* that enables read access on all its objects.

> Setting a bucket policy affects **all** objects in a bucket. To avoid inadvertent disclosure of existing information, consider setting public read policies only on empty buckets.

Prerequisites

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

Setting a public read policy for a bucket

First, create a local policy file named `policy.json`, with the following content (replace `<bucket-name>` with the name of your bucket):

```
{
 "Statement": [
  {
   "Effect": "Allow",
   "Principal": "*",
   "Action": "s3:GetObject",
   "Resource": "arn:aws:s3:::<bucket-name>/*"
  }
 ]
}
```

To apply this policy to a bucket such that read-only access is permitted for everyone, run the following command:

aws **mc** **s3cmd**

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

```
  --bucket <bucket-
name>

mc anonymous set-json
policy.json <region>/
<bucket-name>

s3cmd -c ~/.s3cfg-
<region> setpolicy
policy.json s3://<bucket-
name>
```

cs in a public bucket

ublic bucket from a web browser or a generic HTTP/HTTPS client like `curl`, you must construct its URI as follows:

```
https://s3-<region>.citycloud.com:8080/<project-uuid>:<bucket-name>/object-name
```

> Your project UUID is listed as the `project_id` field in the output of the `openstack ec2 credentials create` command you used to create your S3-compatible credentials.
>
> If you did not note it down at the time of account creation, you can always retrieve it with `openstack ec2 credentials <access-key>`.

For example, to retrieve an object named `bar.pdf` in a bucket named `foo` from the project with the UUID `07576783684248f7b2745e34356c6025` in the Cleura Cloud Kna1 region, you would run:

```
$ curl -O https://s3-kna1.citycloud.com:8080/07576783684248f7b2745e34356c6025:foo/
bar.pdf
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 62703  100 62703    0     0   186k      0 --:--:-- --:--:-- --:--:--  186k
```

Enabling bucket listing

The `policy.json` file above allows anyone to retrieve known objects by name, but does not enable listing the bucket's contents. Most of the time, this is what you want.

However, in case you do need unauthenticated clients to be able to list all objects in a bucket, modify your `policy.json` file as in the following example:

```
{
 "Statement": [
  {
   "Effect": "Allow",
   "Principal": "*",
   "Action": "s3:GetObject",
   "Resource": "arn:aws:s3:::<bucket-name>/*"
```

```
    },
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::<bucket-name>"
    }
  ]
}
```

Note that for the `s3:GetObject` action, `/*` follows the bucket name, whereas for the `s3:ListBucket` action you specify *just* the bucket name with no suffix.

Once you have updated your local `policy.json` file, apply it just as when you created the policy.

You can then open a bucket path in your browser, to retrieve an XML document containing a list of all objects in the bucket. You would construct the URL by the following schema:

```
https://s3-<region>.citycloud.com:8080/<project-uuid>:<bucket-name>
```

Removing a public read policy from a bucket

If you want to remove a previously-set public read policy from a bucket, and revert to its default policy that requires authentication on every object access, run the following command:

aws **mc** **s3cmd**

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

Last update: 2023-01-11
Created: 2023-01-11
Authors: Florian Haas