

Generic secret storage

The simplest way to use Barbican is to create and retrieve a securely stored, generic secret.

How to store a generic secret

It is possible to store any secret data with Barbican. The command below will create a secret of the type `passphrase`, named `mysecret`, which contains the passphrase `my very secret passphrase`.

```
openstack secret store \
  --secret-type passphrase \
  -p "my very secret passphrase" \
  -n mysecret
```

The example output below uses Cleura's `Fra1` region. In other regions, the secret `URIs` will differ.

```
+-----+-----+
| Field      | Value                                                                 |
+-----+-----+
| Secret href | https://fra1.citycloud.com:9311/v1/secrets/33ef0985-f89e-4bf0-
b318-887ecac0cba |
| Name        | mysecret                                                             |
| Created     | None                                                                  |
| Status      | None                                                                  |
| Content types | None                                                                    |
| Algorithm    | aes                                                                    |
| Bit length   | 256                                                                    |
| Secret type  | passphrase                                                            |
| Mode        | cbc                                                                    |
| Expiration   | None                                                                    |
+-----+-----+
```

Note that `passphrase` type secrets are symmetrically encrypted, using the `AES` encryption algorithm with a 256-bit key length. You can select other bit lengths and algorithms with the `-b` and `-a` command line options, if desired.

How to retrieve secrets

Secrets are stored in Barbican in an encrypted format. You can see a list of secrets created for your user with the following command:

openstack secret list

```
+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
| Secret href                                     | Name   | Created           |
| Status | Content types                               | Algorithm | Bit length | Secret type | Mode |
| Expiration |
+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
| https://fra1.citycloud.com:9311/v1/secrets/33ef0985-f89e-4bf0-b318-887ecac0cba |
mysecret | 2021-04-29T10:33:18+00:00 | ACTIVE | {'default': 'application/octet-stream'} |
aes      |      256 | passphrase | cbc | None      |
| https://fra1.citycloud.com:9311/v1/secrets/ad628532-53b8-4d2f-91e5-0097b51da4e |
None     | 2021-04-27T13:52:10+00:00 | ACTIVE | {'default': 'application/octet-stream'} |
aes      |      256 | symmetric  | None | None      |
+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
```

You can retrieve the decrypted secret with the `openstack secret get` command, adding the `-p` (or `--payload`) option:

```
$ openstack secret get -p \
https://fra1.citycloud.com:9311/v1/secrets/33ef0985-f89e-4bf0-b318-887ecac0cba
```

```
+-----+-----+
| Field | Value           |
+-----+-----+
| Payload | my very secret passphrase |
+-----+-----+
```

Unlike many other OpenStack services, which allow you to retrieve object references by name or UUID, Barbican only lets you retrieve secrets by their full **URI**. That URI must include the `https://<region>.citycloud.com:9311/v1/secrets/` prefix.

Last update: 2022-04-19

Created: 2022-04-19

Authors: [Florian Haas](#)