Red Hat OpenStack Platform 12

Deploy Fernet on the Overcloud

Deploy Fernet on the Red Hat OpenStack Platform director overcloud

Last Updated: 2018-05-02

Red Hat OpenStack Platform 12 Deploy Fernet on the Overcloud

Deploy Fernet on the Red Hat OpenStack Platform director overcloud

OpenStack Team
rhos-docs@redhat.com

Legal Notice

Abstract

Deploy Fernet on the Red Hat OpenStack Platform director overcloud.

Table of Contents

CHAPTER 1. USING FERNET TOKENS IN THE OVERCLOUD

Fernet is now the default token provider, replacing **uuid**. This guide describes how to review your Fernet deployment, and how to rotate the Fernet keys.

1.1. REVIEW THE FERNET DEPLOYMENT

This procedure reviews your configuration to confirm that Fernet tokens are working correctly.

1. Retrieve the IP address of the controller node.

```
[stack@director ~]$ source ~/stackrc
[stack@director ~]$ openstack server list
+--------------------------------------+------------------------
+--------+---------------------+
| ID                                   | Name
| Status | Networks            |
+--------------------------------------+------------------------
+--------+---------------------+
| 756fbd73-e47b-46e6-959c-e24d7fb71328 | overcloud-controller-0  |
ACTIVE | ctlplane=192.0.2.16 |
| 62b869df-1203-4d58-8e45-fac6cd4cfbee | overcloud-novacompute-0 |
ACTIVE | ctlplane=192.0.2.8  |
+--------------------------------------+------------------------
+--------+---------------------+
```

2. SSH to the controller.

```
[heat-admin@overcloud-controller-0 ~]$ ssh heat-admin@192.0.2.16
```

3. Retrieve the values of the token driver and provider settings.

```
[heat-admin@overcloud-controller-0 ~]$ sudo crudini --get
/var/lib/config-data/puppet-
generated/keystone/etc/keystone/keystone.conf token driver
sql
[heat-admin@overcloud-controller-0 ~]$ sudo crudini --get
/var/lib/config-data/puppet-
generated/keystone/etc/keystone/keystone.conf token provider
fernet
```

4. Test the Fernet provider.

```
[heat-admin@overcloud-controller-0 ~]$ exit
[stack@director ~]$ source ~/overcloudrc
[stack@director ~]$ openstack token issue
+------------+------------------------------------------------------
------------------------------------------------------------------
------------------------------------------------------------------
------+
| Field | Value |
+------------+------------------------------------------------------
------------------------------------------------------------------
------------------------------------------------------------------
```

```
------+
| expires | 2016-09-20 05:26:17+00:00 |
| id |
gAAAAABX4LppE8vaiFZ992eah2i3edpO1aDFxlKZq6a_RJzxUx56QVKORrmW0-oZK3-
Xuu2wcnpYq_eek2SGLz250eLpZOzxKBR0GsoMfxJU8mEFF8NzfLNcbuS-iz7SV-
N1re3XEywSDG90JcgwjQfXW-8jtCm-n3LL5IaZexAYIw059T_-cd8 |
| project_id | 26156621d0d54fc39bf3adb98e63b63d |
| user_id | 397daf32cadd490a8f3ac23a626ac06c |
+------------+-----------------------------------------------------
------------------------------------------------------------------
------------------------------------------------------------------
------+
```

The result should include the long Fernet token.

1.2. ROTATE THE FERNET KEYS

It is recommended that you rotate your Fernet keys regularly, as a compromised keystone key can allow an attacker to generate their own tokens and subsequently grant themselves access to a project.

Fernet uses three types of keys, which are stored in **`/var/lib/config-data/puppet-generated/keystone/etc/keystone/fernet-keys`**. The highest-numbered directory contains the primary key, which is used to generate new tokens and decrypt existing ones.

During the key rotation process, the primary key is relegated to secondary key status, and a new primary key is issued, thereby reducing the value of a compromised primary key. Secondary keys can only be used to decrypt tokens that were created with previous primary keys, and cannot issue new ones.

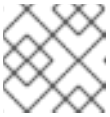
1.2.1. Rotate the Fernet Keys Using Mistral

By default, director is configured to manage the overcloud's Fernet keys; this setting is managed in the environment file using **`ManageKeystoneFernetKeys`**. As a result, the Fernet keys are stored in Mistral (under **`KeystoneFernetKeys`**). This approach means that you can rotate the Fernet keys with Mistral, and they will still persist after stack updates.

1. Review the existing Fernet keys.

   a. Identify the Fernet key location.

   ```
   # SSH back to the controller
   [heat-admin@overcloud-controller-0 ~]$ sudo crudini --get
   /var/lib/config-data/puppet-
   generated/keystone/etc/keystone/keystone.conf fernet_tokens
   key_repository
   /etc/keystone/fernet-keys
   ```

   NOTE

   The **`/etc/keystone/`** directory refers to the container file system path.

   b. Review the current Fernet key directories.

```
[heat-admin@overcloud-controller-0 ~]$ sudo ls /var/lib/config-
data/puppet-generated/keystone/etc/keystone/fernet-keys
0  1  2
```

- **0** - Contains the *staged* key, (which becomes the next primary key) and will always be numbered **0**.
- **1** - Contains the *secondary* key.
- **2** - Contains the *primary* key. This number will increment each time the keys are rotated, with the highest number always serving as the primary key.

NOTE

- The maximum number of keys is determined by the max_active_keys property, by default 5 keys.
- The keys are propagated across all controllers.

2. Rotate the Fernet keys using the Mistral workflow.

```
[stack@director ~]$ source ~/stackrc
[stack@director ~]$ openstack workflow execution create
tripleo.fernet_keys.v1.rotate_fernet_keys '{"container":
"overcloud"}'
+-------------------+----------------------------------------+
| Field             | Value                                  |
+-------------------+----------------------------------------+
| ID                | 58c9c664-b966-4f82-b368-af5ed8de5b47   |
| Workflow ID       | 78f0990a-3d34-4bf2-a127-10c149bb275c   |
| Workflow name     | tripleo.fernet_keys.v1.rotate_fernet_keys |
| Description       |                                        |
| Task Execution ID | <none>                                 |
| State             | RUNNING                                |
| State info        | None                                   |
| Created at        | 2017-12-20 11:13:50                    |
| Updated at        | 2017-12-20 11:13:50                    |
+-------------------+----------------------------------------+
```

3. Get the ID and ensure that the workflow was executed successfully.

```
[stack@director ~]$ openstack workflow execution show 58c9c664-b966-
4f82-b368-af5ed8de5b47
+-------------------+----------------------------------------+
| Field             | Value                                  |
+-------------------+----------------------------------------+
| ID                | 58c9c664-b966-4f82-b368-af5ed8de5b47   |
| Workflow ID       | 78f0990a-3d34-4bf2-a127-10c149bb275c   |
| Workflow name     | tripleo.fernet_keys.v1.rotate_fernet_keys |
| Description       |                                        |
| Task Execution ID | <none>                                 |
| State             | SUCCESS                                |
| State info        | None                                   |
```

```
| Created at        | 2017-12-20 11:13:50                          |
| Updated at        | 2017-12-20 11:15:00                          |
+-------------------+----------------------------------------------+
```

4. On the controller, review the number of Fernet keys, and compare with the previous result.

```
[heat-admin@overcloud-controller-0 ~]$ sudo ls /var/lib/config-
data/puppet-generated/keystone/etc/keystone/fernet-keys
0  1  2  3
```

- **0** - Contains the *staged* key, and will always be numbered **0**. This key will be promoted to a primary key during the next rotation.
- **1 & 2** - Contain the *secondary* keys.
- **3** - Contains the *primary* key. This number will increment each time the keys are rotated, with the highest number always serving as the primary key.

NOTE

- The maximum number of keys is determined by the max_active_keys property, by default 5 keys.
- The keys are propagated across all controllers.