

[controller1]安裝 keystone(驗證)服務

終於要來安裝 controller1 最重要的服務了，也就是 CONTROLLER 所以存在的意義，那就是 OpenStack 整個六大元件中的元件「驗證服務(keystone)」，並且以 keystone-manage 建立管理的服務實體(service entity)與三個 API 端點...等，請參考下列步驟：

安裝 keystone

```
apt-get install keystone
:
python-testtools python-traceback2 python-unittest2 python-webob
python-zope.interface ssl-cert xmlsec1
0 upgraded, 82 newly installed, 0 to remove and 3 not upgraded.
Need to get 7,944 kB of archives.
After this operation, 45.0 MB of additional disk space will be used.
#以上有安裝、更新套件...，要繼續嗎？按下「Y」繼續
Do you want to continue? [Y/n] Y
:
2016-10-27 22:50:32.665 10527 INFO migrate.versioning.api [-] 2 -> 3...
2016-10-27 22:50:32.706 10527 INFO migrate.versioning.api [-] done
2016-10-27 22:50:32.707 10527 INFO migrate.versioning.api [-] 3 -> 4...
2016-10-27 22:50:32.736 10527 INFO migrate.versioning.api [-] done
apache2_invoke: Enable site keystone.conf
Processing triggers for libc-bin (2.23-0ubuntu4) ...
Processing triggers for systemd (229-4ubuntu1) ...
Processing triggers for ureadahead (0.100.0-19) ...
Processing triggers for ufw (0.35-0ubuntu2) ...
```

附註：雖僅安裝 keystone，但也一併裝了如：

apache2、libapache2-mod-wsgi、python-
keystone...等重要工具。

編輯 keystone 組態檔

```
vim /etc/keystone/keystone.conf
614 [database]
642 #connection = sqlite:///var/lib/keystone/keystone.db
643 connection = mysql+pymysql://keystone:KEYSTONE_DBPASS@controller1/keystone
:
2612 [token]
2658 provider = fernet
```

使用 **keystone-manage** 對資料庫做同步化(建立資料表、欄位、基礎資料...等)

```
keystone-manage db_sync
```

```
:
2016-10-27 23:56:01.975 11746 INFO migrate.versioning.api [-] 3 -> 4...
2016-10-27 23:56:01.979 11746 INFO migrate.versioning.api [-] done
2016-10-27 23:56:01.997 11746 INFO migrate.versioning.api [-] 0 -> 1...
2016-10-27 23:56:02.007 11746 INFO migrate.versioning.api [-] done
2016-10-27 23:56:02.007 11746 INFO migrate.versioning.api [-] 1 -> 2...
2016-10-27 23:56:02.113 11746 INFO migrate.versioning.api [-] done
2016-10-27 23:56:02.113 11746 INFO migrate.versioning.api [-] 2 -> 3...
2016-10-27 23:56:02.243 11746 INFO migrate.versioning.api [-] done
2016-10-27 23:56:02.243 11746 INFO migrate.versioning.api [-] 3 -> 4...
2016-10-27 23:56:02.365 11746 INFO migrate.versioning.api [-] done
```

初始化 **Fernet Key** 資料(以下指令為一行，\ 為換行符號)

```
keystone-manage fernet_setup --keystone-user keystone --keystone-group \
keystone
```

```
:
2016-10-28 00:00:21.899 11806 INFO keystone.common.fernet_utils [-] Next primary
key will be: 1
2016-10-28 00:00:21.900 11806 INFO keystone.common.fernet_utils [-] Promoted key 0
to be the primary: 1
2016-10-28 00:00:21.900 11806 INFO keystone.common.fernet_utils [-] Created a new
key: /etc/keystone/fernet-keys/0
```

初始化有關 **Credential** 憑證資料(以下指令為一行，\ 為換行符號)

```
keystone-manage credential_setup --keystone-user keystone \
--keystone-group keystone
```

```
2016-10-28 00:03:41.129 11851 INFO keystone.common.fernet_utils [-] Starting key
rotation with 1 key files: ['/etc/keystone/credential-keys/0']
2016-10-28 00:03:41.129 11851 INFO keystone.common.fernet_utils [-] Current primary
key is: 0
2016-10-28 00:03:41.130 11851 INFO keystone.common.fernet_utils [-] Next primary
key will be: 1
2016-10-28 00:03:41.130 11851 INFO keystone.common.fernet_utils [-] Promoted key 0
to be the primary: 1
2016-10-28 00:03:41.130 11851 INFO keystone.common.fernet_utils [-] Created a new
key: /etc/keystone/credential-keys/0
```

建立 keystone 的 admin 認證服務端點 (以下指令為一行，\ 為換行符號)

```
keystone-manage bootstrap --bootstrap-password ADMIN_PASS \  
--bootstrap-admin-url http://controller1:35357/v3/ \  
--bootstrap-internal-url http://controller1:35357/v3/ \  
--bootstrap-public-url http://controller1:5000/v3/ \  
--bootstrap-region-id RegionOne
```

```
2016-10-28 00:12:36.729 11963 WARNING keystone.assignment.core [-] Deprecated: Use  
of the identity driver config to automatically configure the same assignment driver  
has been deprecated, in the "0" release, the assignment driver will need to be  
explicitly configured if different than the default (SQL).
```

```
2016-10-28 00:12:37.354 11963 INFO keystone.cmd.cli [-] Created domain default
```

```
2016-10-28 00:12:37.385 11963 INFO keystone.cmd.cli [req-89583ddc-b3b2-4639-b453-  
05baf5ffcecf - - - - -] Created project admin
```

```
2016-10-28 00:12:37.407 11963 INFO keystone.cmd.cli [req-89583ddc-b3b2-4639-b453-  
05baf5ffcecf - - - - -] Created user admin
```

```
2016-10-28 00:12:37.411 11963 INFO keystone.cmd.cli [req-89583ddc-b3b2-4639-b453-  
05baf5ffcecf - - - - -] Created role admin
```

```
2016-10-28 00:12:37.427 11963 INFO keystone.cmd.cli [req-89583ddc-b3b2-4639-b453-  
05baf5ffcecf - - - - -] Granted admin on admin to user admin.
```

```
2016-10-28 00:12:37.432 11963 INFO keystone.cmd.cli [req-89583ddc-b3b2-4639-b453-  
05baf5ffcecf - - - - -] Created region RegionOne
```

```
2016-10-28 00:12:37.448 11963 INFO keystone.cmd.cli [req-89583ddc-b3b2-4639-b453-  
05baf5ffcecf - - - - -] Created admin endpoint http://controller1:35357/v3/
```

```
2016-10-28 00:12:37.455 11963 INFO keystone.cmd.cli [req-89583ddc-b3b2-4639-b453-  
05baf5ffcecf - - - - -] Created internal endpoint http://controller1:35357/v3/
```

```
2016-10-28 00:12:37.482 11963 INFO keystone.cmd.cli [req-89583ddc-b3b2-4639-b453-  
05baf5ffcecf - - - - -] Created public endpoint http://controller1:5000/v3/
```

[controller1]設定 apache(網頁)服務

由於 keystone 的驗證需要借助 apache，故我們藉由剛剛安裝 keystone 時，已經一起安裝了 apache，所以我們也需要針對 apache 做一些基礎設定：

編輯 apache 組態檔

```
vim /etc/apache2/apache2.conf
```

```
57 ServerName controller1
```

重新啟動 apache

```
service apache2 restart
```

刪除預設 SQLite 存在的 keystone 資料庫

```
rm -f /var/lib/keystone/keystone.db
```

[controller1]建立 admin(管理)帳號環境

接下來我們要為 keystone 建立一個管理者身份的環境，因為是暫時性的，故使用環境變數的方式建立，請參考下列步驟：

於終端機建立 OpenStack admin 管理環境(環境變數)：管理者名稱

```
export OS_USERNAME=admin
```

於終端機建立 OpenStack admin 管理環境(環境變數)：管理者密碼

```
export OS_PASSWORD=ADMIN_PASS
```

於終端機建立 OpenStack admin 管理環境(環境變數)：專案名稱

```
export OS_PROJECT_NAME=admin
```

於終端機建立 OpenStack admin 管理環境(環境變數)：網域名稱

```
export OS_USER_DOMAIN_NAME=Default
```

於終端機建立 OpenStack admin 管理環境(環境變數)：驗證位置

```
export OS_AUTH_URL=http://controller1:35357/v3
```

於終端機建立 OpenStack admin 管理環境(環境變數)：專案網域名稱

```
OS_PROJECT_DOMAIN_NAME=Default
```

於終端機建立 OpenStack admin 管理環境(環境變數)：驗證的 API 版本

```
export OS_IDENTITY_API_VERSION=3
```

[controller1]建立其他專案、帳號與角色

我們知道 keystone 是提供 OpenStack 各項服務的驗證服務，而驗證使用了 Domain、Project、User、Role，而我們剛剛僅使用了「keystone-manage bootstrap...」幫我們建立了管理者(admin)相關的內容，因為我們往後還有很多服務要建立，以及還要建立一般使用者，所以接下來我們一併建立起來：

建立 keystone project 的 service (以下指令為一行，\ 為換行符號)

```
openstack project create --domain default --description \
"Service Project" service
```

Field	Value
description	Service Project
domain_id	default
enabled	True
id	228783d54a2f4eaf9cec498550d41fae
is_domain	False
name	service
parent_id	default

建立 keystone project 的 demo (以下指令為一行，\ 為換行符號)

```
openstack project create --domain default --description \
"Demo Project" demo
```

Field	Value
description	Demo Project
domain_id	default
enabled	True
id	1be4ac087d564d9c85071a9a5bd9e434
is_domain	False
name	demo
parent_id	default

建立 keystone user 的 demo (需設定 demo 的密碼)

```
openstack user create --domain default --password-prompt demo
```

User Password: DEMO_PASS

Repeat User Password: DEMO_PASS

Field	Value
domain_id	default
enabled	True
id	b66059dd26f942dc891727a9642478d2
name	demo
password_expires_at	None

建立 keystone role 的 user

```
openstack role create user
```

Field	Value
domain_id	None
id	1e5b7fe3fe87490bb1757d617ae7f908
name	user

將 role:user 加入 Project:demo 與 user:demo

```
openstack role add --project demo --user demo user
```

附註：如果您的規畫中有更多的專案、帳號或角色，可以重複本段落的步驟。

[controller1]驗證與測試

接下來要來驗證我們剛剛建立 keystone 是否可以正確驗證與取得 token，以及我們所建立的 project、user、role...等是否正確。

[controller1]驗證安全性設定

另外依照 OpenStack 的建議，因為安全性問題，我們應該移除 OpenStack 預設的臨時 token 與存在我們終端機的管理者密碼，請參考下列步驟：

編輯 keystone-paste.ini 組態

```
vim /etc/keystone/keystone-paste.ini
```

#移除 [pipeline:public_api]、[pipeline:admin_api] 與 [pipeline:api_v3] 段落中的 admin_token_auth 字串

```
57 [pipeline:public_api]
```

```
60 pipeline = cors sizelimit http_proxy_to_wsgi osprofiler url_normalize
    request_id admin_token_auth build_auth_context token_auth json_body
    ec2_extension public_service
```

```
62 [pipeline:admin_api]
```

```
65 pipeline = cors sizelimit http_proxy_to_wsgi osprofiler url_normalize
    request_id build_auth_context token_auth json_body ec2_extension
    s3_extension admin_service
```

```
67 [pipeline:api_v3]
```

```
70 pipeline = cors sizelimit http_proxy_to_wsgi osprofiler url_normalize
    request_id build_auth_context token_auth json_body ec2_extension_v3
    s3_extension service_v3
```

註銷管理者密碼的環境變數

```
unset OS_PASSWORD
```