

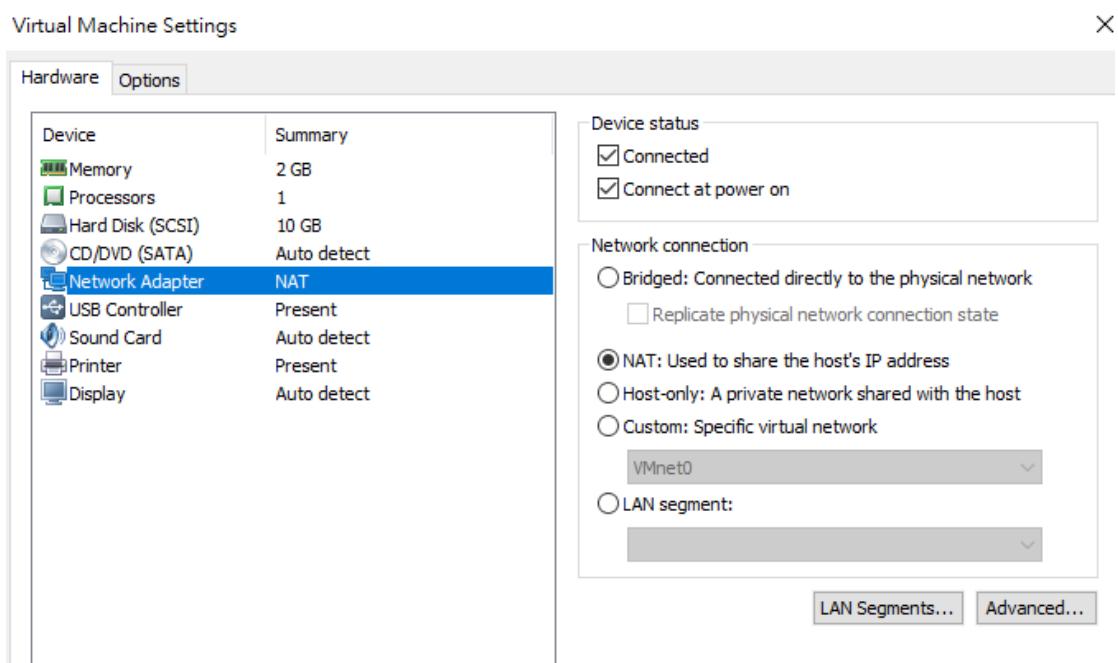
Devstack 建立 1 個 controller + 1 個 node01

1. 安裝 Ubuntu 18.04

Controller Disk 40GB Memory 8192MB

Node Disk 20GB Memory 4048MB

2. 設定網路為 NAT



3. 開啟 terminal，更新線上套件庫資訊，執行系統更新、安裝 Git、net-tools:

```
sudo apt-get update  
sudo apt-get upgrade  
sudo apt -y dist-upgrade  
sudo apt-get install -y git  
sudo apt-get install -y net-tools
```

4.1 ===== 建立 controller

修改 Host name (controller)

```
sudo gedit /etc/hosts
```

```
127.0.0.1      localhost  
127.0.1.1      controller .
```

```
sudo gedit /etc/hostname
```

```
controller  
|~
```

Controller 就設定為 controller

設定完後重開機，開啟 terminal 輸入 hostname 確認是否有更改

4.2 Host name (node01) =====建立 node01

5. 開啟 terminal 並切換到 root:

```
sudo su
```

6. 新增使用者 Stack (Devstack 只能在一般使用者上安裝)

```
useradd -s /bin/bash -d /opt/stack -m stack
```

7. 將使用者 Stack 設定為不需要密碼就能使用 RootS

```
echo "stack ALL=(ALL) NOPASSWD: ALL" | sudo tee /etc/sudoers.d/stack
```

8. 切換到使用者 Stack

```
sudo su - stack
```

9. 在 Stack 使用者下設定 SSH(Stack 訪問其他節點時，會透過 SSH 進行認證)

```
mkdir ~/.ssh
```

```
chmod 700 ~/.ssh
```

```
echo "ssh-rsa
```

```
AAAAB3NzaC1yc2EAAAQABAAQCyYjfgyPazTvGpd8OaAvtU2ut  
L8W6gWC4JdRS1J95GhNNfQd657yO6s1AH5KYQWktcE6FO/xNUC2reEXS  
GC7ezy+sGO1kj9Limv5vrvNHvF1+wts0Cmyx61D2nQw35/Qz8BvpdJANL7V  
wP/cFI/p3yhvx2lsnjFE3hN8xRB2LtLUopUSVdBwACOVUmH2G+2BWMJDj  
VINd2DPqRIA4Zhy09KJ3O1Joabr0XpQL0yt/I9x8BVHdAx6l9U0tMg9dj5+tAj  
ZvMAFFye3PJcYwwsfJoFx8w/SLtqlFX7Ehw++8RtvomvuipldmWCy+T9hIkI  
+gHYE4cS3OIqXH7f49jdJf jykuo@ntut.edu.tw > ~/.ssh/authorized_keys
```

10. 下載 Devstack

```
git clone https://opendev.org/openstack/devstack.git -b stable/train
```

11. 切換到 Devstack

```
cd devstack
```

12. 複製 local.conf 文件到目錄下，並修改

```
cp ./samples/local.conf ./
```

13. ifconfig 查看目前 ip(ens33 interface) ,

14.1 =====建立 controller

編輯設定檔案，sudo gedit local.conf，找到[local|localrc]後，在底下加入

```
[[local|localrc]]  
HOST_IP=192.168.6.130 # change this controller node IP  
FLAT_INTERFACE=ens33 # ens33(ubuntu16)  
FIXED_RANGE=10.4.128.0/20 # unchange  
FIXED_NETWORK_SIZE=4096 # unchange  
FLOATING_RANGE=192.168.8.128/26 # unchange  
MULTI_HOST=1  
  
ADMIN_PASSWORD=labstack  
DATABASE_PASSWORD=stackdb  
RABBIT_PASSWORD=stackqueue  
SERVICE_PASSWORD=$ADMIN_PASSWORD  
  
MYSQL_HOST=$HOST_IP  
RABBIT_HOST=$HOST_IP  
GLANCE_HOSTPORT=$HOST_IP:9292
```

```
# Note that if ``localrc`` is present it will be used in favor of this section.  
[[local|localrc]]  
  
HOST_IP=192.168.6.136 # change this controller node IP  
FLAT_INTERFACE=ens33 # ens33(ubuntu16)  
FIXED_RANGE=10.4.128.0/20 # unchange  
FIXED_NETWORK_SIZE=4096 # unchange  
FLOATING_RANGE=192.168.8.128/26 # unchange  
MULTI_HOST=1  
ADMIN_PASSWORD=labstack # change passwd  
DATABASE_PASSWORD=stackdb  
RABBIT_PASSWORD=stackqueue  
SERVICE_PASSWORD=$ADMIN_PASSWORD  
MYSQL_HOST=$HOST_IP  
RABBIT_HOST=$HOST_IP  
GLANCE_HOSTPORT=$HOST_IP:9292  
  
# Minimal Contents  
# -----  
# While ``stack.sh`` is happy to run without ``localrc``, devlife is better when
```

PS. 要刪除後面的

ADMIN_PASSWORD=...
DATABASE_PASSWORD=...
RABBIT_PASSWORD=...
SERVICE_PASSWORD

14.2 =====建立 node01

編輯設定檔案，sudo gedit local.conf，找到[local|localrc]後，在底下加入
[[local|localrc]]

```
HOST_IP=192.168.6.135 # change this compute node IP
FLAT_INTERFACE=ens33 #ens33(ubuntu16)
FIXED_RANGE=10.4.128.0/20 #unchange
FIXED_NETWORK_SIZE=4096 #unchange
FLOATING_RANGE=192.168.8.128/26 #unchange
MULTI_HOST=1
LOGFILE=/opt/stack/logs/stack.sh.log
ADMIN_PASSWORD=labstack #same with controller
DATABASE_PASSWORD=stackdb
RABBIT_PASSWORD=stackqueue
SERVICE_PASSWORD=$ADMIN_PASSWORD
DATABASE_TYPE=mysql
SERVICE_HOST=192.168.6.136 # change same controller IP
MYSQL_HOST=$SERVICE_HOST
RABBIT_HOST=$SERVICE_HOST
GLANCE_HOSTPORT=$SERVICE_HOST:9292
ENABLED_SERVICES=n-cpu,q-agt,n-api-meta,c-vol,placement-client
NOVA_VNC_ENABLED=True
NOVNCPROXY_URL="http://$SERVICE_HOST:6080/vnc_auto.html"
VNCSERVER_LISTEN=$HOST_IP
VNCSERVER_PROXYCLIENT_ADDRESS=$VNCSERVER_LISTEN
```

```
[[local|localrc]]
HOST_IP=192.168.6.135      # change this compute node IP
FLAT_INTERFACE=ens33        #ens33(ubuntu16)
FIXED_RANGE=10.4.128.0/20   #unchange
FIXED_NETWORK_SIZE=4096    #unchange
FLOATING_RANGE=192.168.8.128/26  #unchange
MULTI_HOST=1
LOGFILE=/opt/stack/logs/stack.sh.log
ADMIN_PASSWORD=labstack    # same with controller
DATABASE_PASSWORD=stackdb
RABBIT_PASSWORD=stackqueue
SERVICE_PASSWORD=$ADMIN_PASSWORD
DATABASE_TYPE=mysql
SERVICE_HOST=192.168.6.136 # change same controller IP
MYSQL_HOST=$SERVICE_HOST
RABBIT_HOST=$SERVICE_HOST
GLANCE_HOSTPORT=$SERVICE_HOST:9292
ENABLED_SERVICES=n-cpu,q-agt,n-api-meta,c-vol,placement-client
NOVA_VNC_ENABLED=True
NOVNCPROXY_URL="http://$SERVICE_HOST:6080/vnc_auto.html"
VNCSERVER_LISTEN=$HOST_IP
VNCSERVER_PROXYCLIENT_ADDRESS=$VNCSERVER_LISTEN
|
# Minimal Contents
# -----
```

15. 修正擁有者

```
sudo chown stack: /opt/stack/.cache
```

```
sudo chown -R stack:stack /opt/stack
```

16. 安裝 Devstack

```
./stack.sh
```

17. 修正問題

```
sudo chown stack: /opt/stack/.config
```

```
virtualenv ..../requirements/.venv/
```

18. 安裝完成

建立網路

1. 在電腦上打開瀏覽器，網址輸入 Controller 的 IP，就可以連結到 openstack 的 dashboard。
2. 輸入帳號，帳號為 admin，密碼是之前 local.conf 中所設定的 ADMIN_PASSWORD



3. 選擇路由器並新增路由器

openstack demo admin

專案 API 存取權 連算 雲硬碟 網路 路由器 路由器名稱 = ▾ 篩選 + 新增路由器 剪除路由器

顯示 2 項 網路拓撲 網路 路由器 名稱 狀態 對外網路 管理員狀態 可用區域 動作

<input type="checkbox"/> test	使用中	-	上線	nova	設定閘道 ▾
<input type="checkbox"/> router1	使用中	public	上線	nova	清除閘道 ▾

顯示 2 項 安全性群組 浮動 IP 管理員 身份

4. 輸入 Router name，選擇對外網路為 public，然後按下新增路由器

新增路由器 ×

路由器名稱 : router

啟用管理員狀態

對外網路 : public

Enable SNAT

Availability Zone Hints ⓘ

nova

取消 新增路由器

5. 選擇網路並按新增網路

openstack demo admin

專案 專案 / 網路 / 網路

API 存取權

運算 >

雲硬碟 >

網路 > 網路

顯示 2 項

	名稱	子網路已聯結	已分享	對外	狀態	管理員狀態	可用區域	動作
<input type="checkbox"/>	private	ipv6-private-subnet fd5a:f74a:57b2::/64 private-subnet 10.0.0.0/26	No	No	使用中	上線	nova	<button>編輯網路</button>
<input type="checkbox"/>	public	ipv6-public-subnet 2001:db8::/64 public-subnet 192.168.42.128/25	No	Yes	使用中	上線	nova	<button>編輯網路</button>

6. 輸入 network 名稱

新增網路

網路 子網域 子網路詳細資訊

網路名稱

network

啟用管理員狀態 ?

已分享

新增子網路

Availability Zone Hints ?

nova

取消 < 上一頁 下一頁 >

新增網路

[網路](#)[子網域](#)[子網路詳細資訊](#)

子網路名稱

網路位址來源

手動輸入網路位址

網路位址 ?

192.168.0.0/24

IP 版本

IPv4

閘道 IP ?

192.168.0.1|

 關閉閘道[取消](#)[« 上一頁](#)[下一個 »](#)

新增網路

[網路](#)[子網域](#)[子網路詳細資訊](#) 啟用 DHCP

為子網路指定額外的屬性。

分配倉庫 ?

DNS 名稱伺服器 ?

8.8.8.8|

主機路由 ?

[取消](#)[« 上一頁](#)[新增](#)

7. 建立 interface 網路介面卡

The screenshot shows the OpenStack interface for managing routers. In the left sidebar, '路由器' (Router) is selected under the '網路' (Network) category. The main area displays a list of routers with the following details:

名稱	狀態	對外網路	管理員狀態	可用區域	動作
test	使用中	-	上線	nova	設定閘道
router1	使用中	public	上線	nova	清除閘道
router	使用中	public	上線	nova	清除閘道

The screenshot shows the detailed view of a specific router named 'router'. Under the '網路卡' (Network Interface) tab, there is a table listing network interfaces:

名稱	固定 IP	狀態	類型	管理員狀態	動作
(4a22ab41-dffb)	• 192.168.42.130 • 2001:db8::4	使用中	對外閘道	上線	刪除網路卡

選擇剛剛建立的 network

加入網路卡

子網域 *

network: 192.168.0.0/24 (d5e425a1-5407-4e1...

IP 位址 (選填) ?

描述 :

您可以連接指定的子網路到路由器。

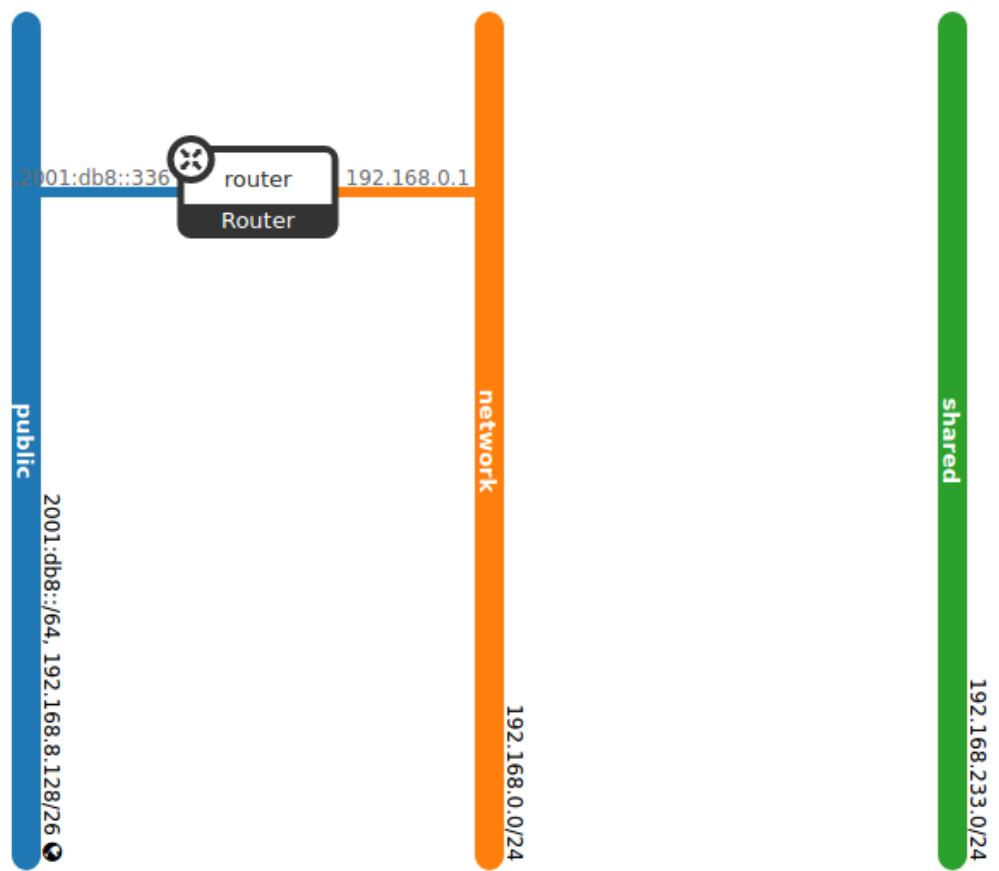
這裡如果不指定一個IP地址，則會使用被選定子網路的閘道器地址作為路由器上新建接口的IP地址。如果閘道器IP地址已經被使用，你必須使用選定子網路的其它地址。

取消

提交

8. 查看網路拓譜 Network Topology

可以看到剛剛建立的 network 與 public 連接在一起



建立 Instance

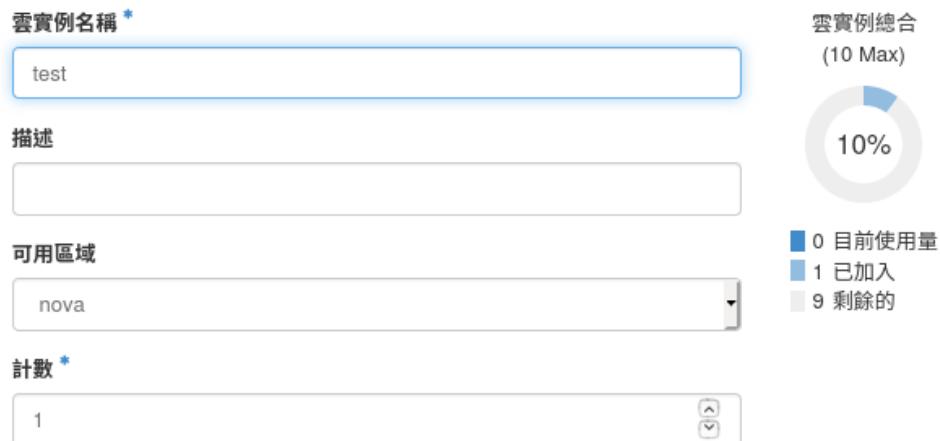
1. 雲實例>發動雲實例



The screenshot shows the OpenStack dashboard interface. The top navigation bar includes the OpenStack logo, project name 'demo', and user 'admin'. Below the navigation, there are dropdown menus for '專案', 'API 存取權', and '運算'. The '運算' menu is expanded, showing '雲實例' (highlighted with a red box). The main content area is titled '雲實例' and has a sub-section '概觀'. A table header for '雲實例' is shown with columns: 映像檔名稱, 映像檔名稱, IP 位址, 虛擬硬體樣板, 密鑰對, 狀態, 可用區域, 任務, 電源狀態, 壽命, 動作. A search bar '實例 ID = ' and a filter button '篩選' are at the top of the table. A large red box highlights the '發動雲實例' (Launch Instance) button. Below the table, it says '沒有項目可以列出'.

2.

請提供雲實例的初始主機名稱、將用來部署的可用區域以及雲實例計數。增加「計數」以使用相同的設定新增多個雲實例。



The screenshot shows the 'Create New Instance' form. It includes fields for '雲實例名稱*' (test), '描述' (empty), '可用區域' (nova), and '計數*' (1). To the right, there is a circular progress bar labeled '雲實例總合 (10 Max)' showing '10%' completion. A legend indicates the colors for the progress bar segments: dark blue for '0 目前使用量', medium blue for '1 已加入', and light grey for '9 剩餘的'.

3. 新增雲硬碟選擇”否”，並且選擇映像檔(本次使用預設)

發動雲實例

詳細資訊

來源

虛擬硬體樣板 *

網路 *

網路埠

安全性群組

密鑰對

設置

伺服器群組

Scheduler Hints

註釋資料

雲實例來源是用來新增雲實例的範本。您可以使用已存在雲實例的即時存檔、雲硬碟或雲硬碟即時存檔（若已啟用）。您也可以選擇新增新的雲硬碟來使用持續性硬碟。

選擇開機來源

新增新雲硬碟

映像檔

是 否

已分配

名稱	已更新	容量	類型	可見性
> cirros-0.4.0-x86_64-disk	2/19/20 10:26 PM	12.13 MB	qcow2	公開

可用 0

Click here for filters or full text search.

名稱	已更新	容量	類型	可見性
無可用的項目				

取消

上一步

下一步 >

發動雲實例

4. 選擇 flavor 虛擬硬碟樣板

詳細資訊

來源

虛擬硬體樣板

網路 *

網路埠

安全性群組

密鑰對

設置

伺服器群組

Scheduler Hints

註釋資料

虛擬硬體樣板管理雲實例的運算、記憶體及儲存容量規格。

已分配

名稱	虛擬處理器	隨機存取記憶體	硬碟總合	根硬碟	暫時性硬碟	公開
> m1.tiny	1	512 MB	1 GB	1 GB	0 GB	是

可用 11

Click here for filters or full text search.

名稱	虛擬處理器	隨機存取記憶體	硬碟總合	根硬碟	暫時性硬碟	公開
> m1.nano	1	64 MB	1 GB	1 GB	0 GB	是
> m1.micro	1	128 MB	1 GB	1 GB	0 GB	是
> cirros256	1	256 MB	1 GB	1 GB	0 GB	是
> ds512M	1	512 MB	5 GB	5 GB	0 GB	是
> ds1G	1	1 GB	10 GB	10 GB	0 GB	是
> m1.small	1	2 GB	20 GB	20 GB	0 GB	是
> ds2G	2	2 GB	10 GB	10 GB	0 GB	是
> m1.medium	2	4 GB	40 GB	40 GB	0 GB	是

5. 選擇網路

發動雲實例

詳細資訊

來源

虛擬硬體樣板

網路

網路埠

安全性群組

密鑰對

設置

伺服器群組

Scheduler Hints

註釋資料

網路為雲實例提供雲中的通訊頻道。

▼ 已分配 ①

從下列列表中選擇網路。

網路	子網路已聯結	已分享	管理員狀態	狀態
network	56c3ccbe-9a4f-4db1-85a1-13809ec487d4	否	上線	使用中

▼ 可用 ②

選擇至少一個網路

網路	子網路已聯結	已分享	管理員狀態	狀態
public	public-subnet ipv6-public-subnet	否	上線	使用中
shared	shared-subnet	是	上線	使用中

取消 < 上一步 下一步 > 發動雲實例

6. 按下右下角的 發動雲實例
7. 按下剛建立好的虛擬機 test

雲實例

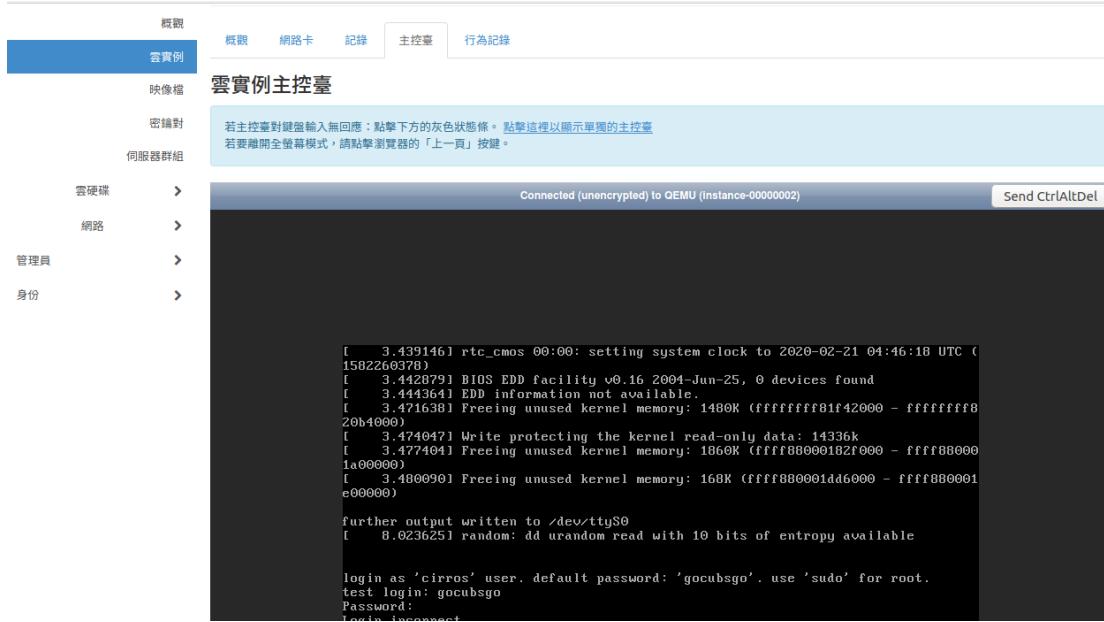
實例 ID = ▾ 設選 發動雲實例 刪除實例 更多動作 ▾

顯示 1 項

雲實例名稱	映像檔名稱	IP 位址	虛擬硬體樣板	密鑰對	狀態	可用區域	任務	電源狀態	時間	動作
test	cirros-0.4.0-x86_64-disk	192.168.0.13	m1.tiny	-	使用中	nova	無	正在執行	2 minutes	新增即時存檔 ▾

顯示 1 項

8. 按下主控台，就可以看到虛擬機



如果遇到下列問題，請輸入指令後再執行 ./stack.sh

Permission denied: '/opt/stack/.cache/pip/ 類型

Solve:

沒有權限 目前使用者是 stack 因此把資料夾權限給 stack

sudo chown stack: /opt/stack/.cache

[ERROR] /opt/stack/devstack/functions-common:601 git call failed: [git clone
git://git.openstack.org/openstack/glance.git /opt/stack/glance --branch master]

Error on exit

Solve:

修改 devstack 資料夾底下的 stackrc

vim ./devstack/stackrc

找到

GIT_BASE=\${GIT_BASE:-git://git.openstack.org}

修改成

GIT_BASE=\${GIT_BASE:-https://git.openstack.org}

或者

GIT_BASE=\${GIT_BASE:-http://git.openstack.org}

[ERROR] /opt/stack/devstack/lib/glance:358 g-api did not start

Solve:

./unstack.sh ./clean.sh rm -rf devstack reboot git clone devstack code cd devstack ./stack.sh

建立多個 node 後，在 node 上建立 instance 時會發生：

Host 'compute node' is not mapped to any cell

Solve:

在 controller 上開啟 terminal 並輸入

nova-manage cell_v2 discover_hosts --verbose

就可以建立 instance

[ERROR] ./stack.sh:227 If you wish to run this script anyway run with

FORCE=yes

/opt/stack/devstack/functions-common: line 239: /opt/stack/logs/error.log: No such file or directory

Solve:

指令改成

FORCE=yes ./stack.sh

env:'/opt/stack/requirements/.venv/bin/pip': No such file or directory

Solve:

輸入 virtualenv .../requirements/.venv/

如果出現：

OSError: [Errno 40] Too many levels of symbolic links:

'/opt/stack/requirements/.venv/bin/python3.5'

輸入

rm /opt/stack/requirements/.venv/*

virtualenv .../requirements/.venv/

在 local.conf 中加入

enable_service placement-api

Ex:如果發生其他類型錯誤，請確認 HOST_IP 是否為虛擬機的 IP

Ex:如果需要修正 local.conf 內容，請在修正完後執行以下指令重新安裝

./unstack.sh

./clean.sh

./stack.sh

當 VM 重開機 openstack 登錄錯誤，或使用程式碼操作 openstack 出現 request time out 類錯誤。

(可能 IP 換了)

1. ./clean.sh ./stack.sh