

OpenStack Mitaka for Ubuntu 16.04

部署手册

参考网址：<http://docs.openstack.org/mitaka/install-guide-ubuntu/>

李越男（00383672）
云核心网 NFV 研究部
liyuanan@huawei.com
欢迎交流 OpenStack 相关问题

一、系统配置

- 1、Partition disks 配置：
- 2、检查网络
- 3、更新软件包数据库

二、基本环境配置

- 1、安装 SQL 数据库（MySQL/MariaDB）
- 2、安装消息队列服务（RabbitMQ）
- 3、安装 Memcached

三、身份服务配置（Identity Service - Keystone）

- 1、在 MariaDB（MySQL）中创建 Keystone 数据库
- 2、安装 Keystone
- 3、创建服务实体（Service Entity）和 API 路径（API Endpoints）
- 4、创建域（Domain）、计划（Project）、用户（User）、角色（Role）
- 5、验证 Keystone 组件配置
- 6、身份服务配置（Identity Service - Keystone）总结

四、镜像服务配置（Image Service - Glance）

- 1、在 MariaDB 中创建 glance 数据库
- 2、创建 Glance 服务实体和 API 路径
- 3、安装和配置 Glance 服务组件
- 4、验证 Glance 服务组件配置
- 5、镜像服务配置（Image Service - Glance）总结

五、计算服务配置（Compute Service - Nova）

- 1、在 MariaDB（MySQL）中创建数据库 nova_api 和 nova
- 2、创建计算服务证书和 API 路径
- 3、安装配置 nova 组件
- 4、安装配置 nova-compute 组件
- 5、验证计算服务是否安装正确
- 6、计算服务配置（Compute Service - Nova）总结

六、网络服务配置（Networking Service - Neutron）

- 1、在 MariaDB（MySQL）中创建 neutron 数据库
- 2、创建网络服务证书和 API 路径
- 3、安装配置 neutron-server 服务组件
- 4、安装配置 Neutron 组件
- 5、配置网络服务代理
- 6、验证网络服务是否安装配置正确
- 7、网络服务配置（Networking Service - Neutron）总结

七、创建虚拟网络

- 7.1、Provider Network
- 7.2、Self-Service Network
- 7.3、创建虚拟机模板
- 7.4、生成密钥对
- 7.5、添加安全组规则
- 7.6、制定虚拟机模板
- 7.7、启动实例：Provider 网络
- 7.8、启动实例：Self-Service 网络

一、系统配置

1、Partition disks 配置：

1.1、单独设置分区/dev/sda1，大小 200MB，用于挂载/boot

选择手动配置-->选择硬盘-->创建新的分区表-->选择空闲空间-->新建分区-->设置大小 200MB-->选择主分区-->选择开始位置-->设置文件系统格式 Ext4-->设置挂载点/boot-->设置分区

1.2、设置 LVM，方便扩展存储，挂载/和 swap

配置逻辑卷 LVM-->将修改写入磁盘并配置 LVM-->新建卷组-->卷组名称：vg_system-->按空格键选中剩余空闲空间/dev/sda-->将修改写入磁盘并配置 LVM-->新建逻辑卷-->选择卷组 vg_system-->逻辑卷名：lv_root-->设置逻辑卷大小：91GB-->新建逻辑卷-->选择卷组 vg_system-->逻辑卷名：lv_swap-->设置逻辑卷大小为剩余空间-->完成

1.3、分别对每一个逻辑卷进行配置

磁盘格式：lv_root 选择 ext4，lv_swap 选择 swap area

挂载点：lv_root 选择"/"

最后选择完成分区并将更改写入磁盘，点击确定即可

2、检查网络

ping www.google.cn

若显示错误：ping: unknown host www.google.com，修改 resolv.conf 文件

vi /etc/resolv.conf

nameserver 8.8.8.8

3、更新软件包数据库

apt-get update && apt-get dist-upgrade

reboot

注：重启之后需要重新修改 resolv.conf。

二、基本环境配置

1、安装 SQL 数据库（MySQL/MariaDB）

1.1、安装 MariaDB 和 MySQL

apt-get install mariadb-server python-mysqldb

1.2、为 MySQL 的 root 用户设置密码

mysqladmin -u root password opnfv

注：本文中所有数据库的密码均设定为 opnfv。

1.3、创建文件 openstack.cnf，添加以下信息

vi /etc/mysql/mariadb.conf.d/openstack.cnf

[mysqld]

bind-address = 192.168.34.88

default-storage-engine = innodb

innodb_file_per_table

collation-server = utf8_general_ci

character-set-server = utf8

注：bind-address 用于绑定 MySQL 服务监听地址到 Controller 节点的 Management Network 网口 IP,以便 Compute&Network 节点访问 MySQL 中的 OpenStack 配置信息。

1.4、重启 MySQL 数据库

service mysql restart

1.5、查看端口监听情况

netstat -ntlp | grep 3306

```
root@lyn:~# netstat -ntlp | grep 3306
tcp        0      0 192.168.34.88:3306    0.0.0.0:*        LISTEN      4934/mysqld
```

1.6、开启 MySQL 安全保护措施

mysql_secure_installation

注：测试环境对安全要求较低可忽略该步骤。

1.7、测试 root 用户能否在本机 localhost 正常登录

mysql -root -p

mysql -h localhost -uroot -p

问题：ERROR 1698 (28000): Access denied for user 'root'@'localhost'

解决方案：

sudo mysql -u root

use mysql;

update user set plugin="" where User='root';

flush privileges;

exit;

2、安装消息队列服务（RabbitMQ）

2.1、安装 RabbitMQ

apt-get install rabbitmq-server

2.2、为 OpenStack 添加用户

rabbitmqctl add_user openstack opnfv

注：opnfv 为本人设定的 openstack 用户的密码。

2.3、为 openstack 用户配置读、写访问权限

rabbitmqctl set_permissions openstack ".*" ".*" ".*"

3、安装 Memcached

3.1、安装 Memcached

apt-get install memcached python-memcache

3.2、修改 memcached.conf，设置 Memcached 服务监听地址

vi /etc/memcached.conf，修改内容：-l 192.168.34.88

3.3、重启 Memcached 服务

service memcached restart

3.4、检查服务端口监听是否正常

netstat -natp

```
root@lyn:~# netstat -natp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:25672           0.0.0.0:*               LISTEN      5670/beam.smp
tcp        0      0 192.168.34.88:3306     0.0.0.0:*               LISTEN      4934/mysqld
tcp        0      0 192.168.34.88:11211    0.0.0.0:*               LISTEN      6338/memcached
tcp        0      0 0.0.0.0:4369           0.0.0.0:*               LISTEN      5621/epmd
tcp        0      0 192.168.34.88:53       0.0.0.0:*               LISTEN      3198/named
tcp        0      0 127.0.0.1:53           0.0.0.0:*               LISTEN      3198/named
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      3196/sshd
tcp        0      0 127.0.0.1:953          0.0.0.0:*               LISTEN      3198/named
tcp        0      0 127.0.0.1:41510        127.0.0.1:4369         ESTABLISHED 5670/beam.smp
tcp        0      0 127.0.0.1:4369         127.0.0.1:41510        ESTABLISHED 5621/epmd
tcp        0      0 192.168.34.88:4369     192.168.34.88:56027    TIME_WAIT   -
tcp        0      0 192.168.34.88:22       172.16.10.16:61123     ESTABLISHED 3329/sshd: opnfv [p
tcp6       0      0 :::5672                :::*                   LISTEN      5670/beam.smp
tcp6       0      0 :::4369                :::*                   LISTEN      5621/epmd
tcp6       0      0 :::53                  :::*                   LISTEN      3198/named
tcp6       0      0 :::22                  :::*                   LISTEN      3196/sshd
tcp6       0      0 :::1:953               :::*                   LISTEN      3198/named
```

三、身份服务配置 (Identity Service - Keystone)

1、在 MariaDB (MySQL) 中创建 Keystone 数据库

1.1、以 root 用户登录 MySQL

```
mysql -u root -p
```

1.2、创建数据库 keystone

```
CREATE DATABASE keystone;
```

1.3、授予数据库 keystone 权限

```
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'192.168.34.88' IDENTIFIED BY 'opnfv';
```

```
GRANT ALL PRIVILEGES ON keystone.* TO 'keystone'@'%' IDENTIFIED BY 'opnfv';
```

注：opnfv 为本人设定的密码；上述授权命令中，%则代表了所有的 host 都能远程访问该 mysql，但 MySQL 官方文档指出，%并不包括本地主机，因此需要对本地主机和%都进行授权。

1.4、生成临时管理身份认证令牌 (ADMIN_TOKEN)

```
openssl rand -hex 10
```

注：记住 ADMIN_TOKEN：e8918f6cfd43dc85c412。

2、安装 Keystone

2.1、关闭 keystone 服务

```
vi /etc/init/keystone.override, 添加内容：manual
```

2.2、安装 keystone

```
apt-get install keystone
```

2.3、修改配置文件 keystone.conf

```
vi /etc/keystone/keystone.conf
```

```
[DEFAULT]
```

```
Admin_token = e8918f6cfd43dc85c412
```

```
[database]
```

```
connection = mysql+pymysql://keystone:opnfv@192.168.34.88/keystone
```

```
[token]
```

```
Provider = fernet
```

注：Admin_token = 中，e8918f6cfd43dc85c412 为临时管理身份认证令牌，opnfv 为之前设定的 keystone 密码。

2.4、将配置信息写入到身份服务数据库 keystone

```
su -s /bin/sh -c "keystone-manage db_sync" keystone
```

注：若失败，尝试去掉 su -s。

2.5、初始化 Fernet keys

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

3、创建服务实体 (Service Entity) 和 API 路径 (API Endpoints)

3.1、向 openstack 命令传递认证令牌值和身份服务 URL

```
export OS_TOKEN=e8918f6cfd43dc85c412
```

```
export OS_URL=http://192.168.34.88:35357/v3
```

```
export OS_IDENTITY_API_VERSION=3
```

注：export OS_TOKEN=中 e8918f6cfd43dc85c412 为临时管理身份认证令牌。

3.2、创建服务实体

```
openstack service create --name keystone --description "OpenStack Identity" identity
```

```
root@lyn:~# openstack service create --name keystone --description "OpenStack Identity" identity
+-----+-----+
| Field | Value |
+-----+-----+
| description | OpenStack Identity |
| enabled | True |
| id | 2de45937ab0345838c1ec5bd35b7ae1b |
| name | keystone |
| type | identity |
+-----+-----+
```

问题:

The program 'openstack' can be found in the following packages:

* python-openstackclient

* python3-openstackclient

Try: apt install <selected package>

解决方案: apt-get install python-openstackclient, apt-get install python3-openstackclient

问题:

The request you have made requires authentication. (HTTP 401) (Request-ID: req-c4c19fcb-5813-42f1-9479-2c6f35e732da)

解决方案: 初始化 MySQL 和 Keystone, service mysql restart, service keystone restart

注: 身份服务管理着一个 OpenStack 的服务目录, 通过服务目录确定其他服务是否可用。

3.3、创建 API 路径

openstack endpoint create --region RegionOne identity public <http://192.168.34.88:5000/v3>

openstack endpoint create --region RegionOne identity internal <http://192.168.34.88:5000/v3>

openstack endpoint create --region RegionOne identity admin <http://192.168.34.88:35357/v3>

```
root@lyn:~# openstack endpoint create --region RegionOne identity public http://192.168.34.88:5000/v3
+-----+
| Field | Value |
+-----+
| enabled | True |
| id | d85e72c4986b479aa2af898849201834 |
| interface | public |
| region | RegionOne |
| region_id | RegionOne |
| service_id | 2de45937ab0345838c1ec5bd35b7ae1b |
| service_name | keystone |
| service_type | identity |
| url | http://192.168.34.88:5000/v3 |
+-----+
root@lyn:~# openstack endpoint create --region RegionOne identity internal http://192.168.34.88:5000/v3
+-----+
| Field | Value |
+-----+
| enabled | True |
| id | 88e2319eda3549b6a066c9fb587130e9 |
| interface | internal |
| region | RegionOne |
| region_id | RegionOne |
| service_id | 2de45937ab0345838c1ec5bd35b7ae1b |
| service_name | keystone |
| service_type | identity |
| url | http://192.168.34.88:5000/v3 |
+-----+
root@lyn:~# openstack endpoint create --region RegionOne identity admin http://192.168.34.88:35357/v3
+-----+
| Field | Value |
+-----+
| enabled | True |
| id | 065e262c47944ffe82bf91e51e2aee4f |
| interface | admin |
| region | RegionOne |
| region_id | RegionOne |
| service_id | 2de45937ab0345838c1ec5bd35b7ae1b |
| service_name | keystone |
| service_type | identity |
| url | http://192.168.34.88:35357/v3 |
+-----+
```

注: OpenStack 每个服务可使用三种 API 路径变体: admin, internal 和 public。默认情况, admin 类型的 API 路径可修改用户 (user) 和租户 (tenant), 而 internal 和 public 类型的 API 路径不允许该操作。

4、创建域 (Domain)、计划 (Project)、用户 (User)、角色 (Role)

4.1、创建默认域

openstack domain create --description "Default Domain" default


```
root@lyn:~# openstack domain create --description "Default Domain" default
```

Field	Value
description	Default Domain
enabled	True
id	db32fe3468364424873d344aea70654b
name	default

- 4.2、创建管理计划、管理用户、管理角色：admin project、admin user、admin role
- ```
openstack project create --domain default --description "Admin Project" admin
openstack user create --domain default --password-prompt admin
openstack role create admin
```

```
root@lyn:~# openstack project create --domain default --description "Admin Project" admin
```

| Field       | Value                            |
|-------------|----------------------------------|
| description | Admin Project                    |
| domain_id   | db32fe3468364424873d344aea70654b |
| enabled     | True                             |
| id          | 6c8939f0da0a4058832bd78b1271acca |
| is_domain   | False                            |
| name        | admin                            |
| parent_id   | db32fe3468364424873d344aea70654b |

```
root@lyn:~# openstack user create --domain default --password-prompt admin
User Password:
Repeat User Password:
The passwords entered were not the same
User Password:
Repeat User Password:
```

| Field     | Value                            |
|-----------|----------------------------------|
| domain_id | db32fe3468364424873d344aea70654b |
| enabled   | True                             |
| id        | 3e9b545f47884d1785e86d25e7239739 |
| name      | admin                            |

```
root@lyn:~# openstack role create admin
```

| Field     | Value                            |
|-----------|----------------------------------|
| domain_id | None                             |
| id        | f35788a9570d403e83f19603a399120b |
| name      | admin                            |

注：添加 admin 用户时需设置密码，本人设置密码为 admin。

- 4.3、将 admin 角色授予 admin 计划和 admin 用户
- ```
openstack role add --project admin --user admin admin
```

- 4.4、创建服务计划

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

```

root@lyn:~# openstack project create --domain default --description "Service Project" service
+-----+-----+
| Field      | Value                                     |
+-----+-----+
| description | Service Project                         |
| domain_id   | db32fe3468364424873d344aea70654b      |
| enabled     | True                                    |
| id          | 71e3d41f94b24391a51f4d2211d62957      |
| is_domain   | False                                  |
| name        | service                                 |
| parent_id   | db32fe3468364424873d344aea70654b      |
+-----+-----+

```

4.5、创建示例计划、示例用户、普通用户角色

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

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

```
openstack role create user
```

```

root@lyn:~# openstack project create --domain default --description "Demo Project" demo
+-----+-----+
| Field      | Value                                     |
+-----+-----+
| description | Demo Project                           |
| domain_id   | db32fe3468364424873d344aea70654b      |
| enabled     | True                                    |
| id          | 02cc1e3c889b4fb187ab45cc1cdaf6b2      |
| is_domain   | False                                  |
| name        | demo                                    |
| parent_id   | db32fe3468364424873d344aea70654b      |
+-----+-----+

```

```

root@lyn:~# openstack user create --domain default --password-prompt demo
User Password:
Repeat User Password:

```

```

+-----+-----+
| Field      | Value                                     |
+-----+-----+
| domain_id   | db32fe3468364424873d344aea70654b      |
| enabled     | True                                    |
| id          | c5877ef0d2f443e6beee36985f3f2b43      |
| name        | demo                                    |
+-----+-----+

```

```
root@lyn:~# openstack role create user
```

```

+-----+-----+
| Field      | Value                                     |
+-----+-----+
| domain_id   | None                                    |
| id          | ddf43debba94e49a875e1b189bfc5ae        |
| name        | user                                    |
+-----+-----+

```

注：添加 demo 用户时需设置密码，本人设置密码为 demo，常规任务（非管理任务）应该使用非特权计划和用户。

4.6、将普通用户角色授予示例计划和示例用户

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

5、验证 Keystone 组件配置

5.1、禁用临时身份认证令牌机制

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

```
[pipeline:public_api]、[pipeline:admin_api]、[pipeline:api_v3]中处移除 admin_token_auth
```

5.2、取消环境变量 OS_TOKEN 和 OS_URL

```
unset OS_TOKEN OS_URL
```


5.4、为 admin 用户申请一个身份认证令牌

```
openstack --os-auth-url http://192.168.34.88:35357/v3 --os-project-domain-name default --os-user-domain-name default --os-project-name admin --os-username admin token issue
```

问题：The request you have made requires authentication. (HTTP 401) (Request-ID: req-f18f0d9c-49a3-4ca8-927b-f41dadac71c4)

解决方案：删除已创建的域、管理计划、管理用户、管理角色、服务计划、示例计划、示例用户、普通用户角色，重新创建。创建后解决此问题。后续分析发现，第一次创建出现 401 错误是由于未执行 4.3 步骤。

注：输入密码为之前创建的 admin 用户密码。

5.5、为 demo 用户申请一个身份认证令牌

```
openstack --os-auth-url http://192.168.34.88:5000/v3 --os-project-domain-name default --os-user-domain-name default --os-project-name demo --os-username demo token issue
```

5.6、为管理用户 admin 创建 OpenStack 客户端环境脚本

```
vi ~/.openstack/.admin-openrc
# Add environment variables for admin
export OS_PROJECT_DOMAIN_NAME=default
export OS_USER_DOMAIN_NAME=default
export OS_PROJECT_NAME=admin
export OS_USERNAME=admin
export OS_PASSWORD=admin
export OS_AUTH_URL=http://192.168.34.88:35357/v3
export OS_AUTH_TYPE=password
export OS_IDENTITY_API_VERSION=3
export OS_IMAGE_API_VERSION=2
```

注：export OS_PASSWORD=admin 这一行中，admin 为 admin 用户密码

5.7、为示例用户 demo 创建 OpenStack 客户端环境脚本

```
vi ~/.openstack/.demo-openrc
# Add environment variables for demo
export OS_PROJECT_DOMAIN_NAME=default
export OS_USER_DOMAIN_NAME=default
export OS_PROJECT_NAME=demo
export OS_USERNAME=demo
export OS_PASSWORD=demo
export OS_AUTH_URL=http://192.168.34.88:5000/v3
export OS_AUTH_TYPE=password
export OS_IDENTITY_API_VERSION=3
export OS_IMAGE_API_VERSION=2
```

注：export OS_PASSWORD=demo 这一行中，“demo”为 demo 用户密码

6、身份服务配置（Identity Service - Keystone）总结

50%的 ERROR 都可以使用以下四条命令解决

初始化 MySQL

```
service mysql restart
```

初始化 keystone

```
service keystone restart
```

将配置信息写入到身份服务数据库 keystone

```
/bin/sh -c "keystone-manage db_sync" keystone
```

初始化 Fernet keys

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

绝大部分的 ERROR 都是由于误操作引起的，每一条指令在输入时请确保正确。

通过 netstat -natp 和查看 log 文件来定位错误，先找到问题再制定解决方案。

四、镜像服务配置 (Image Service - Glance)

1、在 MariaDB 中创建 glance 数据库

1.1、以 root 用户登录 MySQL

```
mysql -u root -p
```

1.2、创建 glance 数据库

```
CREATE DATABASE glance;
```

1.3、授予 glance 数据库适当权限

```
GRANT ALL PRIVILEGES ON glance.* TO 'glance'@'192.168.34.88' IDENTIFIED BY 'opnfv';
```

```
GRANT ALL PRIVILEGES ON glance.* TO 'glance'@'%' IDENTIFIED BY 'opnfv';
```

注：opnfv 为本人设定的 glance 数据库密码。

2、创建 Glance 服务实体和 API 路径

2.1、设置 OpenStack 中 admin 用户环境变量

```
source ~/.openstack/.admin-openrc
```

2.2、在 OpenStack 中创建一个 glance 用户

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

```
root@lyn:/# openstack user create --domain default --password-prompt glance
User Password:
Repeat User Password:
+-----+-----+
| Field | Value |
+-----+-----+
| domain_id | 5d243f9e41aa4ff3af6ba264a9015113 |
| enabled | True |
| id | 95e8c7740a35471aa983db68b660ffd8 |
| name | glance |
+-----+-----+
```

注：输入用户 glance 的密码 opnfv。

2.3、将 admin 角色授予 glance 用户和 service 计划

```
openstack role add --project service --user glance admin
```

2.4、创建 glance 服务实体

```
openstack service create --name glance --description "OpenStack Image" image
```

```
root@lyn:/# openstack service create --name glance --description "OpenStack Image" image
+-----+-----+
| Field | Value |
+-----+-----+
| description | OpenStack Image |
| enabled | True |
| id | 5973f32c3fee47439265d893c9e6d1bb |
| name | glance |
| type | image |
+-----+-----+
```

2.5、创建镜像服务 API 路径

```
openstack endpoint create --region RegionOne image public http://192.168.34.88:9292
```

```
openstack endpoint create --region RegionOne image internal http://192.168.34.88:9292
```

```
openstack endpoint create --region RegionOne image admin http://192.168.34.88:9292
```

```

root@lyn:/# openstack endpoint create --region RegionOne image public http://192.168.34.88:9292
+-----+-----+
| Field | Value |
+-----+-----+
| enabled | True |
| id | f44b7d09773e4850bcf2f596894ca8c6 |
| interface | public |
| region | RegionOne |
| region_id | RegionOne |
| service_id | 5973f32c3fee47439265d893c9e6d1bb |
| service_name | glance |
| service_type | image |
| url | http://192.168.34.88:9292 |
+-----+-----+
root@lyn:/# openstack endpoint create --region RegionOne image internal http://192.168.34.88:9292
+-----+-----+
| Field | Value |
+-----+-----+
| enabled | True |
| id | c951627451654d2c83222a6465295738 |
| interface | internal |
| region | RegionOne |
| region_id | RegionOne |
| service_id | 5973f32c3fee47439265d893c9e6d1bb |
| service_name | glance |
| service_type | image |
| url | http://192.168.34.88:9292 |
+-----+-----+
root@lyn:/# openstack endpoint create --region RegionOne image admin http://192.168.34.88:9292
+-----+-----+
| Field | Value |
+-----+-----+
| enabled | True |
| id | 3b313850ae38473d8fec92d8c8da8ec |
| interface | admin |
| region | RegionOne |
| region_id | RegionOne |
| service_id | 5973f32c3fee47439265d893c9e6d1bb |
| service_name | glance |
| service_type | image |
| url | http://192.168.34.88:9292 |
+-----+-----+

```

3、安装和配置 Glance 服务组件

3.1、安装 Glance

```
apt-get install glance
```

3.2、修改配置文件 glance-api.conf

```
vi /etc/glance/glance-api.conf
```

```
[database]
```

```
connection = mysql+pymysql://glance:opnfv@192.168.34.88/glance
```

```
...
```

```
[keystone_authtoken]
```

```
auth_uri = http://192.168.34.88:5000
```

```
auth_url = http://192.168.34.88:35357
```

```
memcached_servers = 192.168.34.88:11211
```

```
auth_type = password
```

```
project_domain_name = default
```

```
user_domain_name = default
```

```
project_name = service
```

```
username = glance
```

```
password = opnfv
```

```
...
[paste_deploy]
flavor = keystone
...
[glance_store]
stores = file,http
default_store = file
filesystem_store_datadir = /var/lib/glance/images/
```

注： **opnfv** 均为数据库密码，注释掉[keystone_authtoken]中其余内容，[keystone_authtoken]中后面五行需要自行添加。

3.3、修改配置文件 glance-registry.conf

```
vi /etc/glance/glance-registry.conf
[database]
connection = mysql+pymysql://glance:opnfv@192.168.34.88/glance
...
[keystone_authtoken]
auth_uri = http://192.168.34.88:5000
auth_url = http://192.168.34.88:35357
memcached_servers = 192.168.34.88:11211
auth_type = password
project_domain_name = default
user_domain_name = default
project_name = service
username = glance
password = opnfv
...
[paste_deploy]
flavor = keystone
...
```

注： **opnfv** 均为数据库密码，注释掉[keystone_authtoken]中其余内容，[keystone_authtoken]中后面五行需要自行添加。

3.4、将配置信息写入 glance 数据库

```
su -s /bin/sh -c "glance-manage db_sync" glance
```

注：若失败，尝试去掉 **su -s**。

3.5、重启 Image Service

```
service glance-registry restart
service glance-api restart
```

4、验证 Glance 服务组件配置

4.1、设置 OpenStack admin 用户环境变量

```
source ~/.openstack/.admin-openrc
```

4.2、下载 CirrOS 系统镜像

```
wget http://download.cirros-cloud.net/0.3.4/cirros-0.3.4-x86_64-disk.img
```

注：此处采用 **Cirros** 镜像测试，也可选择其他系统镜像

4.3、上传镜像

```
openstack image create "cirros" --file cirros-0.3.4-x86_64-disk.img --disk-format qcow2 --container-format bare --public
```

```
root@lyn:/# openstack image create "cirros" --file cirros-0.3.4-x86_64-disk.img
--disk-format qcow2 --container-format bare --public
```

Field	Value
checksum	eeleca47dc88f4879d8a229cc70a07c6
container_format	bare
created_at	2016-08-08T01:37:07Z
disk_format	qcow2
file	/v2/images/324e986e-27d5-48f7-a356-a53e902455ab/file
id	324e986e-27d5-48f7-a356-a53e902455ab
min_disk	0
min_ram	0
name	cirros
owner	16f63cc27d534d778399e427b42474ef
protected	False
schema	/v2/schemas/image
size	13287936
status	active
tags	
updated_at	2016-08-08T01:37:08Z
virtual_size	None
visibility	public

4.4、验证镜像属性

openstack image list

```
root@lyn:/# openstack image list
```

ID	Name	Status
324e986e-27d5-48f7-a356-a53e902455ab	cirros	active

5、镜像服务配置（Image Service - Glance）总结

在上传镜像时报错，但是始终无法解决。

错误：

HTTPInternalServerError (HTTP 500)

查看 glance-api.log 文件

raise exceptions.DiscoveryFailure('Could not determine a suitable URL '

DiscoveryFailure: Could not determine a suitable URL for the plugin

尝试解决：

方案一、重新将配置信息写入 glance 数据库，重启 Image Service。方案失败。

方案二、修改 glance-api.conf 文件，加入 notifier_strategy = noop，再执行方案一步骤。方案失败。

方案三、修改 glance-api.conf 文件：swift_store_auth_address = 192.168.34.88:5000/v2.0，默认使用 http 协议，再执行方案一步骤。方案失败。

方案四、修改 glance-api.conf 文件，在[glance_store]中加入 filesystem_store_datadirs = /var/glance/store，再执行方案一步骤。方案失败。

方案五、删除 cirros-0.3.4-x86_64-disk.img，重新安装后再进行上传。方案失败。

最后在检查 glance-api.conf 文件过程中，发现[database]下的 auth_uri 和 auth_url 的 ip 地址配置输错了一位数字。此次失误导致自己浪费了好几个小时的时间，最后还是在梁老师协助下解决。

五、计算服务配置（Compute Service - Nova）

1、在 MariaDB（MySQL）中创建数据库 nova_api 和 nova

1.1、以 root 用户登录 MySQL

mysql -u root -p

1.2、创建 nova_api 和 nova 数据库

```
CREATE DATABASE nova_api;  
CREATE DATABASE nova;
```

1.3、授予 nova_api 和 nova 数据库适当权限

```
GRANT ALL PRIVILEGES ON nova_api.* TO 'nova'@'192.168.34.88' IDENTIFIED BY 'opnfv';  
GRANT ALL PRIVILEGES ON nova_api.* TO 'nova'@'%' IDENTIFIED BY 'opnfv';  
GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'192.168.34.88' IDENTIFIED BY 'opnfv';  
GRANT ALL PRIVILEGES ON nova.* TO 'nova'@'%' IDENTIFIED BY 'opnfv';
```

注：opnfv 为本人设定的 nova 和 nova_api 数据库密码。

2、创建计算服务证书和 API 路径

2.1、设置 OpenStack admin 用户环境变量

```
source ~/.openstack/.admin-openrc
```

2.2、在 OpenStack 中创建 nova 用户

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

```
root@lyn:/# openstack user create --domain default --password-prompt nova  
User Password:  
Repeat User Password:  
+-----+-----+  
| Field   | Value                                     |  
+-----+-----+  
| domain_id | 5d243f9e41aa4ff3af6ba264a9015113 |  
| enabled   | True                                   |  
| id        | bd7c3b983362458e94d546f08e09b020 |  
| name      | nova                                  |  
+-----+-----+
```

注：输入用户 nova 的密码 opnfv。

2.3、将 admin 角色授予 nova 用户

```
openstack role add --project service --user nova admin
```

2.4、创建 nova 服务实体

```
openstack service create --name nova --description "OpenStack Compute" compute
```

```
root@lyn:/# openstack service create --name nova --description "OpenStack Compute" compute  
+-----+-----+  
| Field   | Value                                     |  
+-----+-----+  
| description | OpenStack Compute |  
| enabled     | True               |  
| id          | d3030778ea1d45b88fcf28df0dc87fc9 |  
| name        | nova                |  
| type        | compute             |  
+-----+-----+
```

2.5、创建计算服务 API 路径

```
openstack endpoint create --region RegionOne compute public http://192.168.34.88:8774/v2.1/%(tenant_id)s  
openstack endpoint create --region RegionOne compute internal http://192.168.34.88:8774/v2.1/%(tenant_id)s  
openstack endpoint create --region RegionOne compute admin http://192.168.34.88:8774/v2.1/%(tenant_id)s
```



```
root@lyn:/# openstack endpoint create --region RegionOne compute public http://192.168.34.88:8774/v2.1/%(tenant_id)s
```

Field	Value
enabled	True
id	badab8e63265417e9dcbd8bb318d0f8a
interface	public
region	RegionOne
region_id	RegionOne
service_id	d3030778ea1d45b88fcf28df0dc87fc9
service_name	nova
service_type	compute
url	http://192.168.34.88:8774/v2.1/%(tenant_id)s

```
root@lyn:/# openstack endpoint create --region RegionOne compute internal http://192.168.34.88:8774/v2.1/%(tenant_id)s
```

Field	Value
enabled	True
id	f81b564e1a504546a0d6c80a3e206017
interface	internal
region	RegionOne
region_id	RegionOne
service_id	d3030778ea1d45b88fcf28df0dc87fc9
service_name	nova
service_type	compute
url	http://192.168.34.88:8774/v2.1/%(tenant_id)s

```
root@lyn:/# openstack endpoint create --region RegionOne compute admin http://192.168.34.88:8774/v2.1/%(tenant_id)s
```

Field	Value
enabled	True
id	b552d51067cf45809a523997aa392355
interface	admin
region	RegionOne
region_id	RegionOne
service_id	d3030778ea1d45b88fcf28df0dc87fc9
service_name	nova
service_type	compute
url	http://192.168.34.88:8774/v2.1/%(tenant_id)s

3、安装配置 nova 组件

3.1、安装 nova 组件

apt-get install nova-api nova-conductor nova-consoleauth nova-novncproxy nova-scheduler

3.2、修改配置文件 nova.conf

vi /etc/nova/nova.conf

在[DEFAULT]处只启用 compute 和 metadata APIs:

[DEFAULT]

原: enabled_apis=ec2,osapi_compute,metadata

改: enabled_apis = osapi_compute,metadata

在[api_database]和[database]处配置数据库访问连接:

[api_database]

```
connection = mysql+pymysql://nova:opnfv@192.168.34.88/nova_api
```

```
[database]
```

```
connection = mysql+pymysql://nova:opnfv@192.168.34.88/nova
```

注：若没有[api_database]和[database]标记，则手动添加。

在[DEFAULT]和[oslo_messaging_rabbit]处配置 RabbitMQ 消息队里访问：

```
[DEFAULT]
```

```
rpc_backend = rabbit
```

```
[oslo_messaging_rabbit]
```

```
rabbit_host = controller
```

```
rabbit_userid = openstack
```

```
rabbit_password = opnfv
```

在[DEFAULT]和[keystone_authtoken]处配置身份服务访问

```
[DEFAULT]
```

```
auth_strategy = keystone
```

```
[keystone_authtoken]
```

```
auth_uri = http://192.168.34.88:5000
```

```
auth_url = http://192.168.34.88:35357
```

```
memcached_servers = 192.168.34.88:11211
```

```
auth_type = password
```

```
project_domain_name = default
```

```
user_domain_name = default
```

```
project_name = service
```

```
username = nova
```

```
password = opnfv
```

在[DEFAULT]处配置 my_ip 为 Controller 节点 Management Network 网口地址

```
[DEFAULT]
```

```
my_ip = 192.168.34.88
```

注：由于本文为单节点配置，所以 ip 为本机 ip。

在[DEFAULT]处启用网络服务支持

```
[DEFAULT]
```

```
use_neutron = True
```

```
firewall_driver = nova.virt.firewall.NoopFirewallDriver
```

注：默认情况下，计算服务使用主机内部防火墙驱动，因此必须禁用 OpenStack 网络服务中的防火墙驱动。

在[vnc]处，使用 Controller 节点 Management Network 网口地址配置 VNC 代理（VNC proxy）。

```
[vnc]
```

```
vncserver_listen = $my_ip
```

```
vncserver_proxyclient_address = $my_ip
```

在[glance]处配置镜像服务 API 位置

```
[glance]
```

```
api_servers = http://192.168.34.88:9292
```

在[oslo_concurrency]处配置 lock_path

```
[oslo_concurrency]
```

```
lock_path = /var/lib/nova/tmp
```

由于安装包 BUG，需要在[DEFAULT]处移除 logdir

```
[DEFAULT]
```

```
删：logdir=/var/log/nova
```

3.3、将配置信息写入计算服务数据库 nova

```
su -s /bin/sh -c "nova-manage api_db sync" nova
```

```
su -s /bin/sh -c "nova-manage db sync" nova
```

注：若失败，尝试去掉 su -s。

3.4、重启计算服务

```
service nova-api restart
service nova-consoleauth restart
service nova-scheduler restart
service nova-conductor restart
service nova-novncproxy restart
```

4、安装配置 nova-compute 组件

4.1、安装 nova-compute 组件

```
apt-get install nova-compute
```

4.2、修改配置文件 nova.conf

```
vi /etc/nova/nova.conf
```

在[DEFAULT]和[oslo_messaging_rabbit]处配置 RabbitMQ 消息队列访问

```
[DEFAULT]
rpc_backend = rabbit
```

```
[oslo_messaging_rabbit]
rabbit_host = controller
rabbit_userid = openstack
rabbit_password = opnfv
```

在[DEFAULT]和[keystone_authtoken]处配置身份服务访问

```
[DEFAULT]
auth_strategy = keystone
```

```
[keystone_authtoken]
auth_uri = http://192.168.34.88:5000
auth_url = http://192.168.34.88:35357
memcached_servers = 192.168.34.88:11211
auth_type = password
project_domain_name = default
user_domain_name = default
project_name = service
username = nova
password = opnfv
```

注：注释或删除[keystone_authtoken]处其余内容。

在[DEFAULT]处配置 my_ip 为 Compute 节点 Management Network 网口地址

```
[DEFAULT]
my_ip = 192.168.34.88
```

注：由于本文为单节点配置，所以 ip 为本机 ip。

在[DEFAULT]处启用网络服务支持

```
[DEFAULT]
use_neutron = True
firewall_driver = nova.virt.firewall.NoopFirewallDriver
```

在[vnc]处配置远程控制访问

```
[vnc]
enabled = True
vncserver_listen = 0.0.0.0
vncserver_proxyclient_address = $my_ip
novncproxy_base_url = http://192.168.34.88:6080/vnc_auto.html
```

注：VNC 服务器端监听所有地址，VNC 代理客户端只监听 Compute 节点 Management Network 网口地址，base URL 设置 Compute 节点远程控制台浏览器访问地址（若浏览无法解析 controller，则需替换为相应 IP 地址）。

在[glance]处配置镜像服务 API

```
[glance]
api_servers = http://192.168.34.88:9292
```

在[oslo_concurrency]处配置 lock_path

```
[oslo_concurrency]
lock_path = /var/lib/nova/tmp
由于安装包 BUG，需要在[DEFAULT]处移除 logdir
[DEFAULT]
删：logdir=/var/log/nova
```

4.3、检测是否支持虚拟机硬件加速

```
egrep -c '(vmx|svm)' /proc/cpuinfo
```

若返回结果大于等于 1，则支持，无需做额外配置；
若返回结果 0，则不支持硬件加速，需要做以下额外配置：

修改配置文件 nova-compute.conf，使用 QEMU 代替 KVM

```
vi /etc/nova/nova-compute.conf
[libvirt]
virt_type = qemu
```

4.4 重启计算服务

```
service nova-compute restart
```

5、验证计算服务是否安装正确

5.1、设置 OpenStack admin 用户环境变量

```
source ~/.openstack/.admin-openrc
```

5.2、打印服务组件列表，验证每个成功启动和注册的进程。

```
openstack compute service list
```

```
root@lyn:/var/log/nova# openstack compute service list
```

Id	Binary	Host	Zone	Status	State	Updated At
4	nova-conductor	lyn	internal	enabled	up	2016-08-08T08:17:52.000000
6	nova-scheduler	lyn	internal	enabled	up	2016-08-08T08:17:52.000000
7	nova-consoleauth	lyn	internal	enabled	up	2016-08-08T08:17:50.000000
8	nova-compute	lyn	nova	enabled	up	None

6、计算服务配置（Compute Service - Nova）总结

nova 配置过程中，由于官方文档两次都修改了 nova.conf 文件，而昨天同事在配置过程中，修改了 nova.conf 和 nova-compute.conf 文件，并且由于修改内容较多，所以在修改之前进行了备份。然而在/etc/nova 文件夹下的所有文件所属组和所属用户均为 nova，cp 之后所属组和所属用户就变为了 root。在后来删除原文件，使用备份文件时，没有将所属组和所属用户改过来，导致 nova-api 等文件无法正常重启。

六、网络服务配置（Networking Service - Neutron）

1、在 MariaDB（MySQL）中创建 neutron 数据库

1.1、以 root 用户登录 MySQL

```
mysql -u root -p
```

1.2、创建数据库 neutron

```
CREATE DATABASE neutron;
```

1.3、授予数据库 neutron 适当的权限

```
GRANT ALL PRIVILEGES ON neutron.* TO 'neutron'@'192.168.34.88' IDENTIFIED BY 'opnfv';
GRANT ALL PRIVILEGES ON neutron.* TO 'neutron'@'%' IDENTIFIED BY 'opnfv';
```

注：opnfv 为本人设定的 neutron 数据库密码。

2、创建网络服务证书和 API 路径

2.1、设置 OpenStack admin 用户环境变量

```
source ~/.openstack/.admin-openrc
```

2.2、在 OpenStack 中创建 neutron 用户

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

```

root@lyn:/# openstack user create --domain default --password-prompt neutron
User Password:
Repeat User Password:
+-----+-----+
| Field | Value |
+-----+-----+
| domain_id | 5d243f9e41aa4ff3af6ba264a9015113 |
| enabled | True |
| id | 929d3555f7f54ed8ba4658ebac2c3943 |
| name | neutron |
+-----+-----+

```

注：输入用户 **neutron** 的密码 **opnfv**。

2.3、将 admin 角色授予 neutron 用户

```
openstack role add --project service --user neutron admin
```

2.4、创建 neutron 服务实体

```
openstack service create --name neutron --description "OpenStack Networking" network
```

```

root@lyn:/# openstack service create --name neutron --description "OpenStack Net
working" network
+-----+-----+
| Field | Value |
+-----+-----+
| description | OpenStack Networking |
| enabled | True |
| id | e78128e1d16b4f599ea6fe88ed20783a |
| name | neutron |
| type | network |
+-----+-----+

```

2.5、创建网络服务 API 路径

```
openstack endpoint create --region RegionOne network public http://192.168.34.88:9696
```

```
openstack endpoint create --region RegionOne network internal http://192.168.34.88:9696
```

```
openstack endpoint create --region RegionOne network admin http://192.168.34.88:9696
```

```

root@lyn:/# openstack endpoint create --region RegionOne network public http://1
92.168.34.88:9696
96
+-----+-----+
| Field | Value |
+-----+-----+
| enabled | True |
| id | bbbe732bc4d34cc5ad465a83111b0499 |
| interface | public |
| region | RegionOne |
| region_id | RegionOne |
| service_id | e78128e1d16b4f599ea6fe88ed20783a |
| service_name | neutron |
| service_type | network |
| url | http://192.168.34.88:9696 |
+-----+-----+

```

```

root@lyn:/# openstack endpoint create --region RegionOne network internal http://192.168.34.88:9696
+-----+-----+
| Field | Value |
+-----+-----+
| enabled | True |
| id | a3edcde5ede1430fa76240195cccc548 |
| interface | internal |
| region | RegionOne |
| region_id | RegionOne |
| service_id | e78128e1d16b4f599ea6fe88ed20783a |
| service_name | neutron |
| service_type | network |
| url | http://192.168.34.88:9696 |
+-----+-----+
root@lyn:/# openstack endpoint create --region RegionOne network admin http://192.168.34.88:9696
+-----+-----+
| Field | Value |
+-----+-----+
| enabled | True |
| id | 04b4be36489a4d6d851a2d3dca17b2df |
| interface | admin |
| region | RegionOne |
| region_id | RegionOne |
| service_id | e78128e1d16b4f599ea6fe88ed20783a |
| service_name | neutron |
| service_type | network |
| url | http://192.168.34.88:9696 |
+-----+-----+

```

3、安装配置 neutron-server 服务组件

3.1、安装 neutron-server 组件

apt-get install neutron-server neutron-plugin-ml2

3.2、修改配置文件 neutron.conf

vi /etc/neutron/neutron.conf

在[database]处配置网络服务数据库 neutron 访问连接

[database]

connection = sqlite:///var/lib/neutron/neutron.sqlite

connection = mysql+pymysql://neutron:opnfv@192.168.34.88/neutron

注：先注释掉默认的 connection，opnfv 为本人设定的密码。

在[DEFAULT]处启用 Modular Layer 2 (ML2) plug-in、router service 和 overlapping IP addresses

[DEFAULT]

core_plugin = ml2

service_plugins = router

allow_overlapping_ips = True

在[DEFAULT]和[oslo_messaging_rabbit]处配置 RabbitMQ 消息队列访问

[DEFAULT]

rpc_backend = rabbit

[oslo_messaging_rabbit]

rabbit_host = 192.168.34.88

rabbit_userid = openstack

rabbit_password = opnfv

在[DEFAULT]和[keystone_authtoken]处配置身份服务访问

[DEFAULT]

auth_strategy = keystone

[keystone_authtoken]


```
auth_uri = http://192.168.34.88:5000
auth_url = http://192.168.34.88:35357
memcached_servers = 192.168.34.88:11211
auth_type = password
project_domain_name = default
user_domain_name = default
project_name = service
username = neutron
password = opnfv
```

注：注释或删除[keystone_authtoken]处其他配置信息
在[DEFAULT]和[nova]处配置网络来通知计算节点网络拓扑变化。

```
[DEFAULT]
notify_nova_on_port_status_changes = True
notify_nova_on_port_data_changes = True
[nova]
auth_url = http://192.168.34.88:35357
auth_type = password
project_domain_name = default
user_domain_name = default
region_name = RegionOne
project_name = service
username = nova
password = opnfv
```

3.3、修改配置文件 ml2_conf.ini

vi /etc/neutron/plugins/ml2/ml2_conf.ini

在[ml2]处启用 flat、VLAN、VXLAN 网络

```
[ml2]
type_drivers = flat,vlan,vxlan
```

注：在配置 ML2 插件后，移除 type_drivers 设置项的部分值，将导致数据库不一致。

在[ml2]处启用 VXLAN self-service 网络

```
[ml2]
tenant_network_types = vxlan
```

在[ml2]处启用 Linux bridge 和 layer-2 population 机制

```
[ml2]
mechanism_drivers = linuxbridge,l2population
```

注：Linux Bridge Agent 只支持 VXLAN overlay networks

在[ml2]处启用端口安全扩展驱动

```
[ml2]
extension_drivers = port_security
```

在[ml2_type_flat]处配置运营商虚拟网络为 flat network

```
[ml2_type_flat]
flat_networks = provider
```

在[ml2_type_vxlan]处配置自服务网络中的 VXLAN 网络标识符范围

```
[ml2_type_vxlan]
vni_ranges = 1:1000
```

在[securitygroup]处启用 ipset 来增强安全组规则的效率

```
[securitygroup]
enable_ipset = True
```

注：ML2 plug-in 使用 Linux 网桥机制为 OpenStack 实例建立 layer-2 虚拟网络设施（桥接和交换）。

3.4、将配置信息写入 neutron 数据库

```
su -s /bin/sh -c "neutron-db-manage --config-file /etc/neutron/neutron.conf --config-file /etc/neutron/plugins/ml2/ml2_conf.ini upgrade head" neutron
```

3.5、为计算服务配置网络访问服务

vi /etc/nova/nova.conf

```
[neutron]
url = http://192.168.34.88:9696
auth_url = http://192.168.34.88:35357
auth_type = password
project_domain_name = default
user_domain_name = default
region_name = RegionOne
project_name = service
username = neutron
password = opnfv
service_metadata_proxy = True
metadata_proxy_shared_secret = opnfv
```

注：在[neutron]处配置访问参数，启用元数据服务代理，配置元数据代理共享密钥

3.6、重启计算服务 API 和网络服务

```
service nova-api restart
service neutron-server restart
```

4、安装配置 Neutron 组件

4.1、安装网络服务组件

```
apt-get install neutron-linuxbridge-agent neutron-l3-agent neutron-dhcp-agent neutron-metadata-agent
apt-get install neutron-linuxbridge-agent
```

4.2、配置公共服务组件

```
vi /etc/neutron/neutron.conf
```

在[database]处注释掉其他所有设置项，因为 Network 节点不需要直接访问数据库。

在[DEFAULT]和[oslo_messaging_rabbit]处配置 RabbitMQ 消息队列访问

```
[DEFAULT]
```

```
rpc_backend = rabbit
```

```
[oslo_messaging_rabbit]
```

```
rabbit_host = 192.168.34.88
```

```
rabbit_userid = openstack
```

```
rabbit_password = opnfv
```

在[DEFAULT]和[keystone_authtoken]处配置身份服务访问

```
[DEFAULT]
```

```
auth_strategy = keystone
```

```
[keystone_authtoken]
```

```
auth_uri = http://192.168.34.88:5000
```

```
auth_url = http://192.168.34.88:35357
```

```
memcached_servers = 192.168.34.88:11211
```

```
auth_type = password
```

```
project_domain_name = default
```

```
user_domain_name = default
```

```
project_name = service
```

```
username = neutron
```

```
password = opnfv
```

注：注释或删除[keystone_authtoken]处其他配置信息

5、配置网络服务代理

5.1、配置 Linux 网桥代理

```
vi /etc/neutron/plugins/ml2/linuxbridge_agent.ini
```

在[linux_bridge]处将虚拟网络映射到物理网络

```
[linux_bridge]
```

```
physical_interface_mappings = provider:ens160
```

注：ens160 为 Network 节点 External Network 网络接口名称，可通过 ifconfig 查看。

在[vxlan]处启用 VXLAN overlay networks，配置管理 Overlay 网络的物理网络 IP 地址，启用 layer-2 population

```
[vxlan]
```

```
enable_vxlan = True
local_ip = 192.168.34.88
l2_population = True
```

注：local_ip 为 Network 节点 Management Network 网络接口 IP 地址，由于此为单节点配置，所以为本机 ip。
在[securitygroup]处启用安全组，并配置 Linux bridge iptables firewall driver

```
[securitygroup]
enable_security_group = True
firewall_driver = neutron.agent.linux.iptables_firewall.IptablesFirewallDriver
```

注：Linux bridge agent 为实例建立了二层虚拟网络设施，而且可以管理安全组。

5.2、配置三层网络代理

```
vi /etc/neutron/l3_agent.ini
```

在[DEFAULT]处配置 Linux 网桥接口驱动（Linux Bridge Interface Driver）和外网网桥。

```
[DEFAULT]
interface_driver = neutron.agent.linux.interface.BridgeInterfaceDriver
external_network_bridge =
```

注：external_network_bridge 值故意空缺，这样可使多个外部网络共用一个代理。L3(Layer-3) Agent 位自服务网络提供了路由和 NAT 服务。

5.3、配置 DHCP 代理

```
vi /etc/neutron/dhcp_agent.ini
```

在[DEFAULT]处配置 Linux bridge interface driver 和 Dnsmasq DHCP driver，启用独立的 metadata 使运营商网络实例可以访问虚拟网络元信息。

```
[DEFAULT]
interface_driver = neutron.agent.linux.interface.BridgeInterfaceDriver
dhcp_driver = neutron.agent.linux.dhcp.Dnsmasq
enable_isolated_metadata = True
```

5.4、配置元数据代理

```
vi /etc/neutron/metadata_agent.ini
```

在[DEFAULT]处配置元数据主机和共享密钥。

```
[DEFAULT]
nova_metadata_ip = 192.168.34.88
metadata_proxy_shared_secret = opnfv
```

注：元数据代理提供一些诸如证书之类的配置信息。

5.6、重启网络服务代理

```
service neutron-linuxbridge-agent restart
service neutron-dhcp-agent restart
service neutron-metadata-agent restart
service neutron-l3-agent restart
```

6、验证网络服务是否安装配置正确

6.1、在 Controller 节点上设置 OpenStack 管理用户 admin 的环境变量

```
source ~/.openstack/.admin-openrc
```

6.2、列出已加载的扩展包，验证成功启动的 neutron-server 进程

```
neutron ext-list
neutron agent-list
```

```
root@lyn:/etc/neutron# neutron ext-list
+-----+-----+
| alias | name |
+-----+-----+
| default-subnetpools | Default Subnetpools |
| availability_zone | Availability Zone |
| network_availability_zone | Network Availability Zone |
| auto-allocated-topology | Auto Allocated Topology Services |
| binding | Port Binding |
| agent | agent |
| subnet_allocation | Subnet Allocation |
| dhcp_agent_scheduler | DHCP Agent Scheduler |
| tag | Tag support |
| external-net | Neutron external network |
| net-mtu | Network MTU |
| network-ip-availability | Network IP Availability |
| quotas | Quota management support |
| provider | Provider Network |
| multi-provider | Multi Provider Network |
| address-scope | Address scope |
| timestamp_core | Time Stamp Fields addition for core resources |
| extra_dhcp_opt | Neutron Extra DHCP opts |
| security-group | security-group |
| rbac-policies | RBAC Policies |
| standard-attr-description | standard-attr-description |
| port-security | Port Security |
| allowed-address-pairs | Allowed Address Pairs |
+-----+-----+

root@lyn:/etc/neutron# neutron agent-list
+-----+-----+-----+-----+-----+-----+-----+
| id | agent_type | host | availability_zone | alive | admin_state_up | binary |
+-----+-----+-----+-----+-----+-----+-----+
| 851cc5a3-3569-4dcb-a461-31e787e05b2f | DHCP agent | lyn | nova | :- ) | True | neutron-dhcp-agent |
| 89102a79-2c79-4de3-a61c-ac2271a7fa1d | Linux bridge agent | lyn | | :- ) | True | neutron-linuxbridge-agent |
| f336da84-5ff9-4a1b-a2a3-f6f955e43425 | Metadata agent | lyn | | :- ) | True | neutron-metadata-agent |
+-----+-----+-----+-----+-----+-----+-----+
```

7、网络服务配置（Networking Service - Neutron）总结

终于把最重要的四个组件都配置完成了，已经可以在 OpenStack 上起虚拟机了。基本上熟悉了 openstack 组件配置的原理以及流程。在 Neutron 配置过程中，每次重启服务都通过 status 来检查服务启动是否正常，如果重启失败，立刻定位错误。若不检查服务启动情况，等到最后验证的时候发现出错，再定位错误的工程量就会大很多。

七、创建虚拟网络

7.1、Provider Network

7.1.1、设置 OpenStack 中 admin 用户环境变量

```
source ~/.openstack/.admin-openrc
```

7.1.2、创建网络 provider

```
neutron net-create --shared --provider:physical_network provider
--provider:network_type flat provider
```

注：--shared 设置允许所有项目访问虚拟网络，--provider:physical_network provider 和 --provider:network_type flat 将扁平虚拟网络通过 Network 节点的 eth0 口连接到扁平物理网络。

```

root@lyn:/# neutron net-create --shared --provider:physical_network provider \
> --provider:network_type flat provider
Created a new network:
+-----+-----+
| Field | Value |
+-----+-----+
| admin_state_up | True |
| availability_zone_hints | |
| availability_zones | |
| created_at | 2016-08-12T11:30:45 |
| description | |
| id | dbe758a8-2b49-4cb3-8d1c-3b4bfff7563a |
| ipv4_address_scope | |
| ipv6_address_scope | |
| mtu | 1500 |
| name | provider |
| port_security_enabled | True |
| provider:network_type | flat |
| provider:physical_network | provider |
| provider:segmentation_id | |
| router:external | False |
| shared | True |
| status | ACTIVE |
| subnets | |
| tags | |
| tenant_id | 16f63cc27d534d778399e427b42474ef |
| updated_at | 2016-08-12T11:30:45 |
+-----+-----+

```

7.1.3、创建子网

```

neutron subnet-create --name provider --allocation-pool start=192.168.34.89,end=192.168.34.98 --dns-nameserver
8.8.8.8 --gateway 192.168.34.1 provider 192.168.34.0/24

```

注：其中 **start=192.168.34.89,end=192.168.34.98** 是将分配给实例的子网的起始 IP 和结束 IP 地址（这个 IP 地址范围不能包括任何已存在的活动 IP）；**dns-nameserver 8.8.8.8** 是实际 DNS 服务器 IP（可查看 DNS 服务器配置文件 `/etc/resolv.conf`）；**gateway 192.168.34.1** 是外部网络网关 IP。

```

root@lyn:/# neutron subnet-create --name provider --allocation-pool start=192.168.34.89,end=192.168.34.98 --dns-nameserver 8.8.8.8 --gateway 192.168.34.1 provider 192.168.34.0/24
Created a new subnet:
+-----+-----+
| Field | Value |
+-----+-----+
| allocation_pools | {"start": "192.168.34.89", "end": "192.168.34.98"} |
| cidr | 192.168.34.0/24 |
| created_at | 2016-08-12T06:52:34 |
| description | |
| dns_nameservers | 8.8.8.8 |
| enable_dhcp | True |
| gateway_ip | 192.168.34.1 |
| host_routes | |
| id | 4e32780c-8ba5-4d51-85f4-0d04408c7f00 |
| ip_version | 4 |
| ipv6_address_mode | |
| ipv6_ra_mode | |
| name | provider |
| network_id | dbe758a8-2b49-4cb3-8d1c-3b4bfff7563a |
| subnetpool_id | |
| tenant_id | 16f63cc27d534d778399e427b42474ef |
| updated_at | 2016-08-12T06:52:34 |
+-----+-----+

```

7.2、Self-Service Network

7.2.1、设置 OpenStack 中 demo 用户环境变量

```

source ~/.openstack/.demo-openrc

```

7.2.2、创建网络

```
neutron net-create selfservice
```

```
root@lyn:~# neutron net-create selfservice
Created a new network:
+-----+-----+
| Field | Value |
+-----+-----+
| admin_state_up | True |
| availability_zone_hints | |
| availability_zones | |
| created_at | 2016-08-12T07:05:07 |
| description | |
| id | bd665a8b-c04c-4a25-967c-fd3a06c5db1e |
| ipv4_address_scope | |
| ipv6_address_scope | |
| mtu | 1450 |
| name | selfservice |
| port_security_enabled | True |
| router:external | False |
| shared | False |
| status | ACTIVE |
| subnets | |
| tags | |
| tenant_id | 20ac0d3691684ae3a2e435016d85bd14 |
| updated_at | 2016-08-12T07:05:07 |
+-----+-----+
```

7.2.3、创建子网

```
neutron subnet-create --name selfservice --dns-nameserver 8.8.8.8 --gateway 172.168.34.1 selfservice 172.168.34.0/24
```

```
root@lyn:~# neutron subnet-create --name selfservice --dns-nameserver 8.8.8.8 --gateway 172.168.34.1 selfservice 172.168.34.0/24
Created a new subnet:
+-----+-----+
| Field | Value |
+-----+-----+
| allocation_pools | {"start": "172.168.34.2", "end": "172.168.34.254"} |
| cidr | 172.168.34.0/24 |
| created_at | 2016-08-15T02:53:38 |
| description | |
| dns_nameservers | 8.8.8.8 |
| enable_dhcp | True |
| gateway_ip | 172.168.34.1 |
| host_routes | |
| id | 5a9dc544-8ab7-408b-b973-daac4a4d2681 |
| ip_version | 4 |
| ipv6_address_mode | |
| ipv6_ra_mode | |
| name | selfservice |
| network_id | bd665a8b-c04c-4a25-967c-fd3a06c5db1e |
| subnetpool_id | |
| tenant_id | 20ac0d3691684ae3a2e435016d85bd14 |
| updated_at | 2016-08-15T02:53:38 |
+-----+-----+
```

7.2.4、创建配置虚拟路由器

```
source ~/.openstack/.admin-openrc
neutron net-update provider --router:external
source ~/.openstack/.demo-openrc
neutron router-create router
neutron router-interface-add router selfservice
neutron router-gateway-set router provider
```

注: Self-Service 网络通过虚拟路由器（如实现双向 NAT）连接到 Provider 网络。每个虚拟路由器包含至少一个连接到 Self-Service 网络的接口和一个位于 Provider 网络的网关。Provider 网络必须包含 router:external 设置项，以便 Self-Service 网络虚拟路由器可连接到外部网络，可以网络创建时或创建后添加（必须具有 admin 或其他特权用

户权限)。

```
root@lyn:~# source ~/.openstack/.admin-openrc
root@lyn:~# neutron net-update provider --router:external
Updated network: provider
root@lyn:~# source ~/.openstack/.demo-openrc
root@lyn:~# neutron router-create router
Created a new router:
+-----+-----+
| Field | Value |
+-----+-----+
| admin_state_up | True |
| availability_zone_hints | |
| availability_zones | |
| description | |
| external_gateway_info | |
| id | 017a652c-3bef-4247-a2e5-04488bb264ee |
| name | router |
| routes | |
| status | ACTIVE |
| tenant_id | 20ac0d3691684ae3a2e435016d85bd14 |
+-----+-----+
root@lyn:~# neutron router-interface-add router selfservice
Added interface 6943dd2c-d04e-4a49-b036-86fdc763c282 to router router.
root@lyn:~# neutron router-gateway-set router provider
Set gateway for router router
```

7.2.5、验证配置是否正确

在 Network 节点上执行 ip netns，列出网络命名空间

ip netns

设置 admin 环境变量

source ~/.openstack/.admin-openrc

列出 router 路由器网络接口，查看连接 Provider 网络的网关 IP

neutron router-port-list router

```
root@lyn:~# ip netns
qrouter-d4f4092b-8cea-419f-97b3-8dbbb14dce73 (id: 2)
qdhcp-bd665a8b-c04c-4a25-967c-fd3a06c5db1e (id: 1)
qdhcp-dbe758a8-2b49-4cb3-8d1c-3b4bfff7563a (id: 0)
root@lyn:~# neutron router-port-list router
+-----+-----+-----+-----+
| id | name | mac_address | fixed_ips |
+-----+-----+-----+-----+
| 6e646706-21f3-46f6-850d-270ff2323c0a | | fa:16:3e:74:7d:a4 | {"subnet_id": "1c1b34c5-2219-449b-bffe-cf53bd1f688e", "ip_address": "192.168.34.1"} |
+-----+-----+-----+-----+
```

7.3、创建虚拟机模板

7.3.1、为 CirrOS 镜像创建用于测试的虚拟机类型模板 m1.nano

source ~/.openstack/.admin-openrc

openstack flavor create --id 0 --vcpus 1 --ram 64 --disk 1 m1.nano

7.4、生成密钥对

7.4.1、设置普通用户 demo 的认证环境变量

source ~/.openstack/.demo-openrc

7.4.2、生成一个密钥对

ssh-keygen -q -N ""

openstack keypair create --public-key ~/.ssh/id_rsa.pub demokey

注：上面可以跳过 ssh-keygen 命令，直接使用已有的公钥。

```

root@lyn:~/ssh# ssh-keygen -q -N ""
Enter file in which to save the key (/root/.ssh/id_rsa): /root/.ssh/id_rsa
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
root@lyn:~/ssh# openstack keypair create --public-key ~/.ssh/id_rsa.pub demokey
+-----+
| Field      | Value                                     |
+-----+
| fingerprint | e2:d6:cd:59:78:6f:af:0a:95:d7:d5:0c:49:d8:0a:5a |
| name        | demokey                                  |
| user_id     | a9f6696b50784a688e22ecfac2574798        |
+-----+

```

7.4.3、验证密钥对

openstack keypair list

```

root@lyn:~/ssh# openstack keypair list
+-----+
| Name      | Fingerprint                               |
+-----+
| demokey   | e2:d6:cd:59:78:6f:af:0a:95:d7:d5:0c:49:d8:0a:5a |
+-----+

```

7.5、添加安全组规则

默认安全组规则适用于所有实例，并且包含防火墙规则，该防火墙规则拒绝远程访问实例。对于 Linux 镜像，建议至少允许 ICMP (ping)和 secure shell (SSH)。

7.5.1、添加规则允许 ICMP 协议

openstack security group rule create --proto icmp default

```

root@lyn:~/ssh# openstack security group rule create --proto icmp default
+-----+
| Field      | Value                                     |
+-----+
| id         | b0e110a9-1c83-4844-a519-9b3f279112d2 |
| ip_protocol | icmp                                    |
| ip_range    | 0.0.0.0/0                               |
| parent_group_id | 8931c21c-835e-4f1b-8163-a07b693d654d |
| port_range  |                                          |
| remote_security_group |                                          |
+-----+

```

7.5.2、添加规则允许 secure shell (SSH) 访问

openstack security group rule create --proto tcp --dst-port 22 default

```

root@lyn:~/ssh# openstack security group rule create --proto tcp --dst-port 22 default
+-----+
| Field      | Value                                     |
+-----+
| id         | b7d44619-b6d2-444c-9370-6fdea575e8df |
| ip_protocol | tcp                                    |
| ip_range    | 0.0.0.0/0                               |
| parent_group_id | 8931c21c-835e-4f1b-8163-a07b693d654d |
| port_range  | 22:22                                  |
| remote_security_group |                                          |
+-----+

```

7.6、制定虚拟机模板

7.6.1、设置普通用户 demo 的认证环境变量

source ~/.openstack/demo-openrc

7.6.2、列出可用虚拟机模板

虚拟机模板指定了处理器、内存、存储等虚拟资源的分配概况。

openstack flavor list

```
root@lyn:~/ssh# openstack flavor list
```

ID	Name	RAM	Disk	Ephemeral	VCPUs	Is Public
0	m1.nano	64	1	0	1	True
1	m1.tiny	512	1	0	1	True
2	m1.small	2048	20	0	1	True
3	m1.medium	4096	40	0	2	True
4	m1.large	8192	80	0	4	True
5	m1.xlarge	16384	160	0	8	True

7.6.3、列出可用镜像

openstack image list

```
root@lyn:~/ssh# openstack image list
```

ID	Name	Status
324e986e-27d5-48f7-a356-a53e902455ab	cirros	active

7.6.4、列出可用网路

openstack network list

注：若实例启动使用 Provider 网络 provider，或者使用 Self-Service 网络 selfservice，则必须使用 ID 代替名称来使用该网络名称。

```
root@lyn:~/ssh# openstack network list
```

ID	Name	Subnets
db758a8-2b49-4cb3-8d1c-3b4bfff7563a	provider	4e32780c-8ba5-4d51-85f4-0d04408c7f00
bd665a8b-c04c-4a25-967c-fd3a06c5db1e	selfservice	1c1b34c5-2219-449b-bffe-cf53bd1f688e

7.6.5、列出可用安全组

openstack security group list

```
root@lyn:~/ssh# openstack security group list
```

ID	Name	Description	Project
5b3b1870-e5b1-4f87-9d45-71339dd6edb7	default	Default security group	20ac0d3691684ae3a2e435016d85bd14

7.7、启动实例：Provider 网络

7.7.1、在 Provider 网络中启动实例

```
openstack server create --flavor m1.tiny --image cirros \
--nic net-id=db758a8-2b49-4cb3-8d1c-3b4bfff7563a --security-group default \
--key-name mykey provider-instance
```

注：net-id=db758a8-2b49-4cb3-8d1c-3b4bfff7563a 为实际 Provider 网络 ID。

```

root@lyn:~/ssh# openstack server create --flavor m1.tiny --image cirros \
> --nic net-id=dbe758a8-2b49-4cb3-8d1c-3b4bfff7563a --security-group default \
> --key-name mykey provider-instance
+-----+
| Field | Value |
+-----+
| OS-DCF:diskConfig | MANUAL |
| OS-EXT-AZ:availability_zone | |
| OS-EXT-STS:power_state | 0 |
| OS-EXT-STS:task_state | scheduling |
| OS-EXT-STS:vm_state | building |
| OS-SRV-USG:launched_at | None |
| OS-SRV-USG:terminated_at | None |
| accessIPv4 | |
| accessIPv6 | |
| addresses | |
| adminPass | KVs9Zew9jSey |
| config_drive | |
| created | 2016-08-12T08:13:56Z |
| flavor | m1.tiny (1) |
| hostId | |
| id | da3424cb-f116-420c-815c-f1078460e0f9 |
| image | cirros (324e986e-27d5-48f7-a356-a53e902455ab) |
| key_name | mykey |
| name | provider-instance |
| os-extended-volumes:volumes_attached | [] |
| progress | 0 |
| project_id | 20ac0d3691684ae3a2e435016d85bd14 |
| properties | |
| security_groups | [{u'name': u'default'}] |
| status | BUILD |
| updated | 2016-08-12T08:13:57Z |
| user_id | 1556b06ec40c474fbd6b1b5a00d4d501 |
+-----+

```

7.7.2、检查实例状态

openstack server list

```

root@lyn:~# openstack server list
+-----+-----+-----+-----+
| ID | Name | Status | Networks |
+-----+-----+-----+-----+
| 98f7d61b-f700-44f8-bb22-1ef484ab5b68 | provider-instance | ACTIVE | provider=192.168.34.90 |
+-----+-----+-----+-----+

```

7.7.3、使用虚拟控制台访问实例

获取一个 Virtual Network Computing (VNC)会话 URL，通过 Web 浏览器访问该 URL。

openstack console url show provider-instance

```

root@lyn:~# openstack console url show provider-instance
+-----+-----+
| Field | Value |
+-----+-----+
| type | novnc |
| url | http://192.168.34.88:6080/vnc_auto.html?token=414add49-dea3-4b52-8043-f74c1174d8b6 |
+-----+-----+

```

验证 Provider 物理网络网关是否连通

ping -c 4 192.168.34.1

验证互联网是否连通

ping -c 4 openstack.org

7.7.4、远程访问实例

验证能否从 Controller 节点或其他任何位于物理网络中的节点连通实例

ping -c 4 192.168.34.90

验证能否使用 ssh 从 Controller 节点或其他任何位于物理网络中的节点访问实例

ssh cirros@192.168.34.90

注: cirros@192.168.34.90 的初始密码为 cubswin:)

7.8、启动实例：Self-Service 网络

7.8.1、在 Self-Service 网络中启动实例

```
openstack server create --flavor m1.tiny --image cirros \  
--nic net-id=bd665a8b-c04c-4a25-967c-fd3a06c5db1e --security-group default \  
--key-name mykey selfservice-instance
```

注: net-id=bd665a8b-c04c-4a25-967c-fd3a06c5db1e 为实际 Provider 网络 ID。

```
root@lyn:~# openstack server create --flavor m1.tiny --image cirros \  
> --nic net-id=bd665a8b-c04c-4a25-967c-fd3a06c5db1e --security-group default \  
> --key-name mykey selfservice-instance  
+-----+-----+  
| Field                                | Value                                |  
+-----+-----+  
| OS-DCF:diskConfig                    | MANUAL                              |  
| OS-EXT-AZ:availability_zone          |                                      |  
| OS-EXT-STS:power_state                | 0                                    |  
| OS-EXT-STS:task_state                 | scheduling                           |  
| OS-EXT-STS:vm_state                   | building                             |  
| OS-SRV-USG:launched_at                | None                                 |  
| OS-SRV-USG:terminated_at              | None                                 |  
| accessIPv4                            |                                      |  
| accessIPv6                            |                                      |  
| addresses                             |                                      |  
| adminPass                             | MT46zWlvim2z                        |  
| config_drive                          |                                      |  
| created                                | 2016-08-15T02:55:46Z                |  
| flavor                                | m1.tiny (1)                          |  
| hostId                                |                                      |  
| id                                     | 85a5b4b6-2e76-4dd4-9a5b-3bc0ec751800 |  
| image                                 | cirros (324e986e-27d5-48f7-a356-a53e902455ab) |  
| key_name                              | mykey                                |  
| name                                  | selfservice-instance                 |  
| os-extended-volumes:volumes_attached | []                                    |  
| progress                              | 0                                    |  
| project_id                            | 20ac0d3691684ae3a2e435016d85bd14    |  
| properties                            |                                      |  
| security_groups                       | [{u'name': u'default'}]              |  
| status                                | BUILD                                |  
| updated                                | 2016-08-15T02:55:46Z                |  
| user_id                               | 1556b06ec40c474fbd6b1b5a00d4d501    |  
+-----+-----+
```

7.8.2、查看实例状态

```
openstack server list
```

```
root@lyn:~# openstack server list  
+-----+-----+-----+-----+  
| ID                                     | Name                                | Status | Networks |  
+-----+-----+-----+-----+  
| 85a5b4b6-2e76-4dd4-9a5b-3bc0ec751800 | selfservice-instance              | ACTIVE | selfservice=172.168.34.3 |  
| 98f7d61b-f700-44f8-bb22-1ef484ab5b68 | provider-instance                 | ACTIVE | provider=192.168.34.90 |  
+-----+-----+-----+-----+
```

7.8.3、在外部网络上创建一个 floating ip 地址

```
neutron floatingip-create provider
```

```
root@lyn:~# neutron floatingip-create provider
Created a new floatingip:
+-----+-----+
| Field | Value |
+-----+-----+
| description | |
| dns_domain | |
| dns_name | |
| fixed_ip_address | |
| floating_ip_address | 192.168.34.95 |
| floating_network_id | db758a8-2b49-4cb3-8d1c-3b4bfff7563a |
| id | 88068e95-a92f-4554-a175-508242381ecd |
| port_id | |
| router_id | |
| status | DOWN |
| tenant_id | 20ac0d3691684ae3a2e435016d85bd14 |
+-----+-----+
```

7.8.4、将浮动 IP 地址与实例关联

nova floating-ip-associate selfservice-instance 192.168.34.95

7.8.5、使用虚拟控制台访问实例

获取一个 Virtual Network Computing (VNC)会话 URL，通过 Web 浏览器访问该 URL。

openstack console url show selfservice-instance

```
root@lyn:~# openstack console url show selfservice-instance
+-----+-----+
| Field | Value |
+-----+-----+
| type | novnc |
| url | http://192.168.34.88:6080/vnc_auto.html?token=0cab6b01-7da8-48a3-9501-d702ed1aa35e |
+-----+-----+
```

7.8.6、利用浏览器进入 http://192.168.34.88:6080/vnc_auto.html?token=0cab6b01-7da8-48a3-9501-d702ed1aa35e

验证 Self-service 网络网关是否连通

ping -c 4 172.168.34.1

验证互联网是否连通

ping -c 4 openstack.org

7.8.7、远程访问实例

验证能否从 Controller 节点或其他任何位于物理网络中的节点连通实例

ping -c 4 192.168.34.95

验证能否使用 ssh 从 Controller 节点或其他任何位于物理网络中的节点访问实例

ssh cirros@192.168.34.95

注：cirros@192.168.34.95 的初始密码为 cubswin:)