**OpenStack 搭建文档**

说明：此文档为标准安装文档，仅作参考，在部署中需要根据实际情况进行调整。

1. 搭建环境

1.1 硬件

服务器：4台，本文档采用最小化安装，至少4台物理机，每台至少三个网卡，计算节点要支持虚拟化。

存储设备：JBOD（磁盘柜）若干台

网络设备：交换机（普通网络交换机，千兆以上）1台，SAS交换机1台（可选），hba卡等

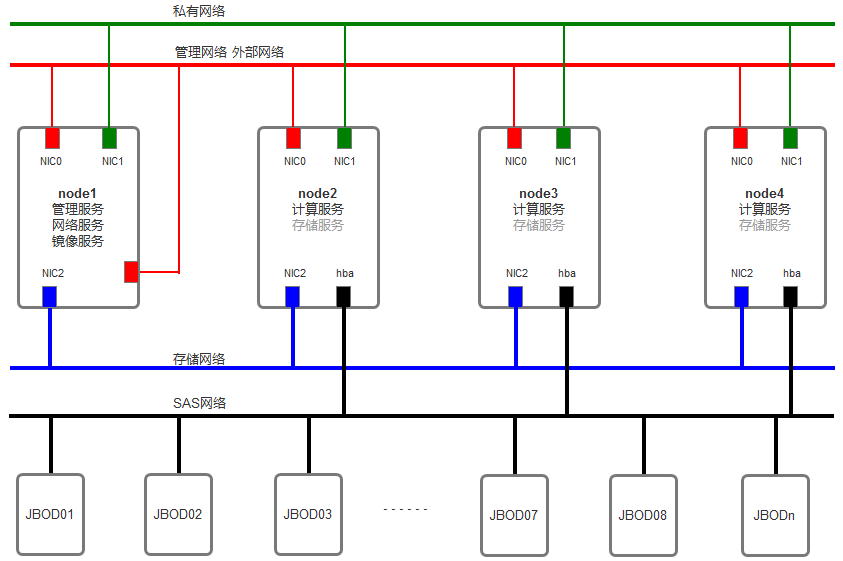
1.2 软件

系统：CentOS 7 最小化安装

基础组件：mariadb,rabbitmq

OpenStack版本：Juno版

1.3 云平台架构



1.4 节点的功能与角色

管理节点：管理云平台，主要安装管理类服务，如keystone,glance,nuetron,dashboard,nova-api等服务，以及一些基础组件，如mariadb,rabbitmq等。

计算节点：安装 nova-compute组件，提供计算虚拟化服务，及所需的资源，如内存，CPU等。

网络节点：安装 Neutron组件，提供网络虚拟化服务，通常与管理节点安装在一起。

存储节点：安装cinder-volume,ceph,nfs,zfs等组件，提供存储服务，可以安装在管理节点或计算节点上，一般由JBOD（磁盘柜）提供磁盘。

1.5 节点网络规划

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **云平台网络与服务器网卡对应表** | | | | | | |
| **交换机端口** | **节点** | **安装的软件及服务** | **网卡** | **IP** | **网络规划** | **其他** |
|  | controller (node1) | MariaDB,RabbitMQ,ntp Keystone Glance Neutron Dashboard nova-api,cinder-api等 | NIC0 | 10.0.33.11/24 | 管理网络 | 千兆 |
|  | NIC1 | 192.168.33.11/24 | 私有网络 | 千兆以上 |
|  | **NIC2** | **172.16.33.11/24** | **存储网络** | **万兆** |
|  | NIC3 | 10.0.38.0/24 | 外部网络 | 千兆 |
|  | IPMI | 10.0.99.11/24 |  |  |
|  | compute1 (node2) | nova-compute cinder-volume ceph | NIC0 | 10.0.33.31/24 | 管理网络 | 千兆 |
|  | NIC1 | 192.168.33.31/24 | 私有网络 | 千兆以上 |
|  | **NIC2** | **172.16.33.31/24** | **存储网络** | **万兆** |
|  | NIC3 |  |  |  |
|  | IPMI | 10.0.99.31/24 |  |  |
|  | compute2 (node3) | nova-compute cinder-volume ceph | NIC0 | 10.0.33.32/24 | 管理网络 | 千兆 |
|  | NIC1 | 192.168.33.32/24 | 私有网络 | 千兆以上 |
|  | **NIC2** | **172.16.33.32/24** | **存储网络** | **万兆** |
|  | NIC3 |  |  |  |
|  | IPMI | 10.0.99.32/24 |  |  |
|  | compute3 (node4) | nova-compute cinder-volume ceph | NIC0 | 10.0.33.33/24 | 管理网络 | 千兆 |
|  | NIC1 | 192.168.33.33/24 | 私有网络 | 千兆以上 |
|  | **NIC2** | **172.16.33.33/24** | **存储网络** | **万兆** |
|  | NIC3 |  |  |  |
|  | IPMI | 10.0.99.33/24 |  |  |

2. 系统初始化

**2.1 准备 Controller Node**

2.1.1 网络，主机名等相关参数配置

cat > /etc/sysconfig/network-scripts/ifcfg-eth0 <<eof

TYPE=Ethernet

BOOTPROTO=static

NAME=eth0

DEVICE=eth0

ONBOOT=yes

IPADDR=10.0.33.11

NETMASK=255.255.255.0

GATEWAY=10.0.0.1

DEFROUTE=yes

NM\_CONTROLLED=no

eof

cat > /etc/sysconfig/network-scripts/ifcfg-eth1 <<eof

TYPE=Ethernet

BOOTPROTO=static

NAME=eth1

DEVICE=eth1

ONBOOT=yes

IPADDR=192.168.33.11

NETMASK=255.255.255.0

DEFROUTE=no

eof

cat > /etc/sysconfig/network-scripts/ifcfg-eth2 <<eof

TYPE=Ethernet

BOOTPROTO=static

NAME=eth2

DEVICE=eth2

ONBOOT=yes

IPADDR=172.16.33.11

NETMASK=255.255.255.0

DEFROUTE=no

eof

cat > /etc/sysconfig/network-scripts/ifcfg-eth3 <<eof

TYPE=Ethernet

BOOTPROTO=none

NAME=eth3

DEVICE=eth3

ONBOOT=yes

eof

echo 'nameserver 10.0.0.1' > /etc/resolv.conf

echo 'nameserver 1.2.4.8' >> /etc/resolv.conf

echo 'nameserver 114.114.114.114' >> /etc/resolv.conf

systemctl disable NetworkManager

systemctl stop NetworkManager

systemctl disable firewalld

systemctl stop firewalld

systemctl disable postfix

systemctl stop postfix

sed -i 's/SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config

setenforce 0

hostnamectl --static set-hostname controller

cat > /etc/hosts <<eof

# compute3

10.0.33.33 compute3

# compute2

10.0.33.32 compute2

# compute1

10.0.33.31 compute1

# controller

10.0.33.11 controller

eof

init 6

2.1.2 安装 ntp 服务，各节点之间时间要同步。

yum install -y ntp

systemctl enable ntpd.service

systemctl start ntpd.service

将controller节点作为ntp服务器：

sed -i '/ 3.centos.pool.ntp.org /a\

restrict -4 default kod notrap nomodify\

restrict -6 default kod notrap nomodify' /etc/ntp.conf

2.1.3 安装数据库并初始化

yum install -y mariadb mariadb-server MySQL-python

sed -i '/symbolic-links=0/a\bind-address = 0.0.0.0\

default-storage-engine = innodb\

innodb\_file\_per\_table\

collation-server = utf8\_general\_ci\

init-connect = "SET NAMES utf8"\

character-set-server = utf8' /etc/my.cnf

systemctl enable mariadb.service

systemctl start mariadb.service

mysql\_secure\_installation

一路回车 ，取默认，并设置 root 用户的密码为 MYSQL\_ROOT\_PASS\_SUR 。

2.1.4 OpenStack 源，通用包，等相关安装源

yum install yum-plugin-priorities

yum install http://dl.fedoraproject.org/pub/epel/7/x86\_64/e/epel-release-7-5.noarch.rpm

yum install http://rdo.fedorapeople.org/openstack-juno/rdo-release-juno.rpm

通用包

yum install -y openstack-selinux openstack-utils

yum upgrade -y

reboot

2.1.6 安装 Rabbitmq 服务，作为消息服务，在各个组件节点之间通信

yum -y install rabbitmq-server

/usr/lib/rabbitmq/bin/rabbitmq-plugins enable rabbitmq\_management

systemctl enable rabbitmq-server.service

systemctl start rabbitmq-server.service

rabbitmqctl change\_password guest RABBIT\_GUEST\_PASS\_SUR

到这里，Controller Node 的初始化完成。

**2.2 准备 Compute Node**

2.2.1 网络，主机名等相关参数配置

cat > /etc/sysconfig/network-scripts/ifcfg-eth0 <<eof

TYPE=Ethernet

BOOTPROTO=static

NAME=eth0

DEVICE=eth0

ONBOOT=yes

IPADDR=10.0.33.31

NETMASK=255.255.255.0

GATEWAY=10.0.0.1

DEFROUTE=yes

NM\_CONTROLLED=no

eof

cat > /etc/sysconfig/network-scripts/ifcfg-eth1 <<eof

TYPE=Ethernet

BOOTPROTO=static

NAME=eth1

DEVICE=eth1

ONBOOT=yes

IPADDR=192.168.33.31

NETMASK=255.255.255.0

DEFROUTE=no

eof

cat > /etc/sysconfig/network-scripts/ifcfg-eth2 <<eof

TYPE=Ethernet

BOOTPROTO=static

NAME=eth2

DEVICE=eth2

ONBOOT=yes

IPADDR=172.16.33.31

NETMASK=255.255.255.0

DEFROUTE=no

eof

cat > /etc/sysconfig/network-scripts/ifcfg-eth3 <<eof

TYPE=Ethernet

BOOTPROTO=none

NAME=eth3

DEVICE=eth3

ONBOOT=no

eof

echo 'nameserver 10.0.0.1' > /etc/resolv.conf

echo 'nameserver 1.2.4.8' >> /etc/resolv.conf

echo 'nameserver 114.114.114.114' >> /etc/resolv.conf

systemctl disable NetworkManager

systemctl stop NetworkManager

systemctl disable firewalld

systemctl stop firewalld

systemctl disable postfix

systemctl stop postfix

sed -i 's/SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config

setenforce 0

hostnamectl --static set-hostname compute1

cat > /etc/hosts <<eof

# compute3

10.0.33.33 compute3

# compute2

10.0.33.32 compute2

# compute1

10.0.33.31 compute1

# controller

10.0.33.11 controller

eof

init 6

2.2.2 安装 ntp 服务，各节点之间时间要同步。

yum install -y ntp

systemctl enable ntpd.service

systemctl start ntpd.service

sed -i '/ 3.centos.pool.ntp.org /a\

server controller prefer iburst' /etc/ntp.conf

sed -i '/.centos.pool.ntp.org iburst/d' /etc/ntp.conf

2.2.3 安装OpenStack等相关安装源，通用包

yum install yum-plugin-priorities

yum install http://dl.fedoraproject.org/pub/epel/7/x86\_64/e/epel-release-7-5.noarch.rpm

yum install http://rdo.fedorapeople.org/openstack-juno/rdo-release-juno.rpm

yum install -y openstack-selinux openstack-utils

yum upgrade -y

reboot

到这里，Compute Node 的初始化也完成。

2.3 验证网络

在 Controller Node 上：

ping -c 4 openstack.org

ping -c 4 compute1

ping -c 4 baidu.com

ping -c 4 compute1

在 Compute Node 上：

ping -c 4 openstack.org

ping -c 4 controller

ping -c 4 baidu.com

ping -c 4 controller

到这里，系统初始化完成。如果有compute2，compute3等节点，执行同样操作，完成初始化配置。

3. 配置Identity Service

3.1 在 Controller Node 上安装 Identity Service

3.1.1 安装Identity Service

yum install -y openstack-keystone python-keystoneclient

3.1.2 配置数据库连接信息

sed -i '/#connection=mysql:/a\

connection = mysql://keystone:KEYSTONE\_DBPASS\_SUR@controller/keystone' \

/etc/keystone/keystone.conf

3.1.3 创建 keystone 数据库及用户

mysql -u root -pMYSQL\_ROOT\_PASS\_SUR

CREATE DATABASE keystone;

GRANT ALL PRIVILEGES ON keystone.\* TO 'keystone'@'localhost' \

IDENTIFIED BY 'KEYSTONE\_DBPASS\_SUR';

GRANT ALL PRIVILEGES ON keystone.\* TO 'keystone'@'%' \

IDENTIFIED BY 'KEYSTONE\_DBPASS\_SUR';

exit

3.1.4 创建表

su -s /bin/sh -c "keystone-manage db\_sync" keystone

3.1.5 定义并配置认证口令 authorization token

sed -i '/#admin\_token=ADMIN/a\

admin\_token = ADMIN\_TOKEN\_SUR' \

/etc/keystone/keystone.conf

3.1.6 创建密钥及证书

keystone-manage pki\_setup --keystone-user keystone --keystone-group keystone

chown -R keystone:keystone /var/log/keystone

chown -R keystone:keystone /etc/keystone/ssl

chmod -R o-rwx /etc/keystone/ssl

3.1.7 启动服务，并配置随机启动

systemctl enable openstack-keystone.service

systemctl start openstack-keystone.service

3.2 定义 Users（用户） Tennts（租户） 及 roles（角色）

3.2.1 配置 authorization token 等环境变量

export OS\_SERVICE\_TOKEN=ADMIN\_TOKEN\_SUR

export OS\_SERVICE\_ENDPOINT=http://controller:35357/v2.0

3.2.2 创建管理员用户

3.2.2.1 创建 admin 用户

keystone user-create --name=admin --pass=*ADMIN\_PASS\_SUR* --email=admin@example.com

+----------+----------------------------------+

| Property | Value |

+----------+----------------------------------+

| email | admin@example.com |

| enabled | True |

| id | 78005529f5fe44dc88ddc2cbda066043 |

| name | admin |

| username | admin |

+----------+----------------------------------+

格式不一样，乱了。

3.2.2.2 创建 admin 角色

keystone role-create --name=admin

+----------+----------------------------------+

| Property | Value |

+----------+----------------------------------+

| id | 60797d9022804ee188dcbf98912b2f99 |

| name | admin |

+----------+----------------------------------+

3.2.2.3 创建 admin 租户

keystone tenant-create --name=admin --description="Admin Tenant"

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| description | Admin Tenant |

| enabled | True |

| id | 219264c7a961437eb0066f357d8bc8c2 |

| name | admin |

+-------------+----------------------------------+

3.2.2.4 关联 admin 用户，admin 角色，及 admin 租户

keystone user-role-add --user=admin --tenant=admin --role=admin

3.2.2.5 关联 admin 用户，\_member\_ 角色，及 admin 租户

keystone user-role-add --user=admin --role=\_member\_ --tenant=admin

3.2.3 创建一个普通用户

3.2.3.1 创建 demo 用户

keystone user-create --name=demo --pass=*DEMO\_PASS\_SUR* --email=demo@example.com

+----------+----------------------------------+

| Property | Value |

+----------+----------------------------------+

| email | demo@example.com |

| enabled | True |

| id | 2a6b0bedaeb94ba58cb8ba225fc0cae4 |

| name | demo |

| username | demo |

+----------+----------------------------------+

3.2.3.2 创建 demo 租户

keystone tenant-create --name=demo --description="Demo Tenant"

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| description | Demo Tenant |

| enabled | True |

| id | b37e273f153d4675a8677a6e71ee5d5e |

| name | demo |

+-------------+----------------------------------+

3.2.3.3 关联 demo 用户，\_member\_ 角色，及 demo 租户

keystone user-role-add --user=demo --role=\_member\_ --tenant=demo

3.2.4 创建 service 租户

keystone tenant-create --name=service --description="Service Tenant"

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| description | Service Tenant |

| enabled | True |

| id | 681a98fa49854fa7969e56e8ebaf2647 |

| name | service |

+-------------+----------------------------------+

3.3 定义服务，及 API 终端

3.3.1 为  Identity Service 创建一个 service entry

keystone service-create --name=keystone --type=identity \

--description="OpenStack Identity"

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| description | OpenStack Identity |

| enabled | True |

| id | 3af24a2a93254358845fded540db21cb |

| name | keystone |

| type | identity |

+-------------+----------------------------------+

service ID 是一串随机字符，在下一步会用到。

3.3.2 用上一步返回的 service ID 为 Identity Service 指定一个 API 终端。终端需要为 public API, internal API, 及 admin API 提供 URLs 。

keystone endpoint-create \

--service-id=$(keystone service-list | awk '/ identity / {print $2}') \

--publicurl=http://*controller*:5000/v2.0 \

--internalurl=http://*controller*:5000/v2.0 \

--adminurl=http://*controller*:35357/v2.0

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| adminurl | http://controller:35357/v2.0 |

| id | 5c881c8a183144cb8dd0f3e6afd8221a |

| internalurl | http://controller:5000/v2.0 |

| publicurl | http://controller:5000/v2.0 |

| region | regionOne |

| service\_id | 3af24a2a93254358845fded540db21cb |

+-------------+----------------------------------+

3.4 验证 Identity Service 安装

3.4.1 取消前面设置的环境变量

unset OS\_SERVICE\_TOKEN OS\_SERVICE\_ENDPOINT

3.4.2 使用基于 用户名 的认证

keystone --os-username=admin --os-password=*ADMIN\_PASS\_SUR* \

--os-auth-url=http://controller:35357/v2.0 token-get

会输出一大片相关信息。

3.4.3 再验证

keystone --os-username=admin --os-password=*ADMIN\_PASS\_SUR* \

--os-tenant-name=admin --os-auth-url=http://controller:35357/v2.0 \

token-get

会输出一大片相关信息。

3.4.4 设置环境变量脚本

为admin设置环境变量脚本

cat > /root/admin-openrc.sh <<eof

export OS\_USERNAME=admin

export OS\_PASSWORD=ADMIN\_PASS\_SUR

export OS\_TENANT\_NAME=admin

export OS\_AUTH\_URL=http://controller:35357/v2.0

eof

让系统登录时自动读取环境变量

echo "source /root/admin-openrc.sh" >> /etc/profile

为demo设置环境变量脚本

cat > /root/demo-openrc.sh <<eof

export OS\_USERNAME=demo

export OS\_PASSWORD=DEMO\_PASS\_SUR

export OS\_TENANT\_NAME=demo

export OS\_AUTH\_URL=http://controller:35357/v2.0

eof

3.4.5 用 source 读取环境变量

source /root/admin-openrc.sh

3.4.6 验证 admin-openrc.sh 是否配置正确

keystone token-get

3.4.7 Verify that your admin account has authorization to perform administrative commands（验证 admin 是否已授权去执行管理命令）。

keystone user-list

+----------------------------------+-------+---------+-------------------+

| id | name | enabled | email |

+----------------------------------+-------+---------+-------------------+

| 78005529f5fe44dc88ddc2cbda066043 | admin | True | admin@example.com |

| 2a6b0bedaeb94ba58cb8ba225fc0cae4 | demo | True | demo@example.com |

+----------------------------------+-------+---------+-------------------+

keystone user-role-list --user admin --tenant admin

+----------------------------------+----------+----------------------------------+----------------------------------+

| id | name | user\_id | tenant\_id |

+----------------------------------+----------+----------------------------------+----------------------------------+

| 9fe2ff9ee4384b1894a90878d3e92bab | \_member\_ | 78005529f5fe44dc88ddc2cbda066043 | 219264c7a961437eb0066f357d8bc8c2 |

| 60797d9022804ee188dcbf98912b2f99 | admin | 78005529f5fe44dc88ddc2cbda066043 | 219264c7a961437eb0066f357d8bc8c2 |

+----------------------------------+----------+----------------------------------+----------------------------------+

5. 配置 Image Service

5.1 安装 Image Service

5.1.1 在 Controller Node 上安装 Image Service

yum install -y openstack-glance python-glanceclient

5.1.2 配置数据库连接信息

openstack-config --set /etc/glance/glance-api.conf database \

connection mysql://glance:*GLANCE\_DBPASS*@*controller*/glance

openstack-config --set /etc/glance/glance-registry.conf database \

connection mysql://glance:*GLANCE\_DBPASS*@*controller*/glance

5.1.3 配置 Image Service 的消息代理

openstack-config --set /etc/glance/glance-api.conf DEFAULT \

rpc\_backend qpid

openstack-config --set /etc/glance/glance-api.conf DEFAULT \

qpid\_hostname *controller*

5.1.4 创建 glance 数据库及用户

mysql -u root -proot

CREATE DATABASE glance;

GRANT ALL PRIVILEGES ON glance.\* TO 'glance'@'localhost' \

IDENTIFIED BY '*GLANCE\_DBPASS*';

GRANT ALL PRIVILEGES ON glance.\* TO 'glance'@'%' \

IDENTIFIED BY '*GLANCE\_DBPASS*';

exit

5.1.5 创建表

su -s /bin/sh -c "glance-manage db\_sync" glance

5.1.6 为 Image Service 创建一个 glance 用户，用于在 Identity Service 中认证。

keystone user-create --name=glance --pass=*GLANCE\_PASS\_SUR* \

--email=*glance@example.com*

+----------+----------------------------------+

| Property | Value |

+----------+----------------------------------+

| email | glance@example.com |

| enabled | True |

| id | ec23fe1110564d3db69693e4df449bc5 |

| name | glance |

| username | glance |

+----------+----------------------------------+

keystone user-role-add --user=glance --tenant=service --role=admin

5.1.7 配置 Image Service 向 Identity Service 认证。

openstack-config --set /etc/glance/glance-api.conf keystone\_authtoken \

auth\_uri http://*controller*:5000

openstack-config --set /etc/glance/glance-api.conf keystone\_authtoken \

auth\_host *controller*

openstack-config --set /etc/glance/glance-api.conf keystone\_authtoken \

auth\_port 35357

openstack-config --set /etc/glance/glance-api.conf keystone\_authtoken \

auth\_protocol http

openstack-config --set /etc/glance/glance-api.conf keystone\_authtoken \

admin\_tenant\_name service

openstack-config --set /etc/glance/glance-api.conf keystone\_authtoken \

admin\_user glance

openstack-config --set /etc/glance/glance-api.conf keystone\_authtoken \

admin\_password *GLANCE\_PASS\_SUR*

openstack-config --set /etc/glance/glance-api.conf paste\_deploy \

flavor keystone

openstack-config --set /etc/glance/glance-registry.conf keystone\_authtoken \

auth\_uri http://*controller*:5000

openstack-config --set /etc/glance/glance-registry.conf keystone\_authtoken \

auth\_host *controller*

openstack-config --set /etc/glance/glance-registry.conf keystone\_authtoken \

auth\_port 35357

openstack-config --set /etc/glance/glance-registry.conf keystone\_authtoken \

auth\_protocol http

openstack-config --set /etc/glance/glance-registry.conf keystone\_authtoken \

admin\_tenant\_name service

openstack-config --set /etc/glance/glance-registry.conf keystone\_authtoken \

admin\_user glance

openstack-config --set /etc/glance/glance-registry.conf keystone\_authtoken \

admin\_password *GLANCE\_PASS\_SUR*

openstack-config --set /etc/glance/glance-registry.conf paste\_deploy \

flavor keystone

5.1.8 向 Identity service 注册 Image Service ，以便其他服务能定位到他。

keystone service-create --name=glance --type=image \

--description="OpenStack Image Service"

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| description | OpenStack Image Service |

| enabled | True |

| id | 7601c26998da4301b48f8005172658f2 |

| name | glance |

| type | image |

+-------------+----------------------------------+

keystone endpoint-create \

--service-id=$(keystone service-list | awk '/ image / {print $2}') \

--publicurl=http://*controller*:9292 \

--internalurl=http://*controller*:9292 \

--adminurl=http://*controller*:9292

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| adminurl | http://controller:9292 |

| id | 2c3bdb44962846538f209ade3a3bf5da |

| internalurl | http://controller:9292 |

| publicurl | http://controller:9292 |

| region | regionOne |

| service\_id | 7601c26998da4301b48f8005172658f2 |

+-------------+----------------------------------+

5.1.9 启动服务，并配置随机启动

systemctl enable openstack-glance-api.service openstack-glance-registry.service

systemctl start openstack-glance-api.service openstack-glance-registry.service

5.2 验证 Image Service 安装

5.2.1 下载镜像

mkdir /tmp/images

cd /tmp/images/

yum install -y wget

wget http://cdn.download.cirros-cloud.net/0.3.2/cirros-0.3.2-x86\_64-disk.img

5.2.2 上传镜像

source /root/admin-openrc.sh

glance image-create --name "cirros-0.3.2-x86\_64" --disk-format qcow2 \

--container-format bare --is-public True --progress < cirros-0.3.2-x86\_64-disk.img

+------------------+--------------------------------------+

| Property | Value |

+------------------+--------------------------------------+

| checksum | 64d7c1cd2b6f60c92c14662941cb7913 |

| container\_format | bare |

| created\_at | 2014-08-26T06:35:58 |

| deleted | False |

| deleted\_at | None |

| disk\_format | qcow2 |

| id | cdcf7239-a2fd-43a9-ae4e-7aba4c8dc7b9 |

| is\_public | True |

| min\_disk | 0 |

| min\_ram | 0 |

| name | cirros-0.3.2-x86\_64 |

| owner | 219264c7a961437eb0066f357d8bc8c2 |

| protected | False |

| size | 13167616 |

| status | active |

| updated\_at | 2014-08-26T06:35:58 |

| virtual\_size | None |

+------------------+--------------------------------------+

5.2.3 确认上传，并显示属性

glance image-list

+--------------------------------------+---------------------+-------------+------------------+----------+--------+

| ID | Name | Disk Format | Container Format | Size | Status |

+--------------------------------------+---------------------+-------------+------------------+----------+--------+

| cdcf7239-a2fd-43a9-ae4e-7aba4c8dc7b9 | cirros-0.3.2-x86\_64 | qcow2 | bare | 13167616 | active |

+--------------------------------------+---------------------+-------------+------------------+----------+--------+

6. 配置 Compute Service

6.1 安装 Compute controller services

6.1.1 在 Controller Node 上安装 Compute packages

yum install -y openstack-nova-api openstack-nova-cert openstack-nova-conductor \

openstack-nova-console openstack-nova-novncproxy openstack-nova-scheduler \

python-novaclient

6.1.2 配置数据库

openstack-config --set /etc/nova/nova.conf \

database connection mysql://nova:NOVA\_DBPASS\_SUR@controller/nova

6.1.3 配置 Rabbitmq 消息代理

openstack-config --set /etc/nova/nova.conf DEFAULT \

rpc\_backend rabbit

openstack-config --set /etc/nova/nova.conf DEFAULT \

rabbit\_host controller

openstack-config --set /etc/nova/nova.conf DEFAULT \

rabbit\_password RABBIT\_GUEST\_PASS\_SUR

6.1.4 设置 VNC 控制台的 IP 为 controller node 的 Managment IP 。

openstack-config --set /etc/nova/nova.conf DEFAULT my\_ip 10.0.33.11

openstack-config --set /etc/nova/nova.conf DEFAULT vncserver\_listen 10.0.33.11

openstack-config --set /etc/nova/nova.conf DEFAULT vncserver\_proxyclient\_address 10.0.33.11

6.1.5 创建数据库

mysql -u root -pMYSQL\_ROOT\_PASS\_SUR

CREATE DATABASE nova;

GRANT ALL PRIVILEGES ON nova.\* TO 'nova'@'localhost' \

IDENTIFIED BY 'NOVA\_DBPASS\_SUR';

GRANT ALL PRIVILEGES ON nova.\* TO 'nova'@'%' \

IDENTIFIED BY 'NOVA\_DBPASS\_SUR';

Exit

6.1.6 创建表

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

6.1.7 创建 nova 用户

keystone user-create --name=nova --pass=*NOVA\_PASS\_SUR* --email=*nova@example.com*

+----------+----------------------------------+

| Property | Value |

+----------+----------------------------------+

| email | nova@example.com |

| enabled | True |

| id | 895635c7ca9d48cfad29993bb1521326 |

| name | nova |

| username | nova |

+----------+----------------------------------+

keystone user-role-add --user=nova --tenant=service --role=admin

6.1.8 配置认证信息

openstack-config --set /etc/nova/nova.conf DEFAULT auth\_strategy keystone

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_uri http://controller:5000/v2.0

openstack-config --set /etc/nova/nova.conf keystone\_authtoken identity\_uri http://controller:35357

openstack-config --set /etc/nova/nova.conf keystone\_authtoken admin\_tenant\_name service

openstack-config --set /etc/nova/nova.conf keystone\_authtoken admin\_user nova

openstack-config --set /etc/nova/nova.conf keystone\_authtoken admin\_password NOVA\_PASS\_SUR

6.1.9 注册服务

keystone service-create --name=nova --type=compute \

--description="OpenStack Compute"

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| description | OpenStack Compute |

| enabled | True |

| id | 978b20c198f5426799d98e314bce4c13 |

| name | nova |

| type | compute |

+-------------+----------------------------------+

keystone endpoint-create \

--service-id=$(keystone service-list | awk '/ compute / {print $2}') \

--publicurl=http://*controller*:8774/v2/%\(tenant\_id\)s \

--internalurl=http://*controller*:8774/v2/%\(tenant\_id\)s \

--adminurl=http://*controller*:8774/v2/%\(tenant\_id\)s

+-------------+-----------------------------------------+

| Property | Value |

+-------------+-----------------------------------------+

| adminurl | http://controller:8774/v2/%(tenant\_id)s |

| id | 3c2c449cb2b7421a876617a1003088ff |

| internalurl | http://controller:8774/v2/%(tenant\_id)s |

| publicurl | http://controller:8774/v2/%(tenant\_id)s |

| region | regionOne |

| service\_id | 978b20c198f5426799d98e314bce4c13 |

+-------------+-----------------------------------------+

6.1.10 将日志等级调为 Debug

sed -i 's/#debug=false/debug=true/g' /etc/nova/nova.conf

6.1.11 启动服务，并配置随机启动

for i in api cert consoleauth scheduler conductor novncproxy;\

do systemctl enable openstack-nova-$i.service;\

systemctl start openstack-nova-$i.service;done;

6.1.12 验证配置

nova image-list

+--------------------------------------+---------------------+--------+--------+

| ID | Name | Status | Server |

+--------------------------------------+---------------------+--------+--------+

| cdcf7239-a2fd-43a9-ae4e-7aba4c8dc7b9 | cirros-0.3.2-x86\_64 | ACTIVE | |

+--------------------------------------+---------------------+--------+--------+

6.2 配置一台 Compute node

6.2.1 在 Compute node 上安装 Compute packages

yum install -y openstack-nova-compute sysfsutils

6.2.2 编辑 /etc/nova/nova.conf 配置文件

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_uri http://controller:5000/v2.0

openstack-config --set /etc/nova/nova.conf keystone\_authtoken identity\_uri http://controller:35357

openstack-config --set /etc/nova/nova.conf keystone\_authtoken admin\_tenant\_name service

openstack-config --set /etc/nova/nova.conf keystone\_authtoken admin\_user nova

openstack-config --set /etc/nova/nova.conf keystone\_authtoken admin\_password NOVA\_PASS\_SUR

6.2.3 配置消息代理

openstack-config --set /etc/nova/nova.conf DEFAULT \

rpc\_backend rabbit

openstack-config --set /etc/nova/nova.conf DEFAULT \

rabbit\_host controller

openstack-config --set /etc/nova/nova.conf DEFAULT \

rabbit\_password RABBIT\_GUEST\_PASS\_SUR

6.2.4 配置 VNC 远程控制台，以便访问实例

openstack-config --set /etc/nova/nova.conf DEFAULT my\_ip 10.0.33.31

openstack-config --set /etc/nova/nova.conf DEFAULT vnc\_enabled True

openstack-config --set /etc/nova/nova.conf DEFAULT vncserver\_listen 0.0.0.0

openstack-config --set /etc/nova/nova.conf DEFAULT vncserver\_proxyclient\_address 10.0.33.31

openstack-config --set /etc/nova/nova.conf \

DEFAULT novncproxy\_base\_url <http://10.0.33.11:6080/vnc_auto.html>

6.2.5 指定 Image Service 服务

openstack-config --set /etc/nova/nova.conf glance host controller

6.2.6 看是否支持虚拟化，结果若为0以上整数，则支持

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

6.2.7 启动服务，并配置随机启动

systemctl enable libvirtd.service openstack-nova-compute.service

systemctl start libvirtd.service openstack-nova-compute.service

7. 添加 Networking Service

7.1 Flat网络模式，跳过

7.2 本教程采用gre网络模式

7.2.1 安装配置在 Controller Node

7.2.1.1 创建数据库及用户

mysql -u root -pMYSQL\_ROOT\_PASS\_SUR

CREATE DATABASE neutron;

GRANT ALL PRIVILEGES ON neutron.\* TO 'neutron'@'localhost' \

IDENTIFIED BY 'NEUTRON\_DBPASS\_SUR';

GRANT ALL PRIVILEGES ON neutron.\* TO 'neutron'@'%' \

IDENTIFIED BY 'NEUTRON\_DBPASS\_SUR';

exit

source /etc/admin-openrc.sh

7.2.1.2 创建用户及注册服务

keystone user-create --name neutron --pass NEUTRON\_PASS\_SUR --email neutron@example.com

keystone user-role-add --user neutron --tenant service --role admin

keystone service-create --name neutron --type network --description "OpenStack Networking"

keystone endpoint-create \

--service-id $(keystone service-list | awk '/ network / {print $2}') \

--publicurl http://controller:9696 \

--adminurl http://controller:9696 \

--internalurl http://controller:9696 \

--region regionOne

7.2.1.3 安装相关软件包

yum install -y openstack-neutron openstack-neutron-ml2 python-neutronclient which

7.2.1.4 配置数据库连接信息

openstack-config --set /etc/neutron/neutron.conf database connection \

mysql://neutron:NEUTRON\_DBPASS\_SUR@controller/neutron

7.2.1.5 配置消息代理

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

rpc\_backend rabbit

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

rabbit\_host controller

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

rabbit\_password RABBIT\_GUEST\_PASS\_SUR

7.2.1.6 配置服务认证相关信息

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

auth\_strategy keystone

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

auth\_uri http://controller:5000/v2.0

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

identity\_uri http://controller:35357

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

admin\_tenant\_name service

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

admin\_user neutron

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

admin\_password NEUTRON\_PASS\_SUR

7.2.1.6 配置ML2

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

core\_plugin ml2

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

service\_plugins router

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

allow\_overlapping\_ips True

7.2.1.7 配置消息通知相关

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

notify\_nova\_on\_port\_status\_changes True

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

notify\_nova\_on\_port\_data\_changes True

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

nova\_url http://controller:8774/v2

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

nova\_admin\_auth\_url http://controller:35357/v2.0

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

nova\_region\_name regionOne

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

nova\_admin\_username nova

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

nova\_admin\_tenant\_id $(keystone tenant-list | awk '/ service / { print $2 }')

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

nova\_admin\_password NOVA\_PASS\_SUR

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

verbose True

7.2.1.8 配置ML2插件

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2 \

type\_drivers flat,gre

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2 \

tenant\_network\_types gre

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2 \

mechanism\_drivers openvswitch

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2\_type\_gre \

tunnel\_id\_ranges 1:1000

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini securitygroup \

enable\_security\_group True

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini securitygroup \

enable\_ipset True

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini securitygroup \

firewall\_driver neutron.agent.linux.iptables\_firewall.OVSHybridIptablesFirewallDriver

7.2.1.9 修改计算服务配置文件

openstack-config --set /etc/nova/nova.conf DEFAULT \

network\_api\_class nova.network.neutronv2.api.API

openstack-config --set /etc/nova/nova.conf DEFAULT \

security\_group\_api neutron

openstack-config --set /etc/nova/nova.conf DEFAULT \

linuxnet\_interface\_driver nova.network.linux\_net.LinuxOVSInterfaceDriver

openstack-config --set /etc/nova/nova.conf DEFAULT \

firewall\_driver nova.virt.firewall.NoopFirewallDriver

openstack-config --set /etc/nova/nova.conf neutron \

url http://controller:9696

openstack-config --set /etc/nova/nova.conf neutron \

auth\_strategy keystone

openstack-config --set /etc/nova/nova.conf neutron \

admin\_auth\_url http://controller:35357/v2.0

openstack-config --set /etc/nova/nova.conf neutron \

admin\_tenant\_name service

openstack-config --set /etc/nova/nova.conf neutron \

admin\_username neutron

openstack-config --set /etc/nova/nova.conf neutron \

admin\_password NEUTRON\_PASS\_SUR

7.2.1.10 完成安装

ln -s /etc/neutron/plugins/ml2/ml2\_conf.ini /etc/neutron/plugin.ini

su -s /bin/sh -c "neutron-db-manage --config-file /etc/neutron/neutron.conf \

--config-file /etc/neutron/plugins/ml2/ml2\_conf.ini upgrade juno" neutron

systemctl restart openstack-nova-api.service openstack-nova-scheduler.service \

openstack-nova-conductor.service

systemctl enable neutron-server.service

systemctl start neutron-server.service

7.2.1.11 验证

source /etc/admin-openrc.sh

neutron ext-list

7.2.2 安装网络服务（还是在Controller Node上）

7.2.2.1 编辑 /etc/sysctl.conf 文件

sed -i "/sysctl.conf(5)/a\\

net.ipv4.ip\_forward=1" /etc/sysctl.conf

sed -i "/net.ipv4.ip\_forward=1/a\\

net.ipv4.conf.all.rp\_filter=0" /etc/sysctl.conf

sed -i "/net.ipv4.conf.all.rp\_filter=0/a\\

net.ipv4.conf.default.rp\_filter=0" /etc/sysctl.conf

cat /etc/sysctl.conf

sysctl -p

7.2.2.2 安装网络组件

yum install -y openstack-neutron openstack-neutron-ml2 openstack-neutron-openvswitch

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2\_type\_flat \

flat\_networks external

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ovs \

local\_ip 192.168.33.11

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ovs \

enable\_tunneling True

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ovs \

bridge\_mappings external:br-ex

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini agent \

tunnel\_types gre

7.2.2.3 配置L3 DHCP agent等

openstack-config --set /etc/neutron/l3\_agent.ini DEFAULT \

interface\_driver neutron.agent.linux.interface.OVSInterfaceDriver

openstack-config --set /etc/neutron/l3\_agent.ini DEFAULT \

use\_namespaces True

openstack-config --set /etc/neutron/l3\_agent.ini DEFAULT \

external\_network\_bridge br-ex

openstack-config --set /etc/neutron/l3\_agent.ini DEFAULT \

router\_delete\_namespaces True

openstack-config --set /etc/neutron/l3\_agent.ini DEFAULT \

verbose True

openstack-config --set /etc/neutron/dhcp\_agent.ini DEFAULT \

interface\_driver neutron.agent.linux.interface.OVSInterfaceDriver

openstack-config --set /etc/neutron/dhcp\_agent.ini DEFAULT \

dhcp\_driver neutron.agent.linux.dhcp.Dnsmasq

openstack-config --set /etc/neutron/dhcp\_agent.ini DEFAULT \

use\_namespaces True

openstack-config --set /etc/neutron/dhcp\_agent.ini DEFAULT \

dhcp\_delete\_namespaces True

openstack-config --set /etc/neutron/dhcp\_agent.ini DEFAULT \

verbose True

openstack-config --set /etc/neutron/dhcp\_agent.ini DEFAULT \

dnsmasq\_config\_file /etc/neutron/dnsmasq-neutron.conf

cat > /etc/neutron/dnsmasq-neutron.conf<<eof

dhcp-option-force=26,1454

eof

killall dnsmasq

7.2.2.4 配置metadata agent

openstack-config --set /etc/neutron/metadata\_agent.ini DEFAULT \

auth\_url http://controller:5000/v2.0

openstack-config --set /etc/neutron/metadata\_agent.ini DEFAULT \

auth\_region regionOne

openstack-config --set /etc/neutron/metadata\_agent.ini DEFAULT \

admin\_tenant\_name service

openstack-config --set /etc/neutron/metadata\_agent.ini DEFAULT \

admin\_user neutron

openstack-config --set /etc/neutron/metadata\_agent.ini DEFAULT \

admin\_password NEUTRON\_PASS\_SUR

openstack-config --set /etc/neutron/metadata\_agent.ini DEFAULT \

nova\_metadata\_ip controller

openstack-config --set /etc/neutron/metadata\_agent.ini DEFAULT \

metadata\_proxy\_shared\_secret METADATA\_SECRET\_SUR

openstack-config --set /etc/neutron/metadata\_agent.ini DEFAULT \

verbose True

7.2.2.5 修改计算服务的配置文件

openstack-config --set /etc/nova/nova.conf neutron \

service\_metadata\_proxy True

openstack-config --set /etc/nova/nova.conf neutron \

metadata\_proxy\_shared\_secret METADATA\_SECRET\_SUR

systemctl restart openstack-nova-api.service

systemctl enable openvswitch.service

systemctl start openvswitch.service

ovs-vsctl add-br br-ex

ovs-vsctl add-port br-ex eth3

ethtool -K eth3 gro off

# ln -s /etc/neutron/plugins/ml2/ml2\_conf.ini /etc/neutron/plugin.ini

cp /usr/lib/systemd/system/neutron-openvswitch-agent.service \

/usr/lib/systemd/system/neutron-openvswitch-agent.service.orig

sed -i 's,plugins/openvswitch/ovs\_neutron\_plugin.ini,plugin.ini,g' \

/usr/lib/systemd/system/neutron-openvswitch-agent.service

for i in openvswitch l3 dhcp metadata;do \

systemctl enable neutron-$i-agent.service;systemctl start neutron-$i-agent.service;done

7.2.2.6 验证

systemctl enable neutron-ovs-cleanup.service

source /etc/admin-openrc.sh

neutron agent-list

7.2.3 配置Compute节点（在compute1上）

7.2.3.1 编辑 /etc/sysctl.conf 文件

sed -i "/sysctl.conf(5)/a\\

net.ipv4.conf.all.rp\_filter=0" /etc/sysctl.conf

sed -i "/net.ipv4.conf.all.rp\_filter=0/a\\

net.ipv4.conf.default.rp\_filter=0" /etc/sysctl.conf

cat /etc/sysctl.conf

sysctl -p

7.2.3.2 安装软件包

yum install -y openstack-neutron-ml2 openstack-neutron-openvswitch

7.2.3.3 配置消息队列

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

rpc\_backend rabbit

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

rabbit\_host controller

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

rabbit\_password RABBIT\_GUEST\_PASS\_SUR

7.2.3.3 配置认证信息

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

auth\_strategy keystone

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

auth\_uri http://controller:5000/v2.0

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

identity\_uri http://controller:35357

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

admin\_tenant\_name service

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

admin\_user neutron

openstack-config --set /etc/neutron/neutron.conf keystone\_authtoken \

admin\_password NEUTRON\_PASS\_SUR

7.2.3.4 启用ML2插件等

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

core\_plugin ml2

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

service\_plugins router

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

allow\_overlapping\_ips True

openstack-config --set /etc/neutron/neutron.conf DEFAULT \

verbose True

7.2.3.5 配置ML2插件

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2 \

type\_drivers flat,gre

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2 \

tenant\_network\_types gre

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2 \

mechanism\_drivers openvswitch

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ml2\_type\_gre \

tunnel\_id\_ranges 1:1000

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini securitygroup \

enable\_security\_group True

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini securitygroup \

enable\_ipset True

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini securitygroup \

firewall\_driver neutron.agent.linux.iptables\_firewall.OVSHybridIptablesFirewallDriver

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ovs \

local\_ip 192.168.33.31

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini ovs \

enable\_tunneling True

openstack-config --set /etc/neutron/plugins/ml2/ml2\_conf.ini agent \

tunnel\_types gre

7.2.3.6 配置OVS服务

systemctl enable openvswitch.service

systemctl start openvswitch.service

7.2.3.7 配置计算服务使用的网络服务

openstack-config --set /etc/nova/nova.conf DEFAULT \

network\_api\_class nova.network.neutronv2.api.API

openstack-config --set /etc/nova/nova.conf DEFAULT \

security\_group\_api neutron

openstack-config --set /etc/nova/nova.conf DEFAULT \

linuxnet\_interface\_driver nova.network.linux\_net.LinuxOVSInterfaceDriver

openstack-config --set /etc/nova/nova.conf DEFAULT \

firewall\_driver nova.virt.firewall.NoopFirewallDriver

openstack-config --set /etc/nova/nova.conf neutron \

url http://controller:9696

openstack-config --set /etc/nova/nova.conf neutron \

auth\_strategy keystone

openstack-config --set /etc/nova/nova.conf neutron \

admin\_auth\_url http://controller:35357/v2.0

openstack-config --set /etc/nova/nova.conf neutron \

admin\_tenant\_name service

openstack-config --set /etc/nova/nova.conf neutron \

admin\_username neutron

openstack-config --set /etc/nova/nova.conf neutron \

admin\_password NEUTRON\_PASS\_SUR

ln -s /etc/neutron/plugins/ml2/ml2\_conf.ini /etc/neutron/plugin.ini

cp /usr/lib/systemd/system/neutron-openvswitch-agent.service \

/usr/lib/systemd/system/neutron-openvswitch-agent.service.orig

sed -i 's,plugins/openvswitch/ovs\_neutron\_plugin.ini,plugin.ini,g' \

/usr/lib/systemd/system/neutron-openvswitch-agent.service

7.2.3.8 完成安装

systemctl restart openstack-nova-compute.service

systemctl enable neutron-openvswitch-agent.service

systemctl start neutron-openvswitch-agent.service

7.2.3.9 验证

source /etc/admin-openrc.sh

neutron agent-list

7.2.4 创建初始网络

7.2.4.1 创建公共网络

source /etc/admin-openrc.sh

neutron net-create Pub-Net --router:external True \

--provider:physical\_network external --provider:network\_type flat

neutron subnet-create Pub-Net --name Pub-Subnet \

--allocation-pool start=10.0.38.11,end=10.0.38.111 \

--disable-dhcp --gateway 10.0.0.1 10.0.38.0/16 \

--dns-nameserver 10.0.0.1 \

--dns-nameserver 8.8.8.8

7.2.4.2 创建租户网络

source /etc/demo-openrc.sh

neutron net-create Pri-Net

neutron subnet-create Pri-Net --name Pri-Subnet \

--gateway 10.1.1.1 10.1.1.0/24 \

--dns-nameserver 8.8.4.4 \

--dns-nameserver 1.2.4.8 \

--dns-nameserver 114.114.114.114

neutron router-create Pri-Router

neutron router-interface-add Pri-Router Pri-Subnet

neutron router-gateway-set Pri-Router Pub-Net

7.2.4.3 验证

ping -c 4 10.0.38.11

8. 添加dashboard

8.1 安装 dashboard

8.1.1 在控制节点上安装 dashboard

yum install -y openstack-dashboard httpd mod\_wsgi memcached python-memcached

8.1.2 修改 vim /etc/openstack-dashboard/local\_settings 中的 CACHES['default']['LOCATION'] 值为如下：

CACHES = {

'default': {

'BACKEND' : 'django.core.cache.backends.memcached.MemcachedCache',

'LOCATION' : '127.0.0.1:11211'

}

}

修改时区 TIME\_ZONE = "UTC"

8.1.3 编辑 /etc/openstack-dashboard/local\_settings 中：

ALLOWED\_HOSTS = ['\*']

sed -i "s/ALLOWED\_HOSTS = \['horizon.example.com', 'localhost'\]/ALLOWED\_HOSTS = \['\*'\]/g" \

/etc/openstack-dashboard/local\_settings

8.1.4 编辑 /etc/openstack-dashboard/local\_settings 中：

OPENSTACK\_HOST = "controller"

sed -i 's/OPENSTACK\_HOST = "127.0.0.1"/OPENSTACK\_HOST = "controller"/g' \

/etc/openstack-dashboard/local\_settings

8.1.5 打开 HTTP 权限

setsebool -P httpd\_can\_network\_connect on

chown -R apache:apache /usr/share/openstack-dashboard/static

8.1.6 启动服务，并配置随机启动

systemctl enable httpd.service memcached.service

systemctl start httpd.service memcached.service

8.1.7 访问 dashboard

http://controller/dashboard controller 替换成对应 IP，如：

http://10.0.33.11/dashboard

admin ADMIN\_PASS\_SUR

9. 添加 Block Storage service

9.1 在 Controller Node上配置 Block Storage service controller

9.1.1 为 Block Storage service 安装包

yum install -y openstack-cinder python-cinderclient python-oslo-db

9.1.2 配置数据库

openstack-config --set /etc/cinder/cinder.conf \

database connection mysql://cinder:CINDER\_DBPASS\_SUR@controller/cinder

9.1.3 创建数据库

mysql -u root -pMYSQL\_ROOT\_PASS\_SUR

CREATE DATABASE cinder;

GRANT ALL PRIVILEGES ON cinder.\* TO 'cinder'@'localhost' \

IDENTIFIED BY 'CINDER\_DBPASS\_SUR';

GRANT ALL PRIVILEGES ON cinder.\* TO 'cinder'@'%' \

IDENTIFIED BY 'CINDER\_DBPASS\_SUR';

exit

9.1.4 创建表（忽略报错）

su -s /bin/sh -c "cinder-manage db sync" cinder

9.1.5 创建 cinder 用户

keystone user-create --name cinder --pass CINDER\_PASS\_SUR --email cinder@example.com

+----------+----------------------------------+

| Property | Value |

+----------+----------------------------------+

| email | cinder@example.com |

| enabled | True |

| id | 90cbd9a437f74fbbb82035048b4531c2 |

| name | cinder |

| username | cinder |

+----------+----------------------------------+

keystone user-role-add --user cinder --tenant service --role admin

9.1.6 编辑 /etc/cinder/cinder.conf 配置文件

openstack-config --set /etc/cinder/cinder.conf DEFAULT \

auth\_strategy keystone

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

auth\_uri http://controller:5000/v2.0

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

identity\_uri http://controller:35357

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

admin\_tenant\_name service

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

admin\_user cinder

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

admin\_password CINDER\_PASS\_SUR

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT my\_ip 10.0.33.11

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT verbose True

9.1.7 配置消息代理

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT rpc\_backend rabbit

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT rabbit\_host controller

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT rabbit\_password RABBIT\_GUEST\_PASS\_SUR

9.1.8 注册服务

keystone service-create --name cinder --type volume --description "OpenStack Block Storage"

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| description | OpenStack Block Storage |

| enabled | True |

| id | b992c8c74a9f4794aef3b26c8dd401a0 |

| name | cinder |

| type | volume |

+-------------+----------------------------------+

keystone endpoint-create \

--service-id $(keystone service-list | awk '/ volume / {print $2}') \

--publicurl http://controller:8776/v1/%\(tenant\_id\)s \

--internalurl http://controller:8776/v1/%\(tenant\_id\)s \

--adminurl http://controller:8776/v1/%\(tenant\_id\)s \

--region regionOne

+-------------+-----------------------------------------+

| Property | Value |

+-------------+-----------------------------------------+

| adminurl | http://controller:8776/v1/%(tenant\_id)s |

| id | a500cd3b61c74cae8851b3317257ddf6 |

| internalurl | http://controller:8776/v1/%(tenant\_id)s |

| publicurl | http://controller:8776/v1/%(tenant\_id)s |

| region | regionOne |

| service\_id | b992c8c74a9f4794aef3b26c8dd401a0 |

+-------------+-----------------------------------------+

9.1.9 再注册一个服务终端

keystone service-create --name cinderv2 --type volumev2 --description="OpenStack Block Storage v2"

+-------------+----------------------------------+

| Property | Value |

+-------------+----------------------------------+

| description | OpenStack Block Storage v2 |

| enabled | True |

| id | 6229f01a962f486b863588e43d03535f |

| name | cinderv2 |

| type | volumev2 |

+-------------+----------------------------------+

keystone endpoint-create \

--service-id $(keystone service-list | awk '/ volumev2 / {print $2}') \

--publicurl http://controller:8776/v2/%\(tenant\_id\)s \

--internalurl http://controller:8776/v2/%\(tenant\_id\)s \

--adminurl http://controller:8776/v2/%\(tenant\_id\)s \

--region regionOne

+-------------+-----------------------------------------+

| Property | Value |

+-------------+-----------------------------------------+

| adminurl | http://controller:8776/v2/%(tenant\_id)s |

| id | e022fe3adc4b4240887b87a41e9ae465 |

| internalurl | http://controller:8776/v2/%(tenant\_id)s |

| publicurl | http://controller:8776/v2/%(tenant\_id)s |

| region | regionOne |

| service\_id | 6229f01a962f486b863588e43d03535f |

+-------------+-----------------------------------------+

9.1.10 启动服务，并配置随机启动

systemctl enable openstack-cinder-api.service openstack-cinder-scheduler.service

systemctl start openstack-cinder-api.service openstack-cinder-scheduler.service

9.1.10 将日志等级调为 Debug

sed -i 's/#debug=false/debug=true/g' /etc/cinder/cinder.conf

9.2 配置 Block Storage service node

9.2.1 本教程将 Controller Node 作为 Block Storage Service Node。Controller Node 上需要挂载一块空白硬盘，假设为 /dev/sdb 。

9.2.2 创建 cinder-volumes 卷组

pvcreate /dev/sdb

vgcreate cinder-volumes /dev/sdb

9.2.3 向 /etc/lvm/lvm.conf 添加设备过滤

devices {

...

filter = [ "a/sda1/", "a/sdb/", "r/.\*/"]

...

}

9.2.4 安装 Block Storage service 软件包

yum install -y lvm2 openstack-cinder targetcli python-oslo-db MySQL-python

9.2.5 配置认证信息，编辑 /etc/cinder/cinder.conf 文件。可以跳过，因为配置 Block Storage service controller 的时候已经配置过了。

openstack-config --set /etc/cinder/cinder.conf DEFAULT \

auth\_strategy keystone

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

auth\_uri http://controller:5000/v2.0

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

identity\_uri http://controller:35357

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

admin\_tenant\_name service

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

admin\_user cinder

openstack-config --set /etc/cinder/cinder.conf keystone\_authtoken \

admin\_password CINDER\_PASS\_SUR

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT my\_ip 10.0.33.11

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT verbose True

9.2.6 配置消息代理（也可以跳过）

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT rpc\_backend rabbit

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT rabbit\_host controller

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT rabbit\_password RABBIT\_GUEST\_PASS\_SUR

9.2.7 配置数据库连接信息（也可以跳过）

openstack-config --set /etc/cinder/cinder.conf \

database connection mysql://cinder:*CINDER\_DBPASS*@*controller*/cinder

9.2.8 配置 Image Service 主机

openstack-config --set /etc/cinder/cinder.conf \

DEFAULT glance\_host *controller*

9.2.9 在 vim /etc/tgt/targets.conf 中添加：

include /etc/cinder/volumes/\*

sed -i '/#include \/etc\/tgt\/temp\/\\*.conf/a\include \/etc\/cinder\/volumes\/\\*' \

/etc/tgt/targets.conf

9.2.10 修复云盘不能挂在到虚拟机的问题

sed -i 's/#iscsi\_helper=tgtadm/iscsi\_helper=tgtadm/g' /etc/cinder/cinder.conf

9.2.11 启动服务，并配置随机启动：

systemctl enable openstack-cinder-volume.service target.service

systemctl restart openstack-cinder-volume.service target.service

9.3 验证 Block Storage 的安装

9.3.1 读取环境变量脚本

source /root/demo-openrc.sh

9.3.2 创建一个卷

cinder create --display-name myVolume 1

+---------------------+--------------------------------------+

| Property | Value |

+---------------------+--------------------------------------+

| attachments | [] |

| availability\_zone | nova |

| bootable | false |

| created\_at | 2014-08-26T10:05:40.781094 |

| display\_description | None |

| display\_name | myVolume |

| encrypted | False |

| id | c2dde81d-8976-4cf5-98d8-9c685a2841bf |

| metadata | {} |

| size | 1 |

| snapshot\_id | None |

| source\_volid | None |

| status | creating |

| volume\_type | None |

+---------------------+--------------------------------------+

9.3.3 查看卷信息

cinder list

+--------------------------------------+-----------+--------------+------+-------------+----------+-------------+

| ID | Status | Display Name | Size | Volume Type | Bootable | Attached to |

+--------------------------------------+-----------+--------------+------+-------------+----------+-------------+

| c2dde81d-8976-4cf5-98d8-9c685a2841bf | available | myVolume | 1 | None | false | |

+--------------------------------------+-----------+--------------+------+-------------+----------+-------------+