环境准备

1，环境规划

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | VM1：controller | VM2：compute | VM3：network | VM3：volume |
| 虚拟机的网卡规划 |  |  |  |  |
| enp0s3: 网卡1，网络地址转换(NAT)，用于VM访问外网 | 自动 | 自动 | 自动 |  |
| enp0s8：网卡2，host-only 2#，用于controller对节点的管理 | 192.168.100.101 | 192.168.100.102 | 192.168.100.103 |  |
| enp0s9：网卡3，host-only 3#,用于计算节点间的通讯 | 192.168.110.101 | 192.168.110.102 | 192.168.110.103 |  |
| enp0s10：网卡4，host-only 4#，用于存储节点间的通讯 | 192.168.120.101 | 192.168.120.102 | 192.168.120.103 |  |

NOTES：以后安装的时候最好不要使用host-only这个virtualbox自动生成的网卡，因为他的网段地址会变化，需要自己添加虚拟网卡

2，虚拟机环境安装

1. 安装virtualbox
2. virtualbox配置：管理->全局设定->网络：在host-only里面创建三个网卡，分别为2#，3#，4#，虚拟网卡的网段分别设置为192.168.100.0/24;192.168.110.0/24;192.168.120.0/24
3. 虚拟机安装，主要网卡选择：网卡1选择网络地址转换；网卡2选择host-only->2#，。。。
4. 虚拟机地址配置：
   1. Vim /etc/network/interfaces

# This file describes the network interfaces available on your system

# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/\*

# The loopback network interface

auto lo

iface lo inet loopback

# The primary network interface

auto enp0s3

iface enp0s3 inet dhcp

# The second network interface

auto enp0s8

iface enp0s8 inet static

address 192.168.100.102

netmask 255.255.255.0

# The third network interface

auto enp0s9

iface enp0s9 inet static

address 192.168.110.102

netmask 255.255.255.0

# The fourth network interface for bridge network

auto enp0s10

iface enp0s10 inet static

address 192.168.120.102

netmask 255.255.255.0

root@compute:~#

注释：启动后修改上述配置文件，然后重启网络服务

root@compute: service networking restart

检查地址配置方法：

ip add show

ip route show

route -n

检查DNS配置：more /etc/resolv.conf

注释：如果需要手工配置DNS，在网络配置里面添加如下配置：

# The third network interface

auto enp0s9

iface enp0s9 inet static

address 192.168.110.102

netmask 255.255.255.0

dns-nameservers 10.1.1.1

dns-nameservers 10.1.1.2

dns-search china.huawei.com

配置完后重启networking服务，然后再检查/etc/resolve.conf文件可以看到变化

3，服务安装

1. 安装数据库服务，通常是才采用的mysql，这里安装mariadb-server + phmysql

<https://docs.openstack.org/liberty/zh_CN/install-guide-ubuntu/environment-sql-database.html>

**大坑啊：**注释：上述文档里面介绍的在/etc/mysql/conf.d/mysqld\_openstack.cnf增加配置感觉是不行的。经过实验修改安装好后原有的配置文件里面bind\_address = 127.0.0.1可以实现。如：

vim /etc/mysql/mariadb.conf.d/50-server.cnf

修改[mysqld]下面的

bind\_address = 127.0.0.1 => 网卡地址

用上述文档介绍的方式mysql服务无法启动，不知道为何？

最新新的，不是不可以在conf.d下面加配置文件，而是说配置被mariadb.d下面的文件给覆盖了，注意安装mariadb+mysql的话，配置文件在conf.d和mariadb.d目录下面都有配置，两者不要冲突了。我换种方式修改，把my.cnf里面include文件的顺序换一下。

注释：修改/etc/mysql/conf.d/myslq.cnf是不行的，会导致mysql服务启动失败，只能修改50-sever.cnf那个文件，但是文档要求的其他参数也需要配置，尤其字符集，否则后面会出错。需要在50-server.cnf/50-client.cnf文件里面找到相应的字符集配置，修改为utf8.否则后面keystone启动的时候会出错。

如果配置错误，keystone启动失败时候，修改字符集配置后重启mysql，删除keystone数据库重建即可。DROP DATABASE keystone，注意重建的时候别忘记了添加权限；

参照上面的链接安装数据服务

小贴士：dpkg -l |grep 软件名 查找安装的软件名称，这样可以卸载相应的软件

小贴士：多网卡的时候，如何配置mysql绑定到哪个地址上，查看/etc/mysql/mariadb.conf.d/50-server.cnf里面[mysqld]下面有bind-address = 127.0.0.1的配置，修改这个项就可以了。。

我这里安装了mariadb，如果仅仅是mysql，可能不是这个配置文件；

设置mysql root用户密码的方法

mysql> use mysql;  
Database changed  
mysql> update user set password=password('new password') where user='root';  
Query OK, 4 rows affected (0.00 sec)  
Rows matched: 4  Changed: 4  Warnings: 0  
mysql> flush privileges;  
Query OK, 0 rows affected (0.00 sec)  
mysql> quit

mysql设置远程接入的方法：

mysql> grant all privileges on <databasename>.\* to <username>@’%’ identified by ‘PASSWORD’ with grant option;

// database 如果是\*表示可以访问任何数据库。‘%’，是表示可以从任意远程地址登陆，如果限制地址就明确填地址

examples：设置root账号可以从任意地址访问任意数据库，有素有的权限

mysql> grant all privileges on \*.\* to root@’%’ identified by ‘197419802008’ with grant option;

mysql> flush privileges;

mysql> quit

ubuntu# service mysql restart

此后远程客户就可以如此访问mysql服务器了：

Ubuntu# mysql -h <mysql服务器地址> -u root -p

如果是限制只有select权限，可以这样：

grant select privileges on …

memcache SERVER配置是否正确？？？

小贴士：export 设置环境变量；unset删除环境变量，examples：

export OS\_TOKEN=122345

unset OS\_TOKEN

**大坑啊：被这个害死了，应该是--os-user-domain-name default，而不是ID，同样的，--os-project-domain-id default应该是--os-project-domain-name default**

1. $ openstack --os-auth-url http://controller:35357/v3 \
2. --os-project-domain-id default --os-user-domain-id default \
3. **--os-project-name admin** --os-username admin --os-auth-type password \
4. token issue
5. Password:
6. +------------+----------------------------------+
7. | Field | Value |
8. +------------+----------------------------------+
9. | expires | 2015-03-24T18:55:01Z |
10. | id | ff5ed908984c4a4190f584d826d75fed |
11. | project\_id | cf12a15c5ea84b019aec3dc45580896b |
12. | user\_id | 4d411f2291f34941b30eef9bd797505a |
13. +------------+----------------------------------+

**OpenStack概念：**

区域：region，这个都是用于管理目的，用于安全隔离，就是一个地理的概念

域名/域ID：domain，感觉不区域小一点，也可能不是一个概念，又好像子DC的概念

服务：service，就是各种服务组件的服务，如身份服务，消息服务

端点：endpoint，API的服务端点，就是一个服务的URL，每类服务都提供三个端点：admin，public，internal，通常为了安全把这三个服务端点放在不同的区域进行管理；在创建端点的时候需要指定区域和服务类型。通常admin和public的服务端点是一样的，inertal的端口不一样。这个API接口就是RESTful接口；

项目：project，有点像租户的概念，或者说是用户群，就是租户的意思，后面的版本都用这个概念

用户：user，就是某个租户的用户，用户属于某个项目，但是在创建用户的时候不制定项目，而是在帮当角色的时候把user和project绑定起来的。

角色：role，用于定于租户(project)或者用户的权限的，项目或用户需要绑定一个role，以便赋予某种权限；

可以参看下面的链接：

<http://blog.chinaunix.net/uid-15041-id-4807612.html>

分别对应openstack的命令：

openstack region create 。。。

openstack endpoint create 。。。 //制定所属的区域

openstack domain 。。。 //缺省是default

openstack service 。。。

openstack project

openstack user

openstack role

openstack role add 。。。//绑定到project或user上

注意：注意keystone的环境变量设置，类似 OS\_XXXX；这个在openstack认证的时候需要，设置错误会导致无权执行，可以用export命令查看。当然没有设置环境变量的情况下也可以在命令里面直接制定，如 --os-auth-type password

opensack就是一堆服务提供完整的功能，每个服务都是解耦的，可以有多种方式提供，主要遵从openstack的RESTful接口规范即可；添加一个服务通常需要做几件事情：

1. 在数据库服务中添加数据库，如镜像服务需要在MySQL中添加glance数据库和对应glance账号
2. 在openstack中添加用户，并绑定一个角色，如添加glance用户，绑定到admin角色上； openstack user add … openstack role add …
3. 创建服务实体，openstack service create …
4. 创建服务的三个端点，用于admin public internal
5. 安装组件包
6. 配置组件包

NOTES:OPENSTACK手册：

<https://docs.openstack.org/admin-guide/>

keystone-manage bootstrap --bootstrap-password 197419802008 \

--bootstrap-admin-url http://192.168.100.102:35357/v3/ \

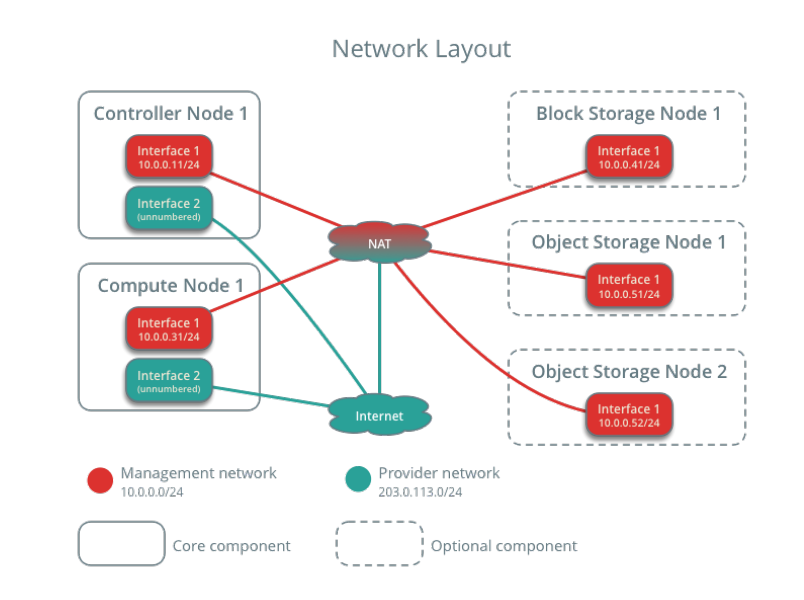
--bootstrap-internal-url http://192.168.100.102:5000/v3/ \

--bootstrap-public-url http://192.168.100.102:5000/v3/ \

--bootstrap-region-id RegionOne

从头来过

**组网是最关键的事情！！！**



红色的是openstack cloud的管理网络，这个管理网络需要访问internet，因为需要安装程序啥的；

蓝色的是内部通讯的网络，但是也需要访问因特网，因为外部网络需要访问openstack的实例，就是openstack拉起的VM。

另外因为我这个环境是安装在虚拟机上，所以还需要提供一个管理虚拟机的网络，用于SSL链接，对VM本身进行管理。

Openstack上述两个网络都通过NAT访问internet网络。

所以每个VM节点的地址规划如下：

1. 管理虚拟机节点的网络，这个采用桥接方式与host通讯，同时也是可以访问interternet网络的。自动分配IP地址，
2. Openstack管理网络：采用host-only 2#网卡，那咋访问inernet网络了？使用20.0.0.0/24网段，gateway 20.0.0.1/24，节点地址分配：

controller node：20.0.0.11/24；

compute：20.0.0.31/24，

object storage：20.0.0.41/51.。。/24

1. Provider网络，这个是openstack计算节点间通讯使用的网络，这个采用host-only 3#网卡，使用203.0.113.0/24 with gateway 203.0.113.1，节点地址分配：

Controller node：203.0.113.11/24

Compute node：203.0.113.31/24

1. 存储服务使用的网络，这个是openstack需要访问存储服务的时候使用的网络，这个采用host-only 4#网卡；采用地址段：192.168.100.0/24，节点地址规划：

Controller node：192.168.100.11/24

Compute node：192.168.100.31/24

如何实现hostonly 2#，3#访问internet网？

虚拟机网卡的分配：

网卡1：openstack管理网络，20.0.0.0/24，gateway 20.0.0.1/24

网卡2：openstack计算服务，203.0.113.0/24

网卡3：openstack网络服务网络，192.168.100.0/24，网关192.168.100.1/24

网卡4：选用NAT网络，用于VM的管理；

前面将的openstack实例要能上因特网实质是通过网络服务节点实现的，就是所有的openstack节点都通过网络服务节点去访问inter网！！！

创建服务的方法：

1. 创建一个数据库和用户，在mysql里面创建
2. 服务凭证，service credentials，即创建一个用户，并授予admin角色，如：

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

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

1. 创建一个服务实体，在openstack里面创建 openstack service create？
2. 创建服务的API端点，openstack里面服务都是RESTful接口，通常需要创建三个API服务端点：admin，public，internal

Openstack的节点就是一个host，计算节点就是上面可以创建虚拟机的host；

计算服务，计算节点，计算服务是提供API供管理虚拟机使用的，计算节点是上面可以创建虚拟机的host。计算节点就是可用的计算资源，计算服务就是管理计算资源，本质上就是管理虚拟机；network服务是提供网络资源管理的服务，如提供一个路由器，一个交换机等；同理，存储服务就是提供存储资源的管理，包括物理存储资源的管理和分配；

所以需要申请这些服务的应用都需要经过身份认证，keystone提供身份认证的基础服务。