# Openstack虚拟化云计算平台详细安装流程报告

## 写在前面的话

Openstack到目前为止（201403），已经包含多个稳定版本（A-H），最新的稳定版本是Havana版，我们现在使用的实验版本也是havana版本。由于openstack的版本相互之间差异比较大，组件的安装命令等也略有相同，所以在参考此文档时要注意区别openstack的版本。

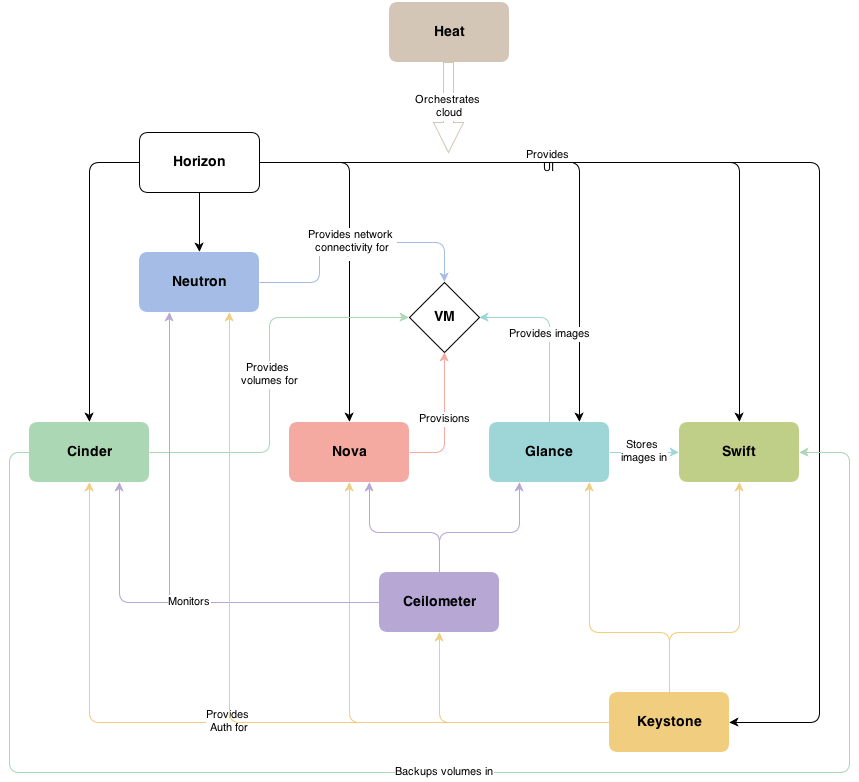
Openstack的各个组件相互之间是通过网络来进行互联互通，不同的组件理论上都可以部署在不同的节点上，可以通过一些的额外的步骤来提高openstack的可用性。基于我们的实验环境和现实情况，我们选择了官网推荐的最经典的安装方式，因此阅读此文档的读者在阅读文档时，注意我们使用的安装方式不是openstack的唯一安装方式。

## Openstack架构

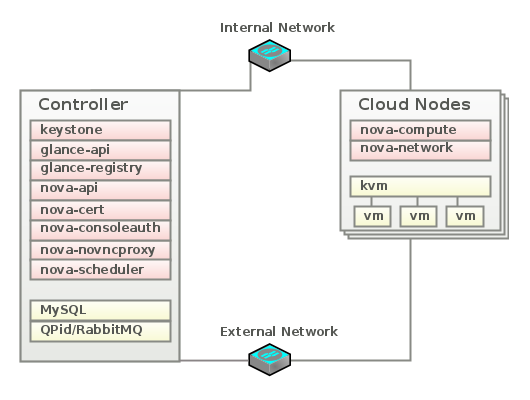
### 2.1基础组件

|  |  |  |  |
| --- | --- | --- | --- |
| **服务** | **组件** | **说明** | **备注** |
| DashBoard | Horizon | 提供WEB界面 |  |
| Computer | Nova | 计算也就是虚拟机 |  |
| Networking | Neutron | 提供给nova网络支持 | 实验环境不需要安装 |
| Object Storage | Swift | 提供对象存储 | 实验环境不需要安装 |
| Block Storage | Cinder | 提供云硬盘给nova |  |
| Identity Sservice | Keystone | 提供所有组件的认证 |  |
| Image Service | Glance | 提供给nova镜像服务 |  |
| Telemetry Service | Cellometer | 监控  cinder,neutron,nova,glance | 实验环境不需要安装 |
| Orchestration Service | Heat | 与AWS cloud兼容 | 实验环境不需要安装 |

### 2.2组件对应关系



### 2.3基础架构



## 基础配置

1. **硬件配置情况介绍：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **机器** | **CPU** | **内存** | **硬盘** | **网卡** | **其他** |
| 虚拟机 |  | 2G | 20G | 双网卡 |  |
| 虚拟机 |  | 2G | 20G | 双网卡 |  |

**2.节点情况介绍：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **机器节点** | **主机名** | **IP地址** | **操作系统** | **用户名密码情况** |
| 主节点 | controller | 10.10.70.21,  192.168.0.10（内网） | CentOS6.5 basicServer | root/rootpass  openstack/openstack |
| 计算节点1 | compute | 10.10.70.22,  192.168.0.11(内网) | CentOS6.5 basicServer | root/rootpass  Openstack/openstack |

**3.网络情况介绍：**

通过一台单独的交换机进行网络访问，因此网络中有大量可用的10.10.70.XX段地址，可以分配给不同的虚拟机。

实验环境的网络速度有限速，所以很多措施是节省网络代码，如果网络条件好，很多步骤可以省略或者替换。

**4.相关密码统计**

|  |  |  |
| --- | --- | --- |
| **所属机器** | **用户名密码** | **用户名密码说明** |
| Controller的用户密码 | root/rootpass | Root用户的密码 |
| openstack/openstack | Openstack的用户名密码 |
| Compute的用户名密码 | root/rootpass | Root用户的密码 |
| openstack/openstack | Openstack的用户名密码 |
| mySQL数据库服务器 | Root/mysqlroot |  |
| Openstack服务密码 | Nova/nova  Keystone/keystone  Glance/glance  Cinder/cinder |  |
| Openstack数据库密码 | Nova/nova  Keystone/keystone  Glance/glance  Cinder/cinder |  |

## 基本的操作系统配置



重要提醒

在进行下面的所有操作时，必须要有root权限，如果root权限不好获取，可以通过sudo命令来获取。

### 4.1基础网络设置

1. **将每个节点的IP设置成静态的IP地址。**

修改文件：

*vi /etc/sysconfig/network-scripts/ifcfg-eth0*

需要注意的内容如下：

DEVICE=eth0 网卡的名称

HWADDR=00:1E:67:24:E7:31 Mac地址的名称

TYPE=Ethernet 网络类型

UUID=c6ee6fd1-467b-4f36-bcdc-307bc678191b

ONBOOT=yes        这是随系统启动而启动的选项

NM\_CONTROLLED=no  这是由系统的networkmanager来管理IP地址， 设置为No

BOOTPROTO=static  设置网卡获取IP地址方式 有static,DHCP,bootp. 分别对应静态指定的 ip地址，通过dhcp协议获得的ip地址，通过bootp协议获得的ip地址

IPADDR=192.168.0.10 如果设置网卡获得ip地址的方式为静态指定，此字段就指定了网卡对应的ip地址

NETMASK=255.255.255.0  子网掩码

ARPCHECK=no

DNS1=10.10.70.211

修改文件：

*vi /etc/sysconfig/network-scripts/ifcfg-eth1*

需要注意的内容如下：

DEVICE=eth0 网卡的名称

HWADDR=00:1E:67:24:E7:31 Mac地址的名称

TYPE=Ethernet 网络类型

UUID=c6ee6fd1-467b-4f36-bcdc-307bc678191b

ONBOOT=yes        这是随系统启动而启动的选项

NM\_CONTROLLED=no  这是由系统的networkmanager来管理IP地址， 设置为No

BOOTPROTO=static  设置网卡获取IP地址方式 有static,DHCP,bootp. 分别对应静态指定的 ip地址，通过dhcp协议获得的ip地址，通过bootp协议获得的ip地址

IPADDR=10.10.70.21 如果设置网卡获得ip地址的方式为静态指定，此字段就指定了网卡对应的ip地址

NETMASK=255.255.255.0  子网掩码

GATEWAY=10.10.70.211 网关地址

ARPCHECK=no

设置完成之后，重启网络：

*service network restart*



重要提醒

1.有些教程中说ifcfg-eth0中的配置网关的GATEWAY关键字是 NETWORK，是不对的 ，实现发现如果设置错误，上不了外网。

2.ARPCHECK=no的作用是解决错误：网络重启报错：Bringing up interface eth0:  Determining if ip address 10.10.70.11 is already in use for device eth0...

1. **设置每个节点的网关**

修改对应网卡的网关的配置文件  
*vi /etc/sysconfig/network*

修改以下内容

NETWORKING=yes(表示系统是否使用网络，一般设置为yes。如果设为no，则不能使用网络，而且很多系统服务程序将无法启动)  
 HOSTNAME=centos(设置本机的主机名，这里设置的主机名要和/etc/hosts中设置的主机名对应)  
 GATEWAY=10.10.70.211(设置本机连接的网关的IP地址。)

设置完成之后，重启网络：

*service network restart*

1. **设置每个节点的DNS**

修改对应网卡的DNS的配置文件

*vi /etc/resolv.conf*

修改一下内容：

nameserver 10.10.70.211

设置完成之后，重启网络：

*service network restart*

1. **设置每个节点的主机名IP对应关系**

1、修改文件：

*vi /etc/sysconfig/network*

设置：HOSTNAME=controller

2、修改文件：

*vi etc/hosts*

在以前的内容上增加：

127.0.0.1 controller

10.10.70.21 controller

10.10.70.22 compute

192.168.0.10 controller

192.168.0.11 compute



重要提醒

设置节点的主机名和IP对应关系时要注意，每个节点都要配置此项，而且每个节点都要配置整个网络中其他节点的IP和主机名的对应关系。这样做的目的是为了在整个网络中，可以通过主机名来访问其他的机器。

1. **关闭每个节点的防火墙以及其他安全策略。**
2. 重启网络

运行以下命令

*service NetworkManager stop*

*service network start*

*chkconfig NetworkManager off*

*chkconfig network on*

1. 关闭防火墙

*service iptables stop*

*chkconfig iptables off*

*service ip6tables stop*

*chkconfig ip6tables off*

1. 关闭SELinux

*setenforce 0*

*vi etc/sysconfig/selinux*

修改SELINUX=permissive



重要提醒

*1. 获取当前 SELinux 运行状态*

#getenforce

可能返回结果有三种：Enforcing、Permissive 和 Disabled。Disabled 代表 SELinux 被禁用，**Permissive 代表仅记录安全警告但不阻止可疑行为**，Enforcing 代表记录警告且阻止可疑行为。

目前常见发行版中，RHEL 和 Fedora 默认设置为 Enforcing，其余的如 openSUSE 等为 Permissive。

*2. 改变 SELinux 运行状态*

setenforce [ Enforcing | Permissive | 1 | 0 ]

该命令可以立刻改变 SELinux 运行状态，在 Enforcing 和 Permissive 之间切换，结果保持至关机。一个典型的用途是看看到底是不是 SELinux 导致某个服务或者程序无法运行。**若是在 setenforce 0 之后服务或者程序依然无法运行，那么就可以肯定不是 SELinux 导致的。**

若是想要**永久变更系统 SELinux 运行环境，可以通过更改配置文件 /etc/sysconfig/selinux 实现**。注意当从 Disabled 切换到 Permissive 或者 Enforcing 模式后需要重启计算机并为整个文件系统重新创建安全标签(touch /.autorelabel && reboot)。

### 4.2时间同步服务器设置

因为涉及到多个节点，所以为了保证每个节点的时间保持一致，教程中提示安装NTP服务。

建议的结构式，将controller所注意NTP服务器，也就是标准时间，让其他的计算节点通过NTP服务来跟controller进行同步。

1. **在每个节点中都安装NTP服务**

*yum install ntp*

*service ntpd start*

*chkconfig ntpd on*

1. **Controller中NTP相关设置**

* 配置NTP服务器

*vi /etc/ntp.conf*

* + 配置需要连接此时间服务器的节点

Hosts on local network are less restricted.

restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap

下添加你要提供时间同步服务的网段，如我的为：

restrict 10.10.70.0 mask 255.255.255.0 nomodify notrap

* + 配置服务

在配置server的地方增加相关的Server配置。

* 配置时间更新到硬件BIOS时钟中。

*vim /etc/sysconfig/ntpd*

在后面添加内容：SYNC\_HWCLOCK=yes

所有配置完成之后，启动ntp服务：

*chkconfig ntpd on && service ntpd start*



重要提醒

NTP设置Server时不要使用：10.10.50.111（公司的时钟服务器，使用了之后发现不太正常）。

配置server时尽量使用网址或者hostname，不要使用IP地址。配置完成之后，可以通过以下两个命令来进行检查是否成功：

#ntpstat:查看上次同步是否成功

#ntpq –p：查看同步的详细信息

1. **Slavers中NTP相关设置**

* 配置服务

*vi /etc/ntp.conf*

在配置server的地方增加：server controller。

* 配置时间更新到硬件BIOS时钟中。

*vim /etc/sysconfig/ntpd*

在后面添加内容：SYNC\_HWCLOCK=yes

所有配置完成之后，启动ntp服务：

*chkconfig ntpd on && service ntpd start*



重要提醒

1.NTP和NTPdate

NTP是一种服务，可以定义的跟时间服务器进行时间比较，然后调整差值。

ntpdate是一个命令，可以一次性的跟时间服务器进行时间差值比较。

比如：在校验ＮＴＰ服务器设置中的server是否可用时，可以通过ntpdate命令来进行测试。

2.NTP服务器的日志位置在：/var/log/messages

3.有些教程描述让在server中加关键字：perfer，实验发现不能加，加上同步不成功。

### 4.3MySQL数据库设置

1. **控制节点安装步骤：**

* 运行以下命令，安装mysql服务器和客户端

*yum install -y mysql mysql-server MySQL-python*

* 启动MySQL服务

*service mysqld start*

*chkconfig mysqld on*

* 初始化MYSQL数据库

*mysql\_install\_db*

*mysql\_secure\_installation*



重要提醒

初始化MYSQL数据库的目的是为了删除mysql自带的测试数据库以及为mysql的root用户增加密码：实验环境的密码为：mysqlroot

1. **计算节点安装步骤：**

*yum install -y mysql MySQL-python*



重要提醒

报错： Host 'slaver1' is not allowed to connect to this MySQL server

以root用户登录到mysql数据库中，执行以下命令：

*Mysql>GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' IDENTIFIED BY "mysqlpass" WITH GRANT OPTION;  
 Mysql>flush privileges;*

### 4.4Openstack基础包下载设置

**在各个节点中，按照以下步骤安装：**

yum install <http://repos.fedorapeople.org/repos/openstack/openstack-havana/rdo-release-havana-6.noarch.rpm> -y

yum install <http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm> -y

yum install openstack-utils -y

yum install openstack-selinux -y

yum upgrade -y

reboot



重要提醒

由于网络原因，可能会提示下载失败等问题，这是由于需要远程国外的网站下载资源，如果一旦碰到下载失败的情况，重复执行命令即可，linux会自动连续下载。

Error: Cannot retrieve metalink for repository: epel. Please verify its path and try again

修改/etc/yum.repos.d/epel.repo文件，做如下改动

#baseurl=http://download.fedoraproject.org/pub/epel/6/$basearch

mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=$basearch

变为

baseurl=http://download.fedoraproject.org/pub/epel/6/$basearch

#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=$basearch

保存文件，再次执行即可

### 5.5消息服务器设置

只在**控制节点**进行消息服务器的安装工作：

1.安装消息服务器

*yum install qpid-cpp-server memcached -y*

2. 修改 /etc/qpidd.conf 文件，将auth的选项设置no，来禁用qpidd的认证。

auth=no

3.启动消息服务器,并设置为自动启动。

*service qpidd start*

*chkconfig qpidd on*

## 配置用户权限服务Idenitity Service

### 5.1用户权限服务的基本概念

用户权限服务主要包含两个功能：

* 用户管理：管理用户以及用户的权限。
* 服务目录：通过服务终端提供相应的服务目录。

主要涉及的名词有：

* 用户（user）：用户指的是使用openstack云服务的使用者，用户具有登录和使用资源的权限，用户隶属于租户。
* 证书(Credentials)：有且只有用户自己知道的一组数据，比如：密码，认证令牌等等
* 认证（**Authentication**）：确定一个用户身份的动作。
* 令牌（token）：组用来访问资源的随机数字，令牌中包含了访问资源的范围和权限等等。
* 租户(tenant)：租户可以被理解为一个项目，团队或组织。你必须指定一个相应的租户(tenant)才可以申请OpenStack服务
* 服务（Service）：openstack的组件表现形式，提供了一个或者多个终端，用户可以通过访问终端来控制资源。
* 终端（Endpoint）：通过网络访问服务的一种方式，通常是URL地址等。
* 角色（Role）：角色代表特定的租户中的用户操作权限

### 5.2安装和配置用户权限服务

在**控制节点**中完成以下操作：

1. 在控制节点中安装keystone服务包

yum install openstack-keystone python-keystoneclient -y

1. 用户权限服务通过数据库来存储基本信息。因此需要在数据库中修改数据库的位置，本实验中使用了mysql数据，通过以下命令来修改数据库配置信息。

openstack-config --set /etc/keystone/keystone.conf sql connection mysql://keystone:keystone@controller/keystone

1. 通过openstack-db 命令来初始化数据库表，同时创建相应的数据库和用户。

openstack-db --init --service keystone --password keystone

1. 使用openssl来生成认证令牌，用来使用用户权限服务和openstack的其他服务之间共享数据。

ADMIN\_TOKEN=$(openssl rand -hex 10)

echo $ADMIN\_TOKEN

openstack-config --set /etc/keystone/keystone.conf DEFAULT admin\_token $ADMIN\_TOKEN



重要提醒

这个步骤里生成ADMIN\_TOKEN，后面需要频繁的使用到，所以此时要记录下来。

1. KeyStone使用了PKI令牌，执行下面的名来安装关键字和证书。

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

chown -R keystone:keystone /etc/keystone/\* /var/log/keystone/keystone.log

1. 启动keystone的服务并设置为开机自启动。

service openstack-keystone start&&chkconfig openstack-keystone on

### 5.3定义用户，租户，角色等

安装完用户权限服务之后，可以增加相应的角色，租户，用户等。

1. 设置环境变量

export OS\_SERVICE\_TOKEN=使用上面生成的ADMIN\_TOKEN

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

1. 创建租户

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

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

1. 创建用户

keystone user-create --name=admin --pass=admin --email=admin@example.com

1. 创建角色

keystone role-create --name=admin

1. 将用户添加到对应的角色中去。

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



重要提醒

如果keystone服务报错：

The request you have made requires authentication. (HTTP 401)

检查：

1. 检查环境变量是否设置

1.检查AMDIN\_PASS是不是没有替换，如果没有替换

a.通过keystone user-password-update

  b.通过 openstack-db --drop --service keystone 删除数据库重建

### 5.4定义服务和端点

在openstack中注册服务和终端，注册之前确保前面定义的OS\_SERVICE\_TOKEN可用。

1. 注册服务

keystone service-create --name=keystone --type=identity --description="Keystone Identity Service"

执行完命令之后，记录下服务产生的ID。

1. 注册终端

keystone endpoint-create \

--service-id=上面流程中注册服务产生的服务的ID \

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

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

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

## 配置镜像服务

镜像服务就是openstack用来存储服务器镜像的一个服务。镜像的存储位置可以有几种配置方式，可以通过配置文件/etc/glance/glance-cache.conf配置文件来进行修改。

在控制节点执行以下命令：

1. 安装镜像服务。

yum install -y openstack-glance

1. 在配置文件中修改镜像服务所在的数据库信息。

openstack-config --set /etc/glance/glance-api.conf DEFAULT sql\_connection mysql://glance:glance@controller/glance

openstack-config --set /etc/glance/glance-registry.conf DEFAULT sql\_connection mysql://glance:glance@controller/glance

1. 通过openstack-db来初始化数据库，用户，表。

openstack-db --init --service glance --password glance

1. 创建glance用户，并添加到对用的角色上。

keystone user-create --name=glance --pass=glance --email=glance@example.com

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



重要提醒

运行命令keystone user-role-add --user=glance --tenant=service --role=admin后，通过keystone user-role-list查看不到glance的相关配置，不是配置错误，可以通过命令：keystone --os-tenant-name service --os-username glance --os-password glance user-role-list来查看

1. 在配置文件中做一些修改，来保证镜像服务和用户权限服务结合。

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 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

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 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

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

1. 在配置文件中增加认证相关配置。

首先复制配置文件到正确的位置

cp /usr/share/glance/glance-api-dist-paste.ini /etc/glance/glance-api-paste.ini

cp /usr/share/glance/glance-registry-dist-paste.ini /etc/glance/glance-registry-paste.ini

编辑这两个配置文件，将以下配置添加到选项：[filter:authtoken]中

vi /etc/glance/glance-api-paste.ini

vi /etc/glance/glance-registry-paste.ini

[filter:authtoken]

paste.filter\_factory=keystoneclient.middleware.auth\_token:filter\_factory

auth\_host=controller

admin\_user=glance

admin\_tenant\_name=service

admin\_password=glance

1. 注册服务

keystone service-create --name=glance --type=image --description="Glance Image Service"

1. 注册服务终端

keystone endpoint-create \

--service-id=上面注册服务时产生的服务ID\

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

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

--adminurl=http://controller:9292

1. 启动镜像服务病设置为自动启动

service openstack-glance-api start && chkconfig openstack-glance-api on

service openstack-glance-registry start && chkconfig openstack-glance-registry on

## 配置计算服务

计算服务是云算机控制器，它是IaaS系统的主要部分，用它来承载和管理云计算系统。

### 7.1安装计算服务

#### 7.1.1 控制节点

1. 安装计算包

yum install -y openstack-nova python-novaclient

1. 在配置文件中修改数据库存储信息

openstack-config --set /etc/nova/nova.conf database connection mysql://nova:nova@controller/nova

1. 在配置文件中配置消息服务器

openstack-config --set /etc/nova/nova.conf DEFAULT rpc\_backend nova.openstack.common.rpc.impl\_qpid

openstack-config --set /etc/nova/nova.conf DEFAULT qpid\_hostname controller

使用openstack-db命令来初始化数据库，用户，表

openstack-db --init --service nova --password nova

1. 在配置文件中配置vnc相关的配置

openstack-config --set /etc/nova/nova.conf DEFAULT my\_ip 10.10.70.21

openstack-config --set /etc/nova/nova.conf DEFAULT vncserver\_listen 10.10.70.21

openstack-config --set /etc/nova/nova.conf DEFAULT vncserver\_proxyclient\_address 10.10.70.21



重要提醒

**my\_ip是一个非常重要的参数，一定要配置成外网的IP。官方文档中使用了内网IP，是不对。一定注意。**

1. 创建用户和角色

keystone user-create --name=nova --pass=nova --email=nova@example.com

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

1. 修改配置信息，使配置文件和身份认证服务结合。

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

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_host controller

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_protocol http

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_port 35357

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

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

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

1. 在配置文件中增加证书,配置文件的[filter:authtoken]选项中增加配置段。

vi /etc/nova/api-paste.ini

[filter:authtoken]

paste.filter\_factory = keystoneclient.middleware.auth\_token:filter\_factory

auth\_host = controller

auth\_port = 35357

auth\_protocol = http

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

admin\_tenant\_name = service

admin\_user = nova

admin\_password = nova

1. 创建服务

keystone service-create --name=nova --type=compute --description="Nova Compute service"

1. 创建服务终端

keystone endpoint-create \

--service-id=上面注册服务时产生的服务ID \

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

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

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

1. 启动服务以及服务设置为自动启动。

service openstack-nova-api start

service openstack-nova-cert start

service openstack-nova-consoleauth start

service openstack-nova-scheduler start

service openstack-nova-conductor start

service openstack-nova-novncproxy start

chkconfig openstack-nova-api on

chkconfig openstack-nova-cert on

chkconfig openstack-nova-consoleauth on

chkconfig openstack-nova-scheduler on

chkconfig openstack-nova-conductor on

chkconfig openstack-nova-novncproxy on

#### 7.1.2结算节点

在各个不同的计算节点中运行以下命令

1. 安装计算包

yum install openstack-nova-compute

1. 修改配置文件跟用户权限服务结合。

openstack-config --set /etc/nova/nova.conf database connection mysql://nova:nova@controller/nova

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

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_host controller

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_protocol http

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_port 35357

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

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

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

1. 在配置文件中配置消息服务器

openstack-config --set /etc/nova/nova.conf DEFAULT rpc\_backend nova.openstack.common.rpc.impl\_qpid

openstack-config --set /etc/nova/nova.conf DEFAULT qpid\_hostname controller

1. 在配置文件中配置VNC服务器相关设置【每个服务器不同】

openstack-config --set /etc/nova/nova.conf DEFAULT my\_ip 10.10.70.22

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 10.10.70.22

openstack-config --set /etc/nova/nova.conf DEFAULT novncproxy\_base\_url [http://controller:6080/vnc\_auto.html](http://master:6080/vnc_auto.html)



重要提醒

**my\_ip是一个非常重要的参数，一定要配置成外网的IP。官方文档中使用了内网IP，是不对。一定注意。**

1. 在配置文件中配置镜像的相关服务

openstack-config --set /etc/nova/nova.conf DEFAULT glance\_host controller

1. 在配置文件中增加认证相关配置

vi /etc/nova/api-paste.ini

[filter:authtoken]

paste.filter\_factory = keystoneclient.middleware.auth\_token:filter\_factory

auth\_host = controller

auth\_port = 35357

auth\_protocol = http

admin\_tenant\_name = service

admin\_user = nova

admin\_password = nova

1. 启动服务并设置自动启动

service libvirtd start

service messagebus start

chkconfig libvirtd on

chkconfig messagebus on

service openstack-nova-compute start

chkconfig openstack-nova-compute on

### 7.2网络设置

在**所有的计算节点**中运行以下命令：

1. 安装网络相关包

yum install -y openstack-nova-network

1. 在各个计算节点的nova.conf文件中加入网络模块

openstack-config --set /etc/nova/nova.conf DEFAULT network\_manager nova.network.manager.FlatDHCPManager

openstack-config --set /etc/nova/nova.conf DEFAULT firewall\_driver nova.virt.libvirt.firewall.IptablesFirewallDriver

openstack-config --set /etc/nova/nova.conf DEFAULT network\_size 254

openstack-config --set /etc/nova/nova.conf DEFAULT allow\_same\_net\_traffic False

openstack-config --set /etc/nova/nova.conf DEFAULT multi\_host True

openstack-config --set /etc/nova/nova.conf DEFAULT send\_arp\_for\_ha True

openstack-config --set /etc/nova/nova.conf DEFAULT share\_dhcp\_address True

openstack-config --set /etc/nova/nova.conf DEFAULT force\_dhcp\_release True

penstack-config --set /etc/nova/nova.conf DEFAULT flat\_interface eth0

openstack-config --set /etc/nova/nova.conf DEFAULT flat\_network\_bridge br100

openstack-config --set /etc/nova/nova.conf DEFAULT public\_interface eth1



重要提醒

**多节点双网卡的设置一定要注意flatc\_interface设置成eth0（不要设置成eth1）,否则会造成虚拟机无法上网的问题。**

官网的文档有问题。

1. 安装api服务，使计算节点能够访问虚拟实例。

yum install -y openstack-nova-api

service openstack-nova-metadata-api start

chkconfig openstack-nova-metadata-api on

1. 启动服务并注册自动启动

service openstack-nova-network start

chkconfig openstack-nova-network on

在**控制节点**运行命令：

nova network-create vmnet --fixed-range-v4=10.0.0.0/24 --bridge=br100 --multi-host=T



重要提醒

1. 创建网络的命令必须在控制节点上运行。
2. 如果网络后，虚拟机无法上网，不需要考虑IPv6的问题，经过测试没有影响。
3. 有资料说，可能跟网卡的混杂模式有关，经过测试没有发现关系。



重要提醒

1. allow\_same\_net\_traffic参数资料上说，可以改变虚拟机之间的访问关系，但是经过测试，发现没有关系，同一节点之间的虚拟机可以互相访问，不同节点之间的虚拟机不能互相访问。
2. 虚拟机创建后，只能通过虚拟机所在的计算节点（或其他计算节点）进行访问，控制节点无法访问。控制节点如果访问，需要配置浮动IP来设置。
3. 可以通过制定: --availability-zone nova:slaver来制定在那个节点上创建虚拟机。
4. 如果希望不通过密码进行登录虚拟机，要在不同的机器上上传不同的key。

## 配置控制面板

在控制节点上运行以下步骤：

1. 安装控制面版的服务。

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

1. 修改基础的配置。

Vi /etc/openstack-dashboard/local\_settings

1. 修改[default][Location][CACHES]选项：

CACHES = {

'default': {

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

'LOCATION' : '127.0.0.1:11211'

}

}

1. 修改[ALLOWED\_HOSTS]选项

ALLOWED\_HOSTS = ['\*']

1. 修改[OPENSTACK\_HOST]选项：

OPENSTACK\_HOST = "controller"

OPENSTACK\_KEYSTONE\_DEFAULT\_ROLE = "\_member\_"

1. 设置自动启动

service httpd start

service memcached start

chkconfig httpd on

chkconfig memcached on

## 设置浮动IP

通过以上步骤创建的虚拟机只能允许在计算节点访问或者通过控制台访问，如果需要让局域网内的所有机器访问（也就是70段地址），需要增加一个浮动IP的概念。也就是配一个外网地址到虚拟机的实例中。

可以在控制节点运行以下命令：

1. 首先通过命令创建浮动IP池

nova-manage floating create 10.10.70.20/30

1. 查看可用的浮动IP资源

nova-manage floating list

1. 从浮动IP池中获取可用的浮动IP

nova floating-ip-create

1. 将浮动IP设置到某个实例中。

nova floating-ip-associate c8e37f03-98c2-40fd-ad4e-47827c5b9098 10.10.70.65

## 配置硬盘服务

没有特殊说明的情况下，下面所有的命令在控制节点中运行：

1. 安装云硬盘服务包

yum install openstack-cinder openstack-utils openstack-selinux

1. 在配置文件中加入数据库配置信息

openstack-config --set /etc/cinder/cinder.conf database connection mysql://cinder:cinder@controller/cinder

1. 使用openstack-db初始化数据库，用户，表

openstack-db --init --service cinder --password cinder

1. 创建cinder用户

keystone user-create --name=cinder --pass=cinder --email=cinder@example.com

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

1. 在配置文件中增加认证信息

vi /etc/cinder/api-paste.ini

[filter:authtoken]

paste.filter\_factory=keystoneclient.middleware.auth\_token:filter\_factory

auth\_host=controller

auth\_port = 35357

auth\_protocol = http

auth\_uri = http://controller:5000

admin\_tenant\_name=service

admin\_user=cinder

admin\_password=cinder

1. 在配置文件中增加消息服务器地址

openstack-config --set /etc/cinder/cinder.conf DEFAULT rpc\_backend cinder.openstack.common.rpc.impl\_qpid

openstack-config --set /etc/cinder/cinder.conf DEFAULT qpid\_hostname controller

1. 注册服务和终端

keystone service-create --name=cinder --type=volume --description="Cinder Volume Service"

keystone endpoint-create \

--service-id=注册上面的服务时产生的服务ID\

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

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

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

1. 注册第二版本的服务和终端

keystone service-create --name=cinderv2 --type=volumev2 --description="Cinder Volume Service V2"

keystone endpoint-create \

--service-id=注册上面的服务时产生的服务ID \

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

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

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

1. 启动服务和设置自启动

service openstack-cinder-api start

service openstack-cinder-scheduler start

chkconfig openstack-cinder-api on

chkconfig openstack-cinder-scheduler on