# （一）Nova子服务节点分配

计算服务nova较之前的服务稍显复杂（但没有网络服务neutron复杂），它需要在控制节点和计算节点都安装。

控制节点主要安装nova-api（nova主服务）、nova-scheduler（nova调度服务）、nova-conductor（nova数据库服务，提供数据库访问）、nova-novncproxy（nova的vnc服务，提供实例的控制台）等服务。

计算节点主要安装nova-compute（nova计算服务）。

# （二）Nova子服务Controller节点部署

## 1. 创建数据库并授权

创建三个数据库nova, nova\_api, nova\_cell0，并授予nova用户对三个数据库的本地登录以及远程登录权限，并设定密码为NOVA\_DBPASS：

MariaDB [(none)]> CREATE DATABASE nova\_api;

Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> CREATE DATABASE nova;

Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> CREATE DATABASE nova\_cell0;

Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON nova\_api.\* TO 'nova'@'localhost' IDENTIFIED BY 'NOVA\_DBPASS';

Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON nova\_api.\* TO 'nova'@'%' IDENTIFIED BY 'NOVA\_DBPASS';

Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON nova.\* TO 'nova'@'localhost' IDENTIFIED BY 'NOVA\_DBPASS';

Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON nova.\* TO 'nova'@'%' IDENTIFIED BY 'NOVA\_DBPASS';

Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON nova\_cell0.\* TO 'nova'@'localhost' IDENTIFIED BY 'NOVA\_DBPASS';

Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON nova\_cell0.\* TO 'nova'@'%' IDENTIFIED BY 'NOVA\_DBPASS';

Query OK, 0 rows affected (0.001 sec)

## 2. 创建nova用户

创建OpenStack的用户nova，密码为NOVA\_PASS：

[root@controller ~]# openstack user create --domain default --password NOVA\_PASS nova

## 3. 向nova用户添加admin角色

为nova用户添加admin角色，并且加入项目service中：

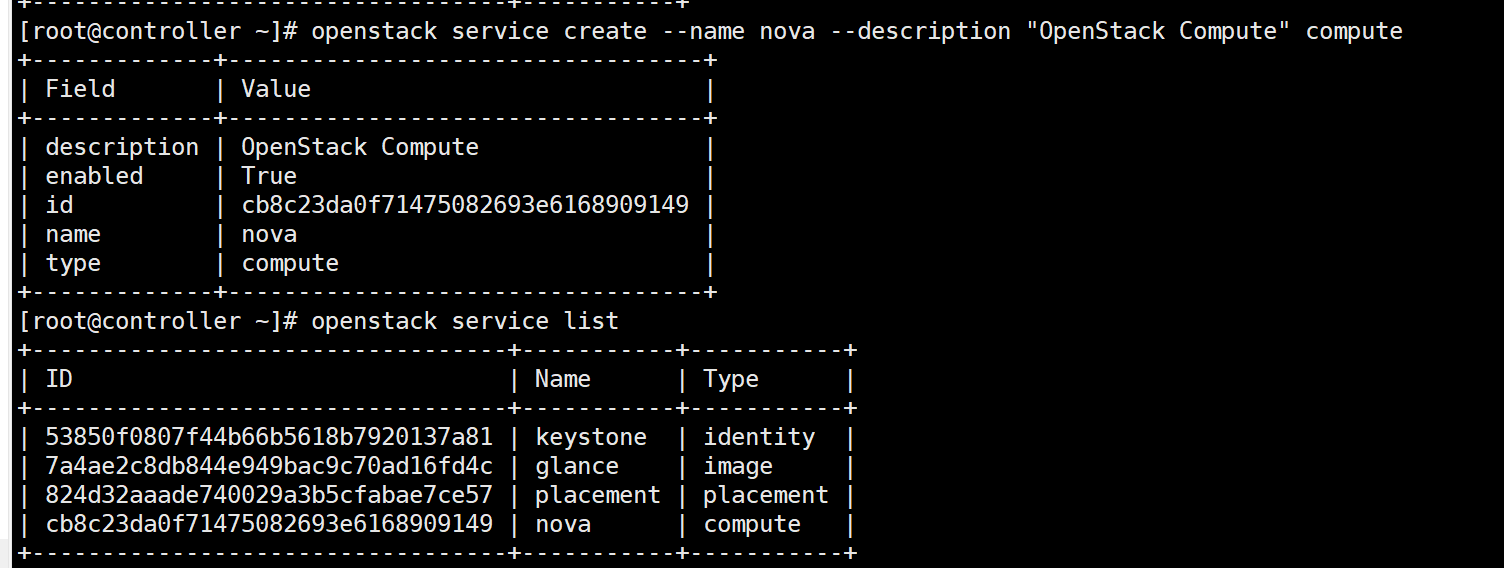
[root@controller ~]# openstack role add --project service --user nova admin

## 4. 创建nova服务实体

创建nova服务service，类型是compute：

[root@controller ~]# openstack service create --name nova --description "OpenStack Compute" compute

查看当前的service列表：



## 5. 给Nova服务关联endpoint（端点）

创建三个endpoint端点，与之前一样：

[root@controller ~]# openstack endpoint create --region RegionOne compute public http://192.168.112.146:8774/v2.1

[root@controller ~]# openstack endpoint create --region RegionOne compute internal http://192.168.112.146:8774/v2.1

[root@controller ~]# openstack endpoint create --region RegionOne compute admin http://192.168.112.146:8774/v2.1

## 6.Controller节点需要安装的nova子服务

nova-api, nova-conductor, nova-novncproxy, nova-scheduler四个组件：

[root@controller ~]# yum install openstack-nova-api openstack-nova-conductor openstack-nova-novncproxy openstack-nova-scheduler -y

## 7. 修改nova.conf配置文件

精简nova.conf配置文件：

cp -a /etc/nova/nova.conf{,.bak}

grep -Ev '^$|#' /etc/nova/nova.conf.bak > /etc/nova/nova.conf

指定支持的API类型

openstack-config --set /etc/nova/nova.conf DEFAULT enabled\_apis osapi\_compute,metadata

定义Controller节点的IP

openstack-config --set /etc/nova/nova.conf DEFAULT my\_ip 192.168.112.146

通过neutron获取IP地址

openstack-config --set /etc/nova/nova.conf DEFAULT use\_neutron true

防火墙驱动

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

指定连接的rabbitmq

openstack-config --set /etc/nova/nova.conf DEFAULT transport\_url rabbit://openstack:RABBIT\_PASS@192.168.112.146

nova\_api数据库的mysql连接

openstack-config --set /etc/nova/nova.conf api\_database connection mysql+pymysql://nova:NOVA\_DBPASS@192.168.112.146/nova\_api

nova数据库的mysql连接

openstack-config --set /etc/nova/nova.conf database connection mysql+pymysql://nova:NOVA\_DBPASS@192.168.112.146/nova

placement数据库的mysql连接

openstack-config --set /etc/nova/nova.conf placement\_database connection mysql+pymysql://placement:PLACEMENT\_DBPASS@192.168.112.146/placement

nova的认证策略为keystone

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

keystone的认证url

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_url http://192.168.112.146:5000/v3

memcached的服务地址

openstack-config --set /etc/nova/nova.conf keystone\_authtoken memcached\_servers 192.168.112.146:11211

keystone认证方式为password

openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_type password

keystone的项目Domain为Default

openstack-config --set /etc/nova/nova.conf keystone\_authtoken project\_domain\_name Default

用户的Damain为Default

openstack-config --set /etc/nova/nova.conf keystone\_authtoken user\_domain\_name Default

项目名为service

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

keystone的登录用户名为nova

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

nova用户在keystone的登录密码为NOVA\_PASS

openstack-config --set /etc/nova/nova.conf keystone\_authtoken password NOVA\_PASS

vnc如果配置不正确，则连接不上虚拟机的控制台

openstack-config --set /etc/nova/nova.conf vnc enabled true

vnc监听地址

openstack-config --set /etc/nova/nova.conf vnc server\_listen ' $my\_ip'

server的客户端地址为本机地址；此地址是管理网的地址

openstack-config --set /etc/nova/nova.conf vnc server\_proxyclient\_address ' $my\_ip'

glance的地址

openstack-config --set /etc/nova/nova.conf glance api\_servers <http://192.168.112.146:9292>

指定锁路径, 锁的作用是创建虚拟机时，在执行某个操作的时候，需要等此步骤执行完后才能执行下一个步骤，不能并行执行，保证操作是一步一步的执行

openstack-config --set /etc/nova/nova.conf oslo\_concurrency lock\_path /var/lib/nova/tmp

placement服务的region名称

openstack-config --set /etc/nova/nova.conf placement region\_name RegionOne

placement服务的项目所属Domain为Default

openstack-config --set /etc/nova/nova.conf placement project\_domain\_name Default

placement服务的项目名为service

openstack-config --set /etc/nova/nova.conf placement project\_name service

placement服务的认证方式为password

openstack-config --set /etc/nova/nova.conf placement auth\_type password

placement服务的用户所属Domain为Default

openstack-config --set /etc/nova/nova.conf placement user\_domain\_name Default

placement服务的认证url

openstack-config --set /etc/nova/nova.conf placement auth\_url http://192.168.112.146:5000/v3

placement服务的用户名为placement

openstack-config --set /etc/nova/nova.conf placement username placement

placement服务的密码为PLACEMENT\_PASS

openstack-config --set /etc/nova/nova.conf placement password PLACEMENT\_PASS

## 8. 填充nova-api数据库

初始化nova\_api数据库

[root@controller ~]# su -s /bin/sh -c "nova-manage api\_db sync" nova

## 9.注册cell0数据库

nova服务内部把资源划分到不同的cell中，把计算节点划分到不同的cell中；openstack内部基于cell把计算节点进行逻辑上的分组

[root@controller ~]# su -s /bin/sh -c "nova-manage cell\_v2 map\_cell0" nova

创建cell1单元格

[root@controller ~]# su -s /bin/sh -c "nova-manage cell\_v2 create\_cell --name=cell1 --verbose" nova

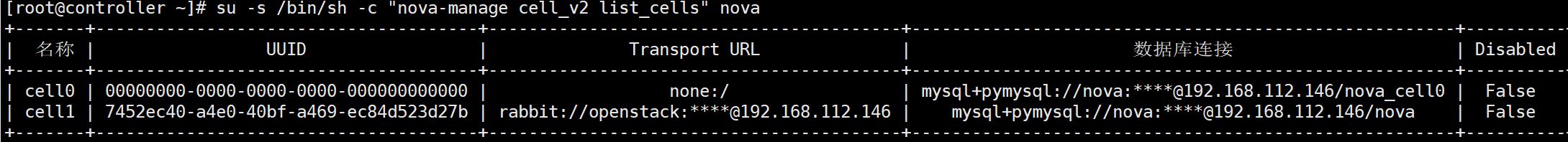
初始化nova数据库，可以通过 /var/log/nova/nova-manage.log 日志判断是否初始化成功

[root@controller ~]# su -s /bin/sh -c "nova-manage db sync" nova

可使用以下命令验证cell0和cell1是否注册成功

su -s /bin/sh -c "nova-manage cell\_v2 list\_cells" nova

验证cell0和cell1组件是否注册成功



## 10. 启动计算服务nova并将其配置为开机自启

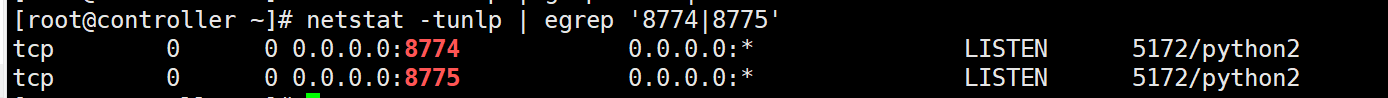
将Controller节点的4个nova子服务设置开机自启动

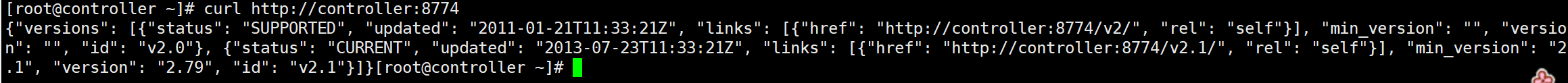
[root@controller ~]# systemctl enable openstack-nova-api.service openstack-nova-scheduler.service openstack-nova-conductor.service openstack-nova-novncproxy.service

[root@controller ~]# systemctl start openstack-nova-api.service openstack-nova-scheduler.service openstack-nova-conductor.service openstack-nova-novncproxy.service

## 11.检测nova服务启动情况

同样，使用netstat -tnlup查看端口情况，如出现8774和8775端口则表示nova服务正常启动。



再通过访问Controller节点的8774端口，返回信息：  


# （三）Nova子服务Compute节点部署

## 1.安装OpenStack Train版本源

[root@compute01 ~]# yum install centos-release-openstack-train -y

## 2.安装nova-compute软件包

[root@compute01 ~]# yum install openstack-nova-compute -y

## 3.修改计算节点的nova配置文件

精简nova.conf配置文件

[root@compute01 ~]# cp -a /etc/nova/nova.conf{,.bak}

[root@compute01 ~]# grep -Ev '^$|#' /etc/nova/nova.conf.bak > /etc/nova/nova.conf

安装openstack-utils工具

[root@controller ~]# yum install -y openstack-utils

指定支持的API类型

openstack-config --set /etc/nova/nova.conf DEFAULT enabled\_apis osapi\_compute,metadata

定义Compute节点的IP

openstack-config --set /etc/nova/nova.conf DEFAULT my\_ip 192.168.112.145

通过neutron获取IP地址

openstack-config --set /etc/nova/nova.conf DEFAULT use\_neutron true

防火墙驱动

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

指定连接的rabbitmq，在Controller节点中

openstack-config --set /etc/nova/nova.conf DEFAULT transport\_url rabbit://openstack:RABBIT\_PASS@192.168.112.146

认证策略为keystone

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf api auth\_strategy keystone

keystone认证URL（Controller中）

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_url <http://192.168.112.146:5000/v3>

memcached服务地址（Controller中）

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf keystone\_authtoken memcached\_servers 192.168.112.146:11211

keystone的认证方式为password

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf keystone\_authtoken auth\_type password

keystone的项目Damain名为Default

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf keystone\_authtoken project\_domain\_name Default

keystone的用户Damain名为Default

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf keystone\_authtoken user\_domain\_name Default

keystone的project为service

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf keystone\_authtoken project\_name service

keystone的用户名为nova

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf keystone\_authtoken username nova

keystone的认证密码为NOVA\_PASS

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf keystone\_authtoken password NOVA\_PASS

vnc连接控制台

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf vnc enabled true

vnc服务监听IP

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf vnc server\_listen 0.0.0.0

vnc代理客户端地址为本机地址

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf vnc server\_proxyclient\_address ' $my\_ip'

vnc代理基本url（Controller中）

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf vnc novncproxy\_base\_url <http://192.168.112.146:6080/vnc_auto.html>

glance的api

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf glance api\_servers <http://192.168.112.146:9292>

线程锁位置：

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf oslo\_concurrency lock\_path /var/lib/nova/tmp

placement服务的Region名称

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf placement region\_name RegionOne

placement服务的project所属的Domain为Default

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf placement project\_domain\_name Default

placement服务的project名为service

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf placement project\_name service

placement服务的认证方式为password

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf placement auth\_type password

placement服务的用户Domain为Default

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf placement user\_domain\_name Default

placement服务的认证地址url

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf placement auth\_url <http://192.168.112.146:5000/v3>

placement服务的用户名为placement

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf placement username placement

placement服务的密码

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf placement password PLACEMENT\_PASS

compute子服务的libvirt类型

[root@compute01 ~]# openstack-config --set /etc/nova/nova.conf libvirt virt\_type qemu

## 4. 确定计算节点是否支持虚拟机硬件加速

检查当前计算节点：

[root@compute01 ~]# egrep -c '(vmx|svm)' /proc/cpuinfo



如果此命令返回值不是0，则计算节点支持硬件加速，不需要加入下面的配置。

如果此命令返回值是0，则计算节点不支持硬件加速，并且必须配置libvirt为使用QEMU而不是KVM，需要编辑/etc/nova/nova.conf 文件中的[libvirt]部分： [libvirt] virt\_type = qemu

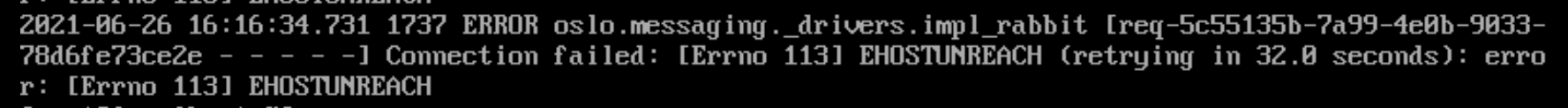
## 5.启动Compute服务及其相关服务，并设置开机自启

[root@compute01 ~]# systemctl enable libvirtd.service openstack-nova-compute.service

[root@compute01 ~]# systemctl start libvirtd.service openstack-nova-compute.service

执行命令后，openstack-nova-compute.service服务一直启动不起来，到/var/

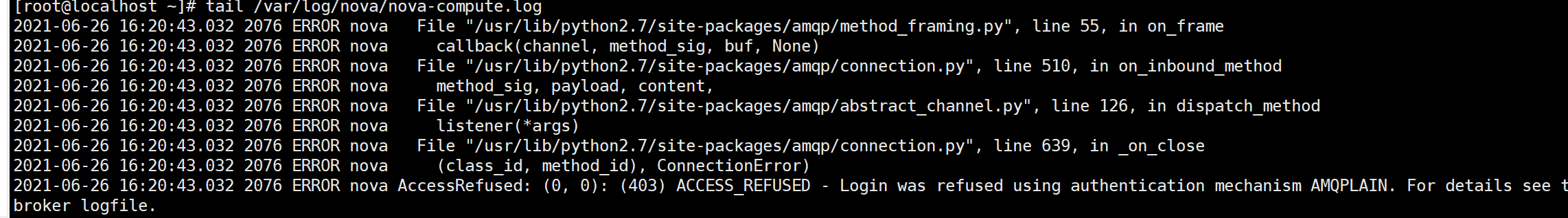
log/nova/nova-compute.log日志中查看启动失败情况，结果如下：



可以看到，是rabbitmq的连接问题，此时我并没有开启Controller节点的rabbitmq，因此尝试开启Controller节点。

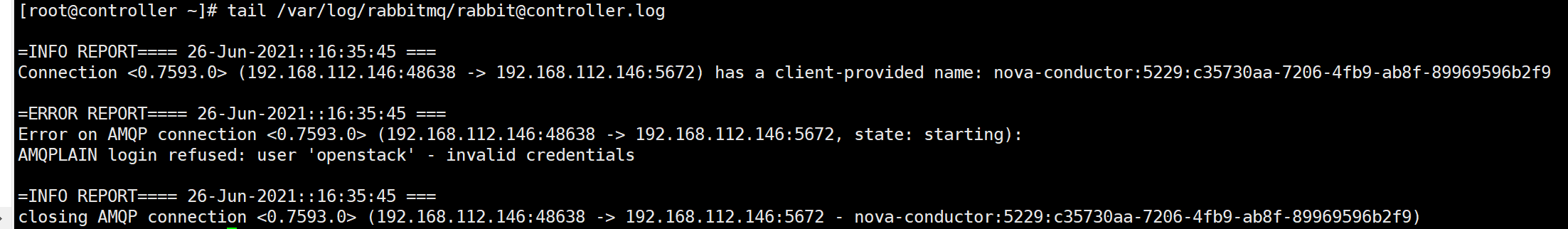
可是openstack-nova-compute.service服务依然启动失败，查看compute节点的nova-compute.log日志可以看到如下信息：

tail /var/log/nova/nova-compute.log



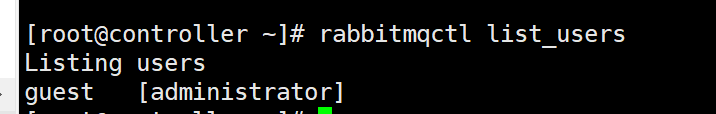
从最后可以看到，是AccessRefused导致的，即AMQPLAIN拒绝登录，说明是登录rabbitmq失败，因此到Controller节点中的查看rabbitmq的日志：

tail /var/log/rabbitmq/rabbitt@controller.log



从中间的一行可以看到，是openstack用户没有有效的认证，下面查看rabbitmq中的用户信息：

rabbitmqctl list\_users



可以看到，只有一个guest用户，权限是admin，因此需要添加一个openstack用户，设置密码为RABBIT\_PASS，这个密码是我们在Compute节点的nova.conf中配置的密码：

rabbitmqctl add\_user openstack RABBIT\_PASS

增加配置、读取及写入相关权限

rabbitmqctl set\_permissions openstack ".\*" ".\*" ".\*"

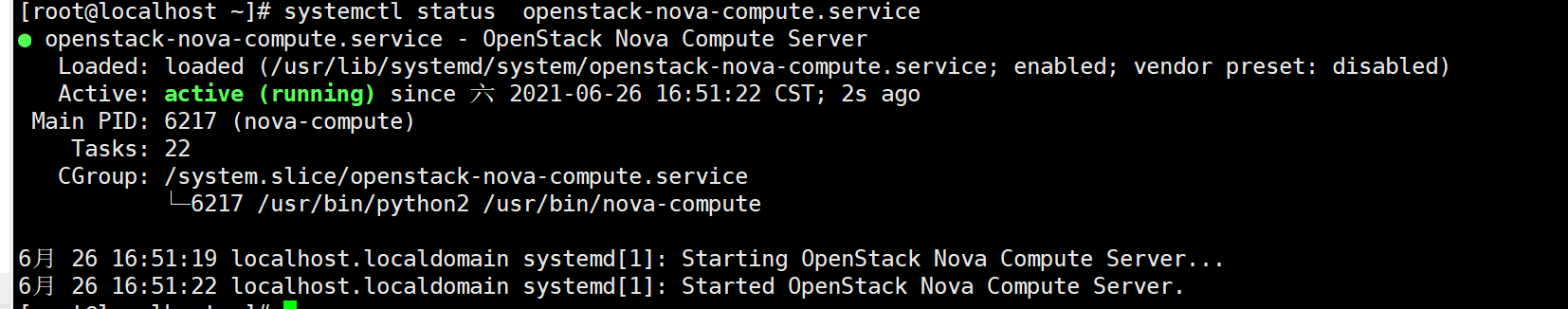
重启Controller节点中的rabbitmq服务，

systemctl restart rabbitmq-server.service

重启Compute节点中的openstack-nova-compute服务

systemctl restart openstack-nova-compute.service

最终，查看openstack-nova-compute服务启动成功，如下所示：



## 6.修改Compute节点主机名配置

修改主机名，方便操作

hostnamectl set-hostname computer

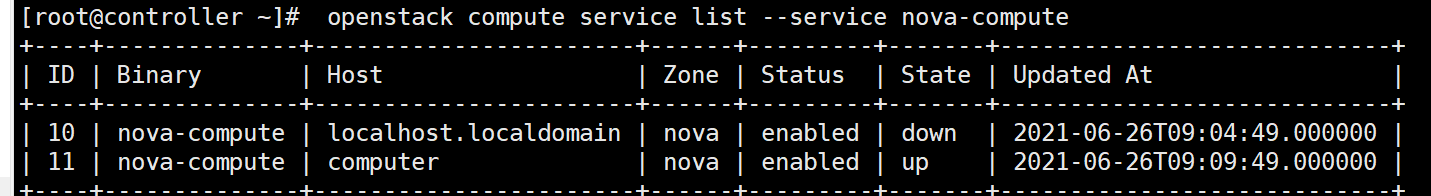
配置hosts

192.168.112.145 compute

## 7.验证（controller节点操作）

查看compute节点是否注册到controller上，通过消息队列；需要在controller节点执行以下的指令：

[root@ct ~]# openstack compute service list --service nova-compute



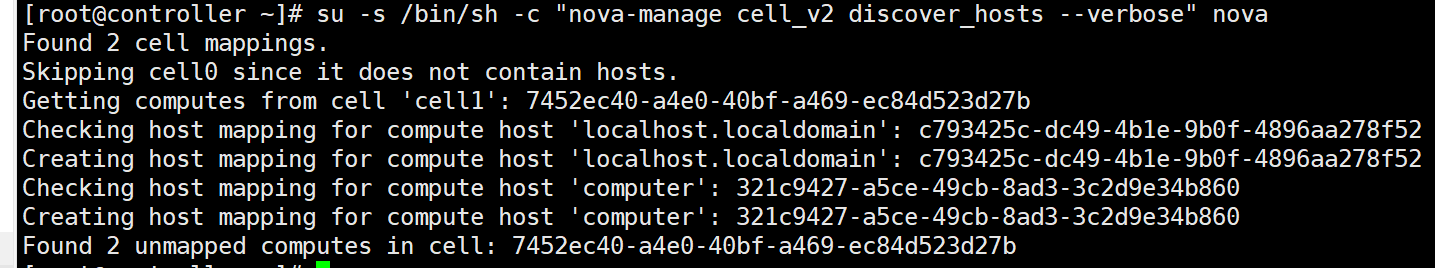
这里有两个是因为我在之前没有设置Compute节点的hostname，修改完hostname之后的状态为UP。

## 8.发现计算节点主机（Controller节点操作）

扫描当前openstack中有哪些计算节点可用，发现后会把计算节点创建到cell中，后面就可以在cell中创建虚拟机；相当于openstack内部对计算节点进行分组，把计算节点分配到不同的cell中

[root@ct ~]# su -s /bin/sh -c "nova-manage cell\_v2 discover\_hosts --verbose" nova

可以看到，发现两个cell，一个是cell1，其中不包含host注解；另一个是cell，其中包含了两个host，就是我们之前在未设置hostname和设置之后的两个host：



由于在以后添加新的计算节点时，必须在控制器节点上运行”su -s /bin/sh -c "nova-manage cell\_v2 discover\_hosts --verbose" nova“以注册这些新的计算节点。这样很麻烦，因此可以修改Controller节点的nova.conf文件：

[root@ct ~]# vim /etc/nova/nova.conf

[scheduler]

discover\_hosts\_in\_cells\_interval = 300 #每300秒扫描一次

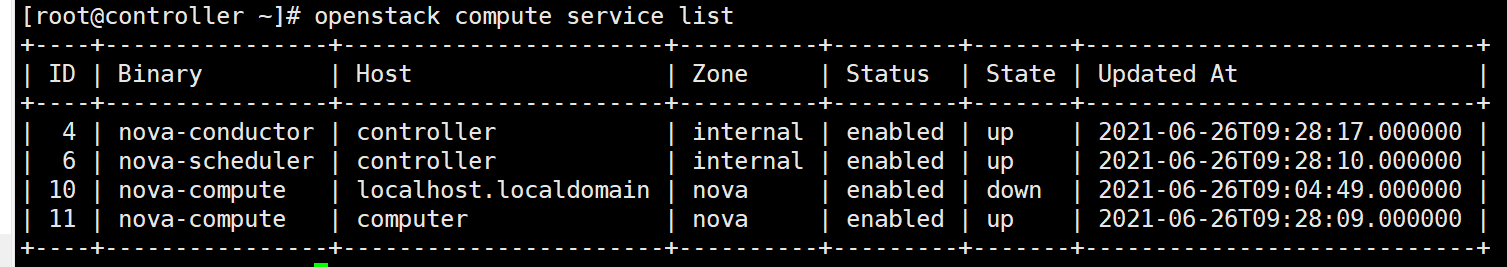
设置扫描时间为300秒扫描一次。

重启Controller节点的nova-api.service服务：  
systemctl restart openstack-nova-api.service

## 9.验证计算节点服务（Controller节点操作）

检查 nova 的各个服务是否都是正常，以及 compute 服务是否注册成功

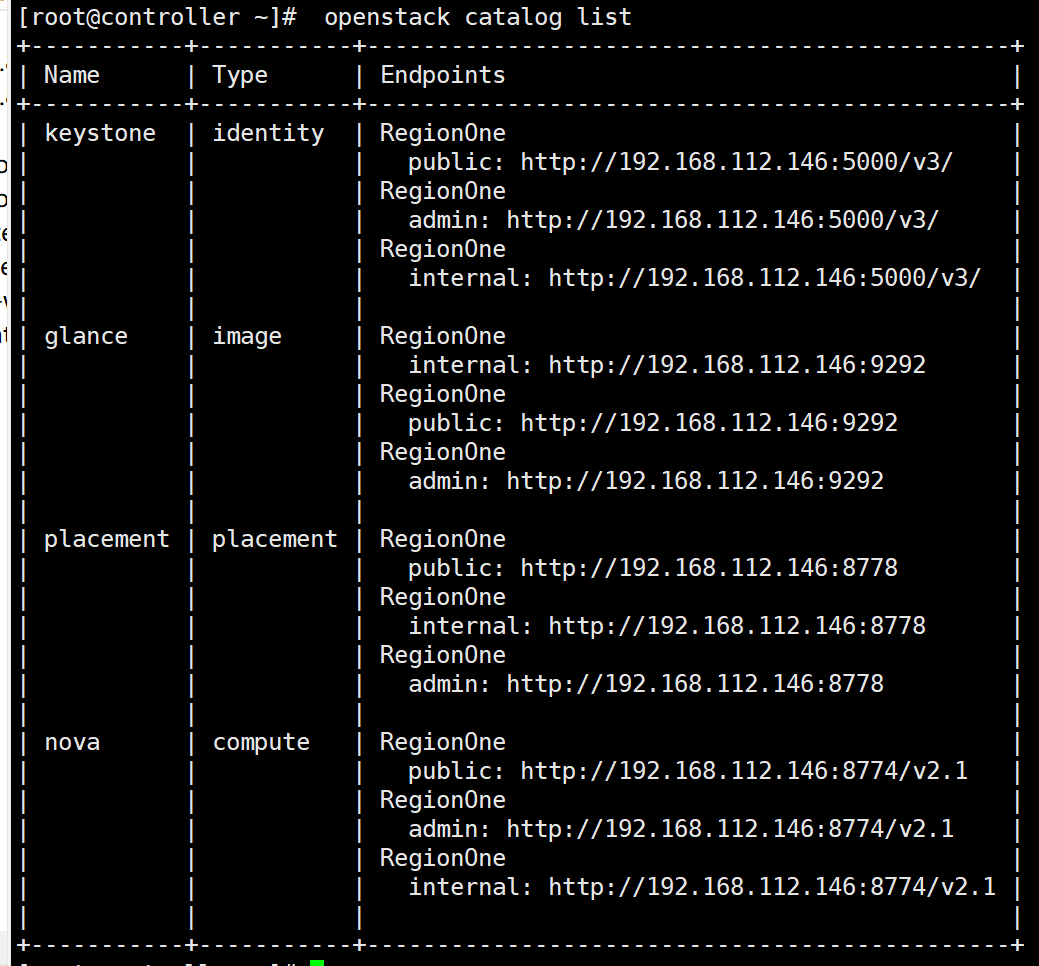
[root@ct ~]# openstack compute service list



可以看到，controller节点中启动了两个服务，compute节点启动了一个服务。

查看OpenStack各个组件的 访问api 是否正常

[root@ct ~]# openstack catalog list



查看cell的api和placement的api是否正常，只要其中一个有误，后期无法创建虚拟机

[root@ct ~]# nova-status upgrade check

