OpenStack R版部署及VPC配置详解！

### 秦伟 云技术之家 4天前

我们知道OpenStack是云操作平台和工具集，其目的在于为组织提供可扩展的、灵活的计算、存储、网络服务。

在传统意义上的操作系统主要包含： 处理器管理， 作业管理， 存储器管理， 设备管理和文件管理等。

OpenStack云操作系统包含以下组件：

Nova - 计算服务Neutron-网络服务Swift - 对象存储服务Cinder-块存储服务Glance - 镜像服务Keystone - 认证服务Horizon - UI服务Ceilometer-监控服务Heat-集群服务

Trove-数据库服务

...

可以看出OpenStack包含了很多的组件，正因为如此，它的安装对于一般没接触过的人来说，不仅要了解每个组件的安装过程，最重要的是要理解组件之间的联系，在各个组件的配置文件中将它们之间的联系建立起来，这无疑是非常复杂的。而且OpenStack项目更新周期很快，平均每半年发布一次新的版本，作为一名想要一探OpenStack究竟的体验者来说，可以采取另外一种方式：即利用packstack进行自动化安装， 采取allinone的方式，将openstack的主要服务安装在一个节点上，如果一切顺利的话，在一个小时以内就可以安装完成，进行OpenStack的初体验。

一、安装过程



本次宿主机（ 即安装OpenStack 的机器） 的操作系统是CentOS 7.5 。安装的OpenStack 是目前最新的rocky版本，官方文档建议机器至少有16 GB的内存，处理器硬件虚拟化扩展,和至少一个网络适配器。当然

8G内存也可安装，如果用虚机，需要给虚机分配至少6G以上内存，否则速度很慢。

1.改造之前首先确保环境可以上外网。

可以试着ping [www.baidu.com来测试一下](http://www.baidu.com来测试一下/)，通了的话可以上外网。

2.开始安装Openstack之前先关闭NetworkManager并启动network服务。

$ sudo systemctl disable firewalld # 可能没有firewalld，不过没有的话不影响接下来操作

## $ sudo systemctl stop firewalld

$ sudo systemctl disable NetworkManager

$ sudo systemctl stop NetworkManager

$ sudo systemctl enable network

$ sudo systemctl start network

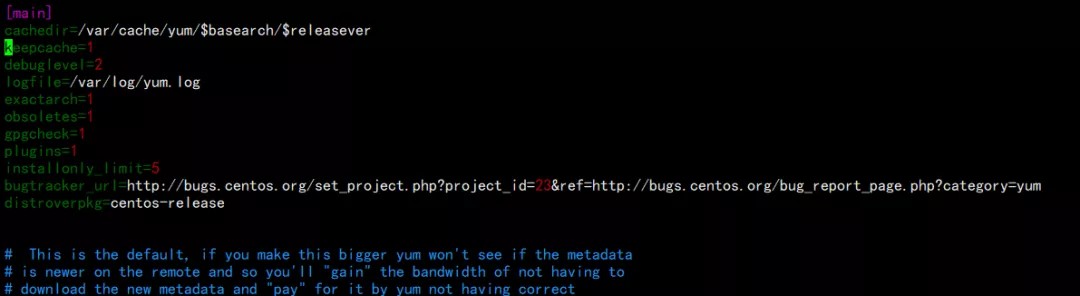
同时禁selinux，

## vim /etc/selinux/config，将SELINUX设置为disabled。

3. 编辑yum配置

## vim /etc/yum.conf

修改后可以下载rpm包。

将keepcache的值由原来的0（表示安装后删除软件包）修改为1（表示安装后保留软件包）

4.配置OpenStack yum源：

## On RHEL, download and install the RDO repository RPM to set up the OpenStack repository: sudo yum install -y https://rdoproject.org/repos/rdo-release.rpm

On CentOS, the Extras repository provides the RPM that enables the OpenStack repository. Extras is enabled by default on CentOS 7, so you can simply install the RPM to set up the OpenStack repository.

$ sudo yum install -y centos-release-openstack-rocky

之后，在/etc/yum.repos.d目录下会生成rdo-release.repo文件。编辑这个文件

vi rdo-release.repo，设置如下：gpkcheck=0 {表示对从这个源下载的rpm包不进行校验；

enable=1 表示启用这个源

5. 升级系统

## yum update -y

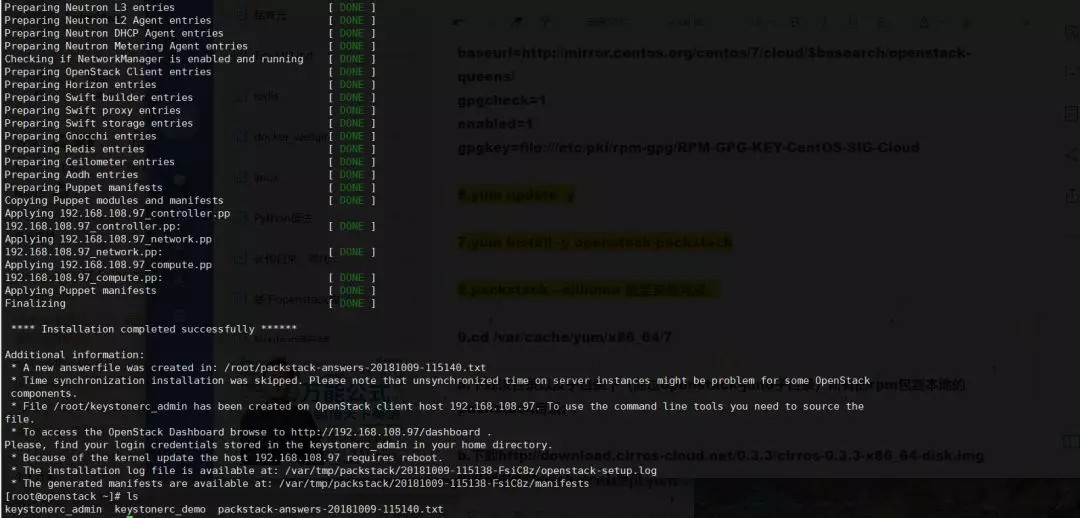
6.安装OpenStack packstack

## yum install -y openstack-packstack

7.通过allinone模式按照OpenStack

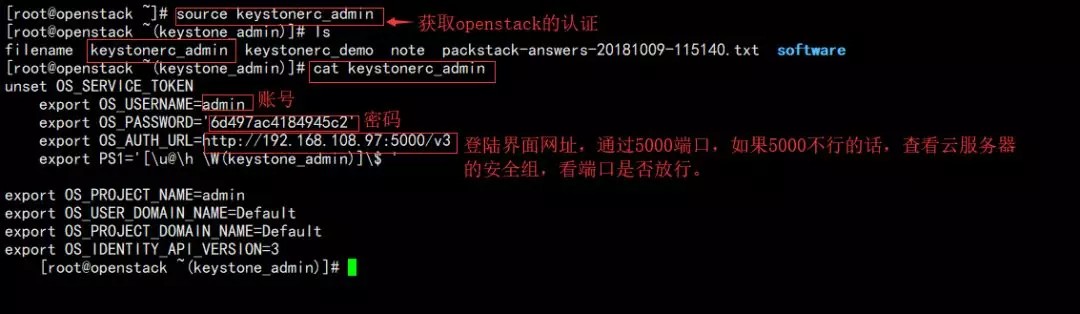
## packstack --allinone

直至安装完成。

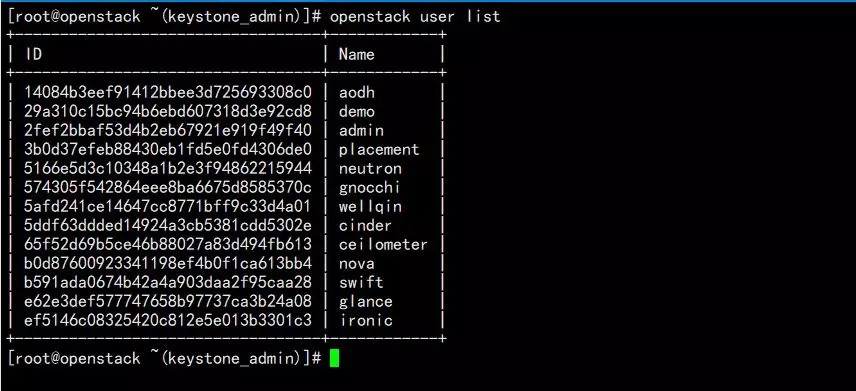
安装完成后如图所示：

接下来就可以进入dashboard界面（openstack的图形化操作界面）进行操作： 首先在命令行中：source一下认证服务

注意认证后，在主机名旁边出现（keystone\_admin）,说明已经成功认证了。

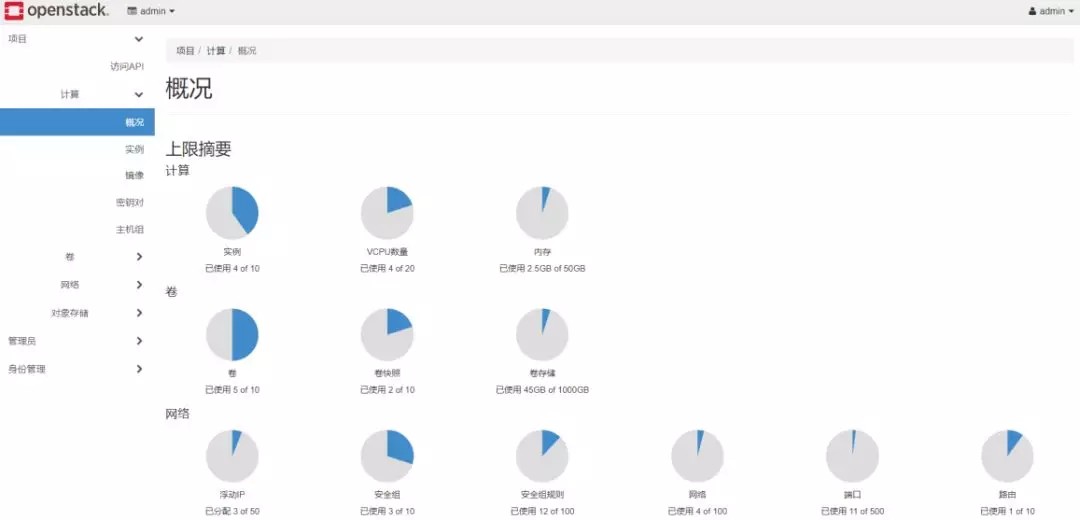


也可以输入OpenStack命令来确认，如：openstack user list列出用户列表。



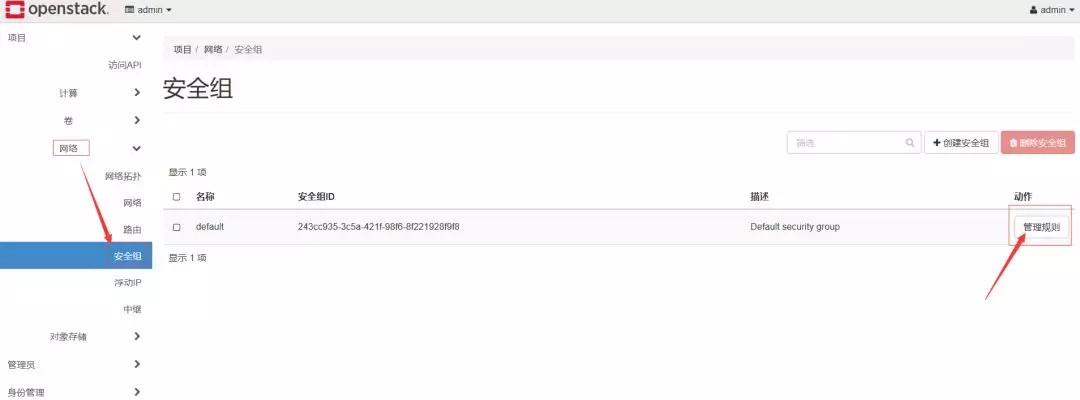
有时候为了方便，”. keystonerc\_admin”也可以认证，注意点号后面有一个空格。

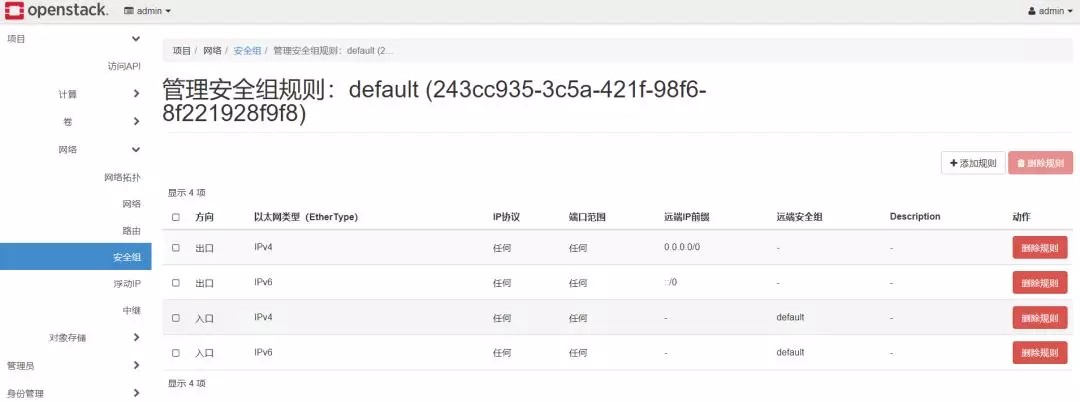
如果5000端口加上登陆不上的话，直接输入ip会默认跳转到登陆页面，输入账号密码即可登陆。界面如下：



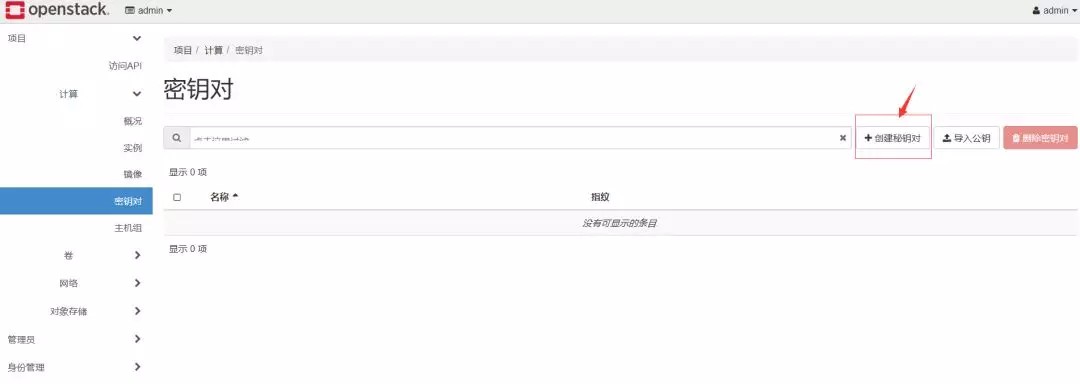
# 二、在OpenStack-horizon创建实例

在OpenStack中实例指的就是vm虚拟机，在dashboard上创建vm的基本步骤：在创建虚拟机之前，要做一些准备，添加安全组，创建ssh密钥对等。当然安全组，密钥对不是必须的，不创建的话，有默认的。下面操作仅为演示：

1、首先为虚拟机添加安全组规则：

添加出口入口icmp规则和ssh规则，保证在vm创建后可以ping通并且可以ssh访问

2、创建密钥对，为后续ssh登陆使用，也可以通过密码登陆：

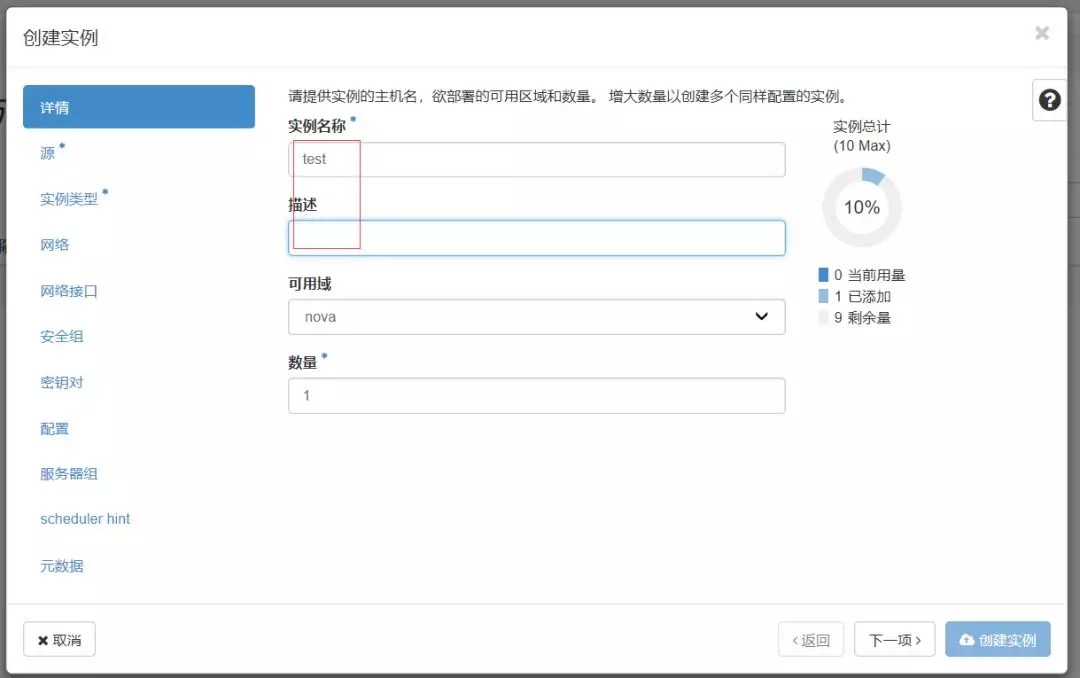


点击创建密钥对后，浏览器会下载后缀为.pem的私钥文件。

将pem文件加权限：chmod 600 key.pem，个人测试的话777也是可以的。

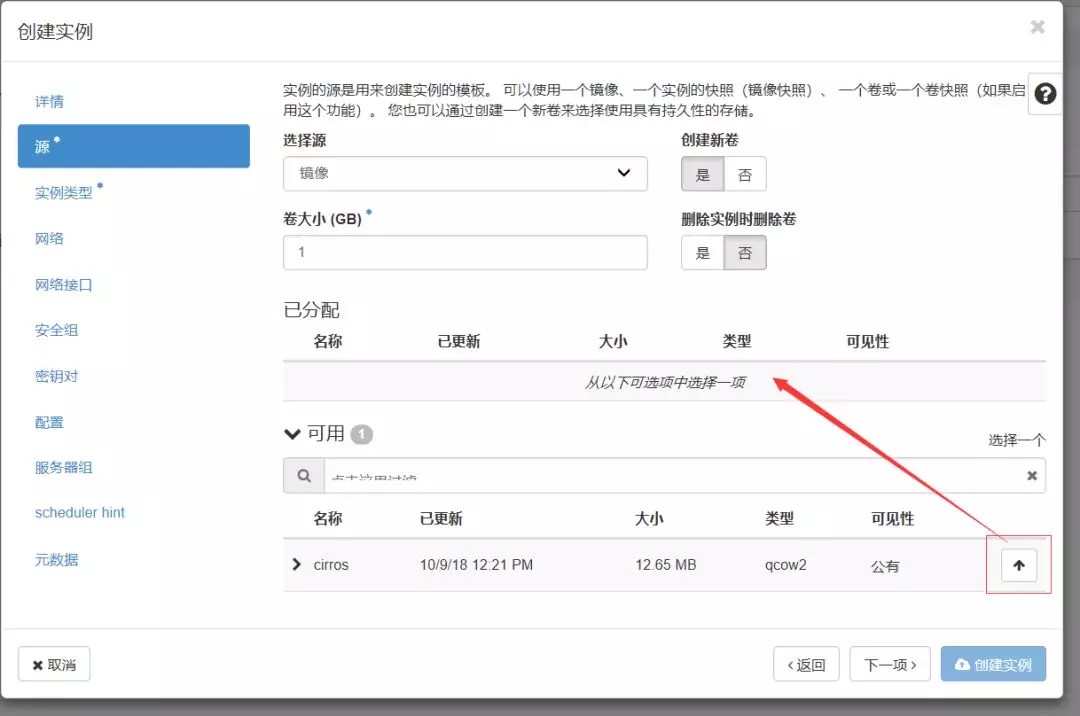
3、在“项目”-“实例”，点击创建实例：

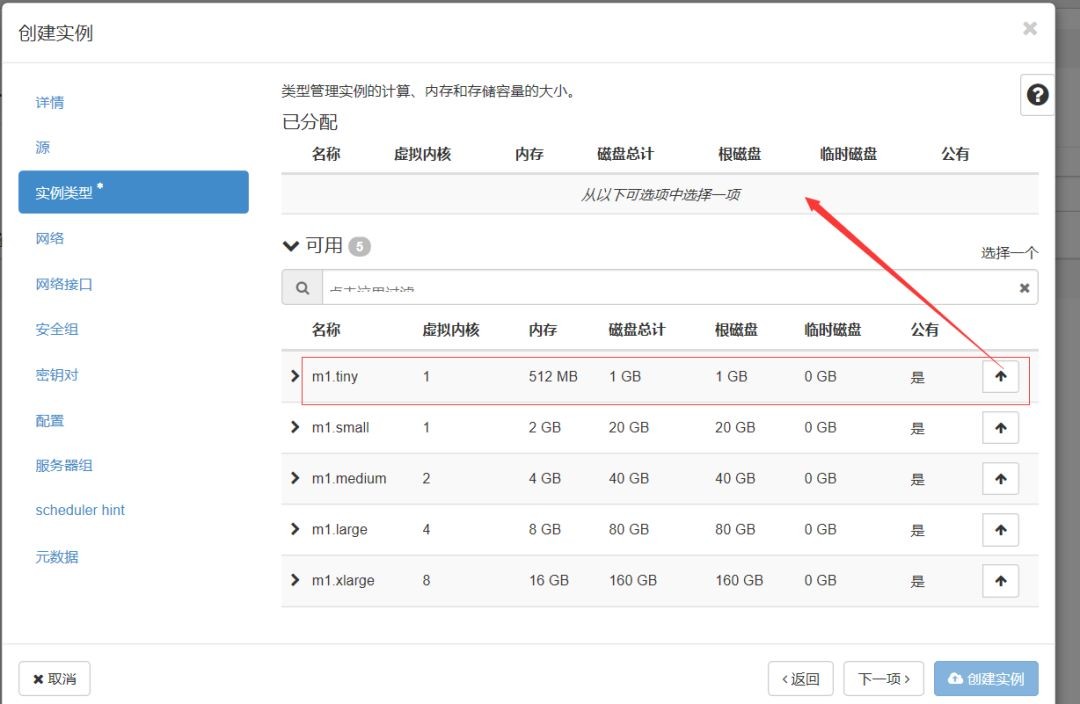
创建名为“test”的实例：



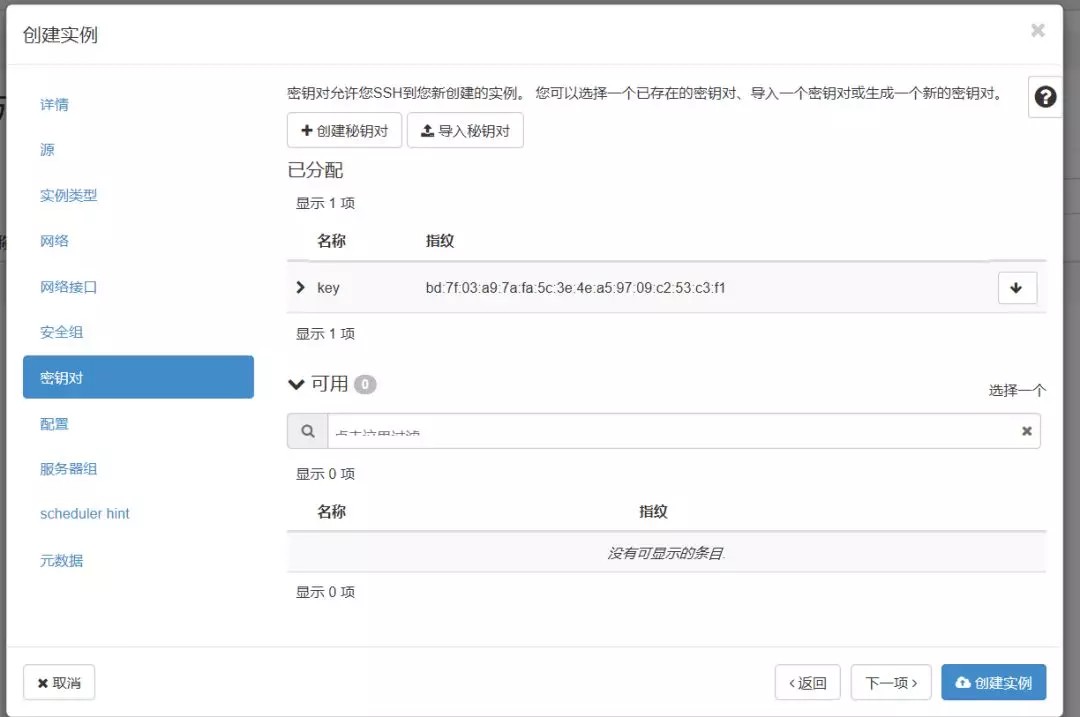
从镜像中选择实例的源：这里我们选择cirros镜像，这是一个在openstack测试使用的镜像，默认自带的， 体积很小，使用起来非常方便。本人曾经安装了一个正常的500M左右的centos镜像，在创建实例过程中， 磁盘必须保证在35GB以上，不然创建就会报错。所以这里测试，就可以使用cirros镜像，大约在十几M左右，后续的flavor（实例类型）就可以选择小的了，减轻了宿主机的压力。

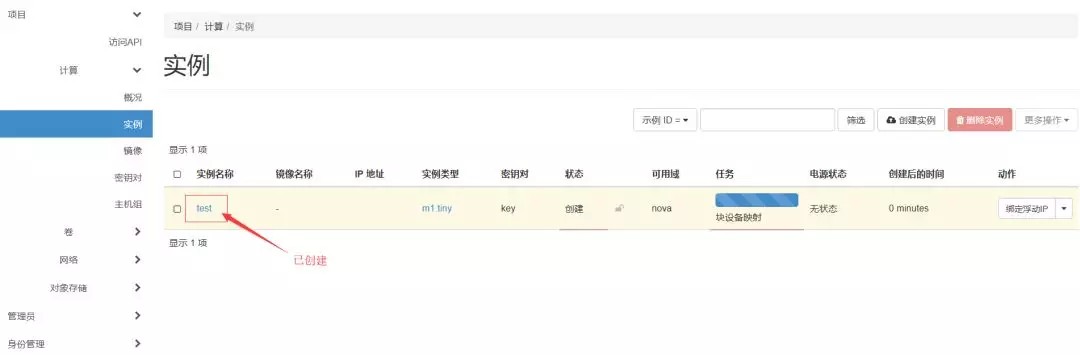
同时选择源：镜像，可以不创建新卷，创建会增加宿主机压力。如果条件允许，那么这一切都不是问题，可以大镜像并且创建新卷。



实例类型可根据需要选取，此处选m1.tiny：

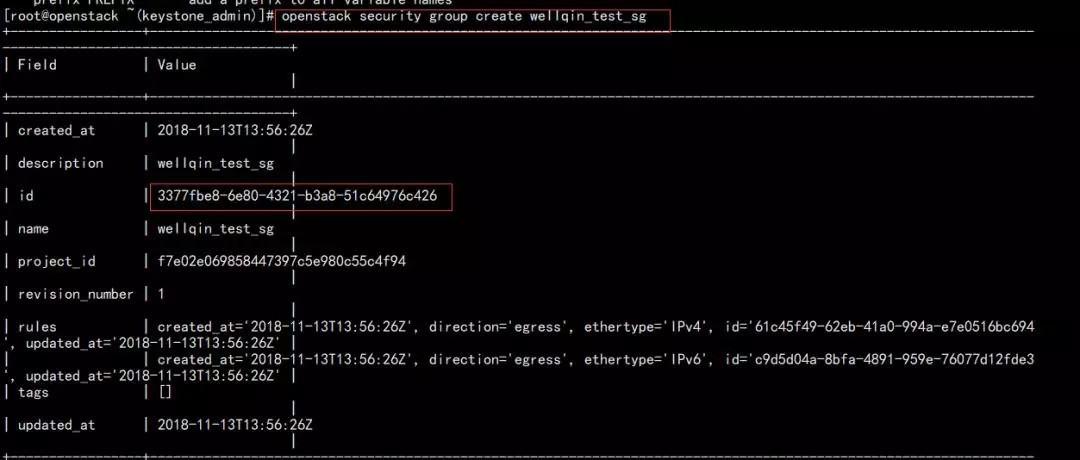
网络暂时默认只有一个私有网络，选择“public\_net”：

再选择创建的密钥对key：

点击创建实例，实例创建需要等待一小段时间，创建完毕后如下图：

任务经历了：块设备映射----孵化----运行中，至此创建实例完成

由于创建的vm是在私有网络中，外界无法直接访问，所以需要绑定floating ip，才可以从外界访问。这一步可以先不执行，可以在后来创建实例完成后绑定。



# 三、OpenStack命令行创建实例

1、首先source环境变量

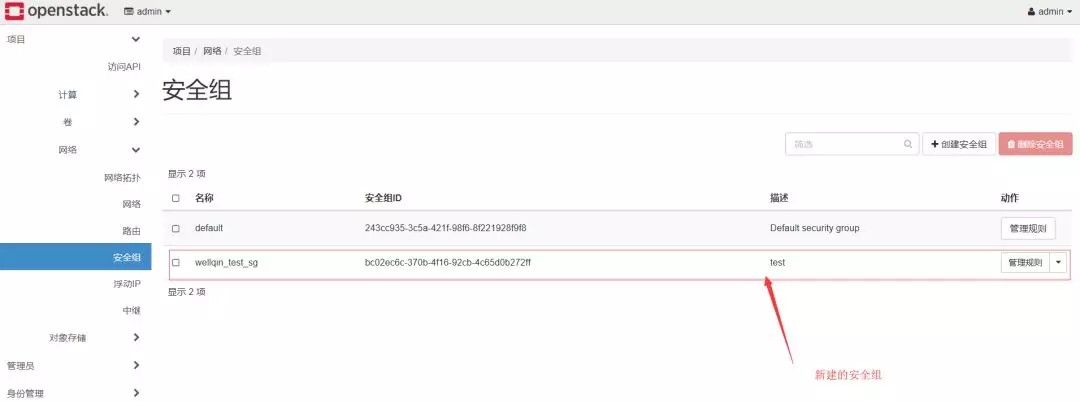
每次登陆节点需要操作openstack命令时，都需要source一次环境变量，这样才可以正确的访问相应的用户信息。

2、生成密钥对

## openstack keypair create --public-key ~/.ssh/id\_rsa.pub default

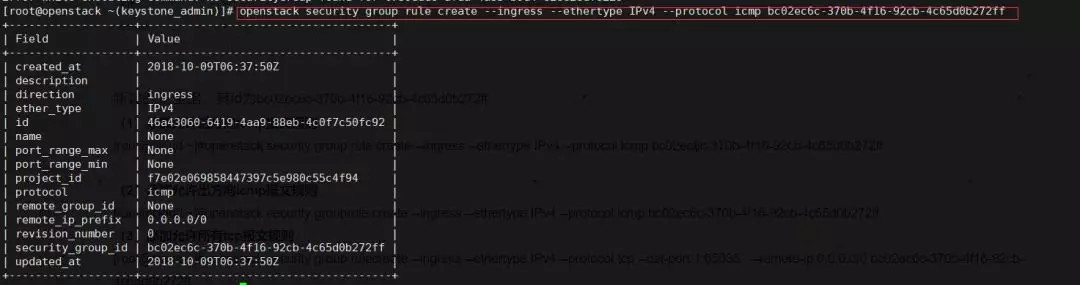
创建安全组规则

在界面上能看到新的安全组：以及id，下图是之前已经创建好的。以它作为示范也是一样的，我们用之前的安全组，其id为bc02ec6c-370b-4f16-92cb-4c65d0b272ff



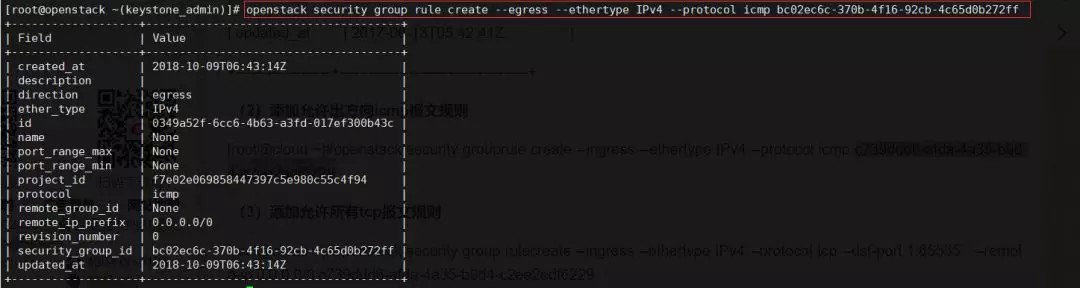
（1）添加允许进方向icmp报文规则

[root@cloud ~]#openstack security group rule create --ingress --ethertype IPv4 --protocol icmp bc02ec6c-370b-4f16-92cb-4c65d0b272ff



（2）添加允许出方向icmp报文规则

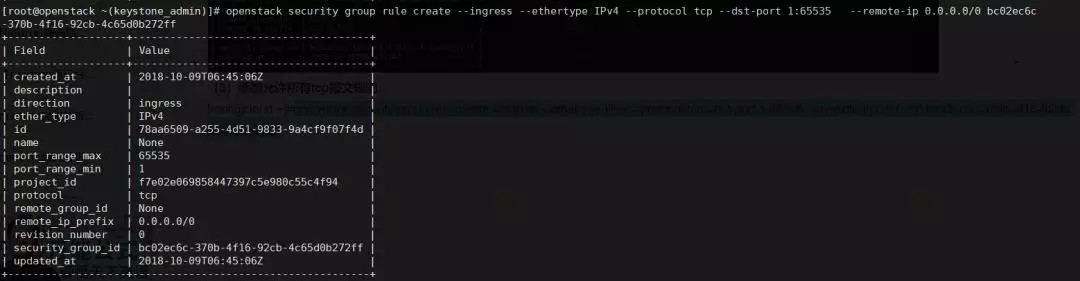
[root@cloud ~]#openstack security group rule create --egress --ethertype IPv4 --protocol icmp bc02ec6c-370b-4f16-92cb-4c65d0b272ff

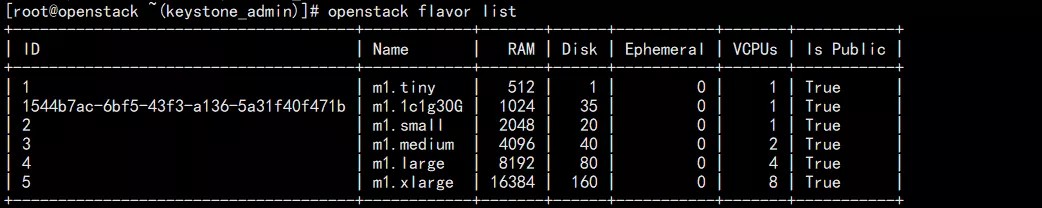


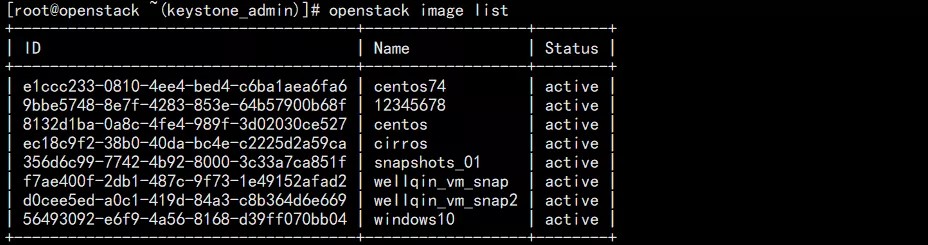
（3）添加允许所有tcp报文规则

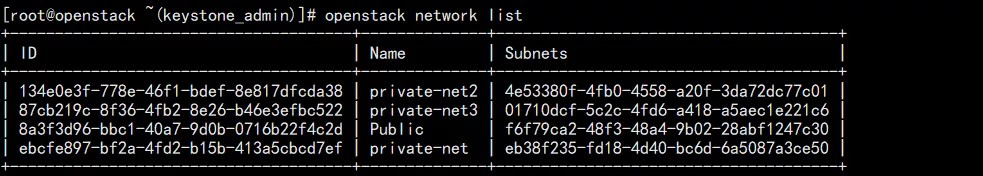
[root@cloud ~]#openstack security group rule create --ingress --ethertype IPv4 --protocol tcp

--dst-port 1:65535 --remote-ip 0.0.0.0/0 bc02ec6c-370b-4f16-92cb-4c65d0b272ff



查看类型模板：openstack flavor list

查看可用镜像：openstack image list

查看可用网络：openstack network list

查看安全组： openstack security group list

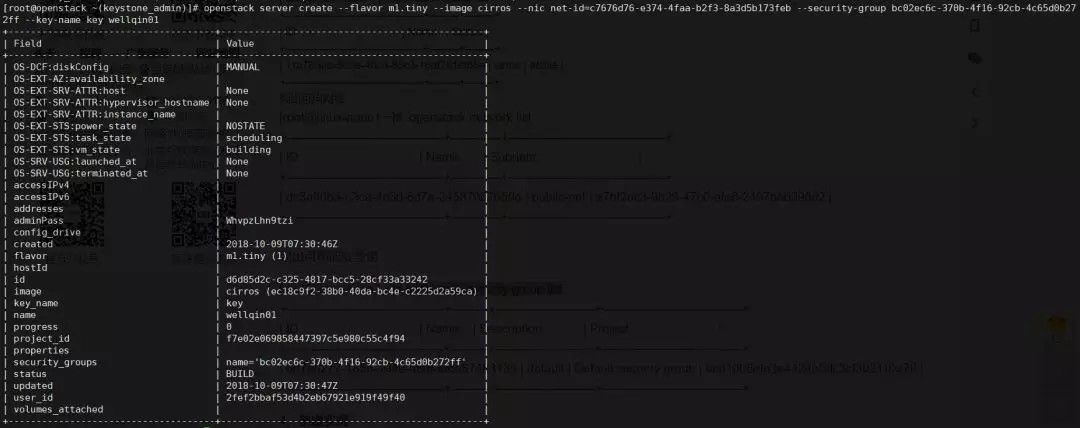
创建实例

之前的查看类型模板、镜像、安全组、网络等就是为了在创建实例的时候，给主机选择合适的而且存在的选项。

执行命令如下：

openstack server create --flavor m1.tiny --image cirros --nic net- id=f7e02e069858447397c5e980c55c4f94 --security-group bc02ec6c-370b-4f16-92cb- 4c65d0b272ff --key-name key wellqin01

# 其中flavor类型模板，image镜像，选的是cirros镜像， net-id是网络id，-security-group后面是指定的安全组id，--key-name是创建的秘钥对名称key，wellqin01是实例名称



此时查看运行的openstack实例，发现已经创建成功了。（上图的net-id值目前被我删掉了，在实际创建之前它是存在的，你可以换成任何存在的id值，希望不要造成误解）

检查实例的状态和登录实例:

## openstack server list

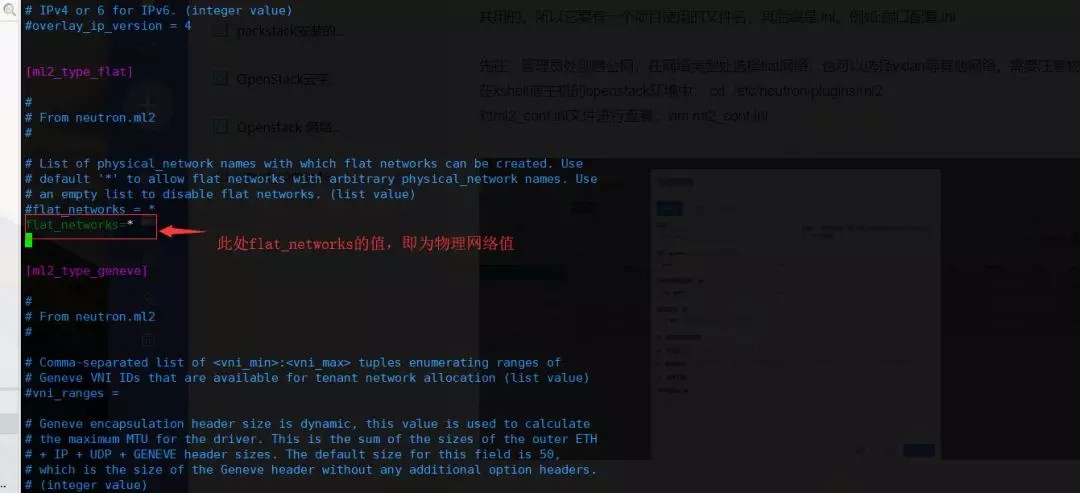
虽然实例创建完成，但是我们一直没有给主机分配浮动ip，这样的话，就不能从外界去访问它，ssh远程连接失败，那么接下来就可以创建vpc网络（为了试验，我们可以分离我们之前的网络），并分配浮动ip给主机。

# 四、OpenStack的VPC网络配置

OpenStack的VPC对应租户的概念(即tenant，后续更新为project)，可以将它简单理解为一个具有隔离性质的网络资源集合，在其内可以提供各种网络资源供虚拟机使用。

现在，我们在OpenStack rocky版本的allinone部署方式中，创建“VPC”网络，并且连接到实例，最后通过浮动ip实现从宿主机外部访问虚机实例。

首先创建外网，admin管理员才有权限创建外网，普通租户是没有权限的。先在管理员处创建公网，在网络类型处选择flat网络，也可以选择vxlan等其他网络。需要注意物理网络的选择。



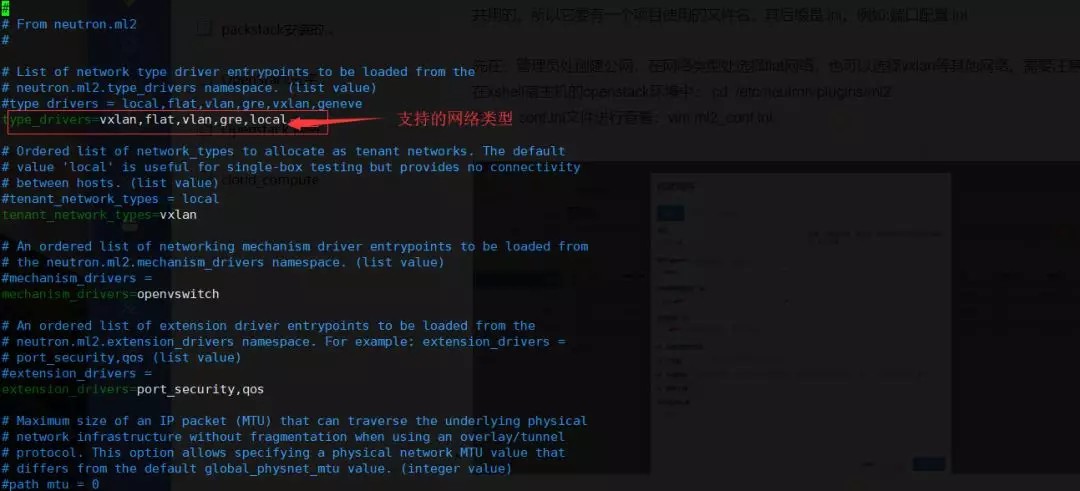
在xshell连接宿主机的OpenStack环境中：

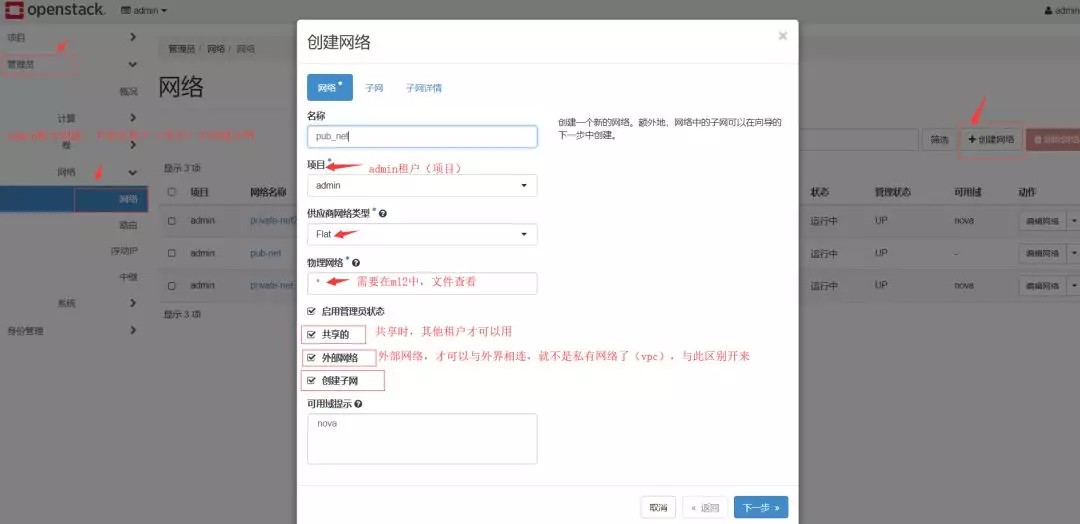
## cd /etc/neutron/plugins/ml2

此文件夹中有以下这三个文件ml2\_conf.ini openvswitch\_agent.ini sriov\_agent.ini

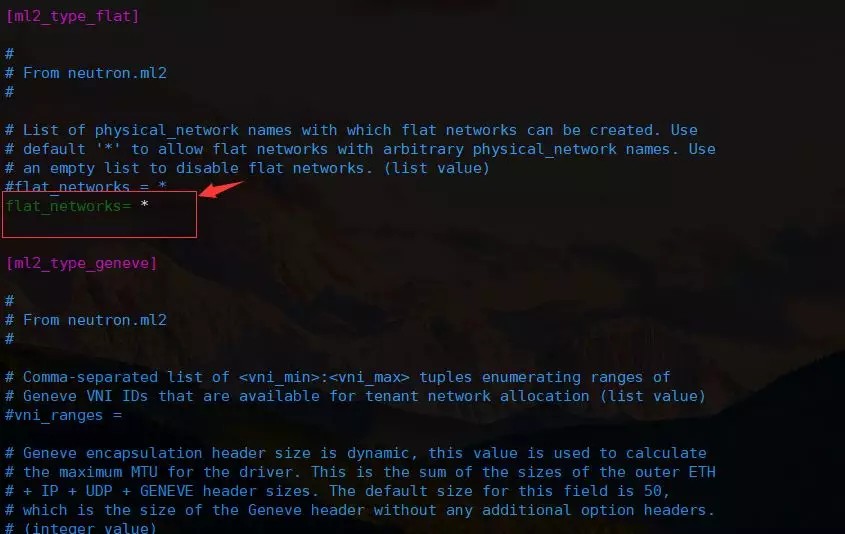
对ml2\_conf.ini文件进行查看：vim ml2\_conf.ini，可看到允许配置的网络属性。

【配置文件.ini，是Initialization File的缩写，即初始化文件，此处是项目中的配置文件，其后缀是.ini。例如:ml2\_conf.ini】

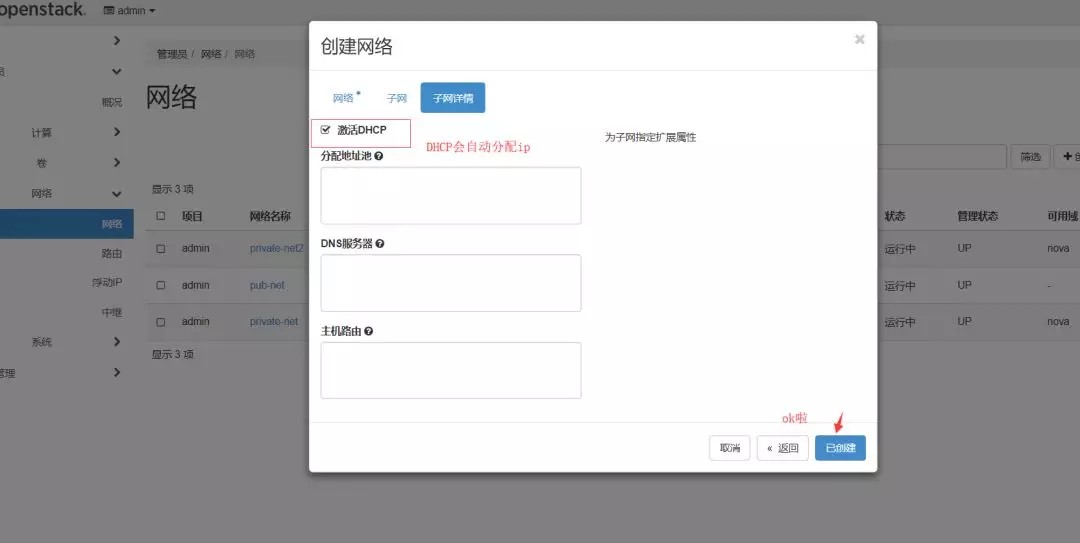




关于物理类型需要填写的值：在上面的ml2\_conf.ini文件中，注意flat\_networks的值，就是物理类型要填 写的值，这里默认为\*号。

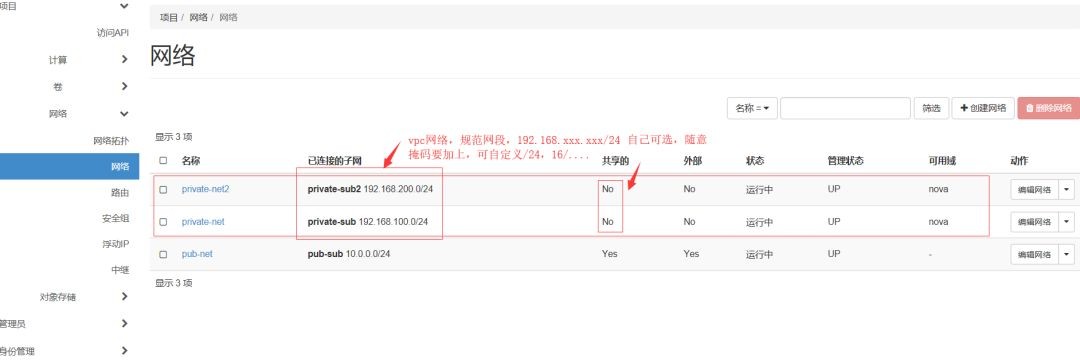
但其实在openvswitch\_agent.ini文件中，bridge\_mappings=extnet:br-ex，所以可填写为extnet。





此时，公网对外创建完成，并且共享。其他租户可以连接。

现在可以去租户界面处，操作，自己创建自己的网络，即vpc网络即虚拟专用网络。创建完成后如下图所示：

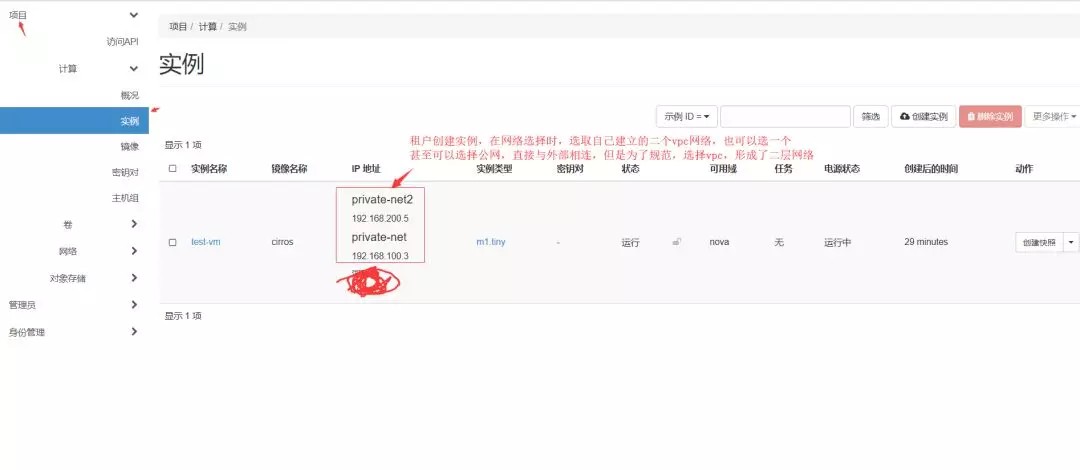


此时，一个对外公网，和二个vpc网络创建完成。可以创建虚拟路由器了。注意路由器是基于外部网络创建的，即上面管理员建立的公网pub\_net

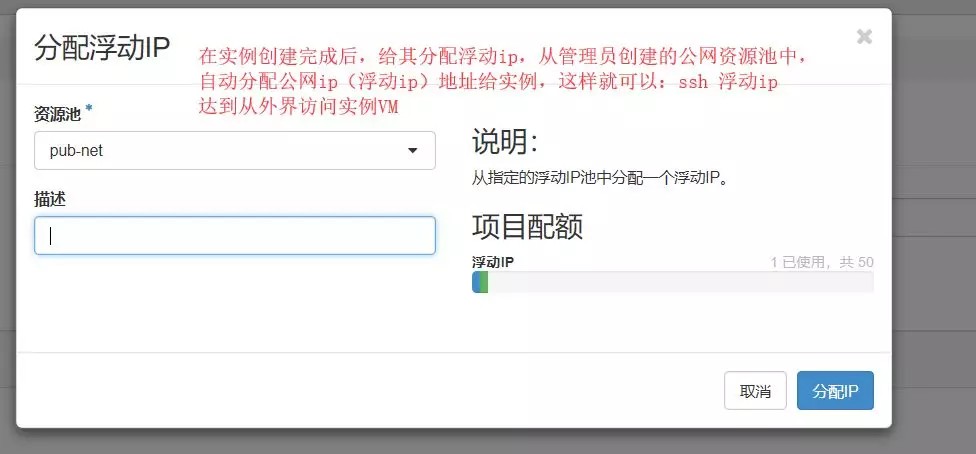


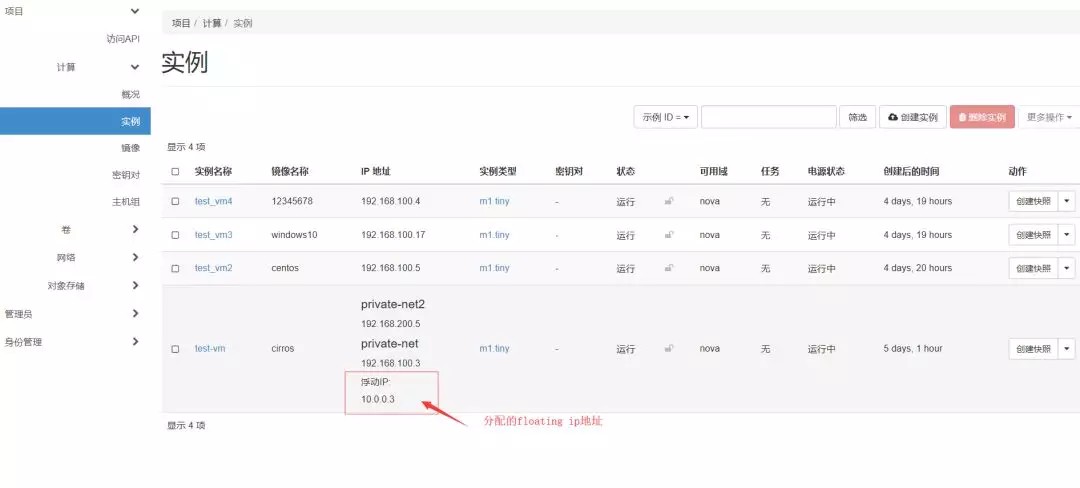
创建路由完成后，点击路由详情，选择路由接口，将以上网络通过路由接口连接起来。

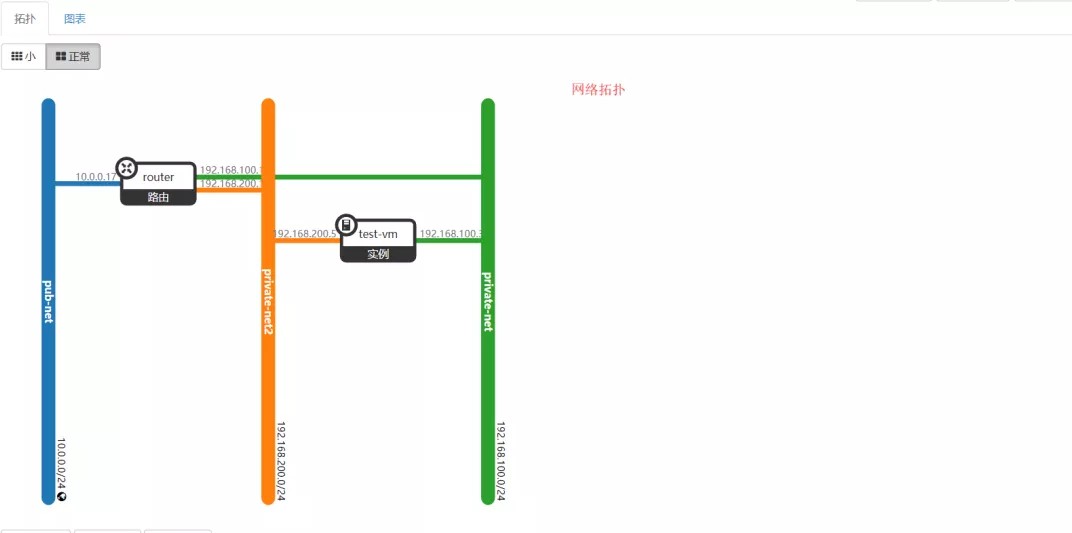
此时基本网络拓扑已经搭建起来，可以创建实例了，可自行选择网络



创建实例完成后如上图，此时可以给创建完成的实例，分配浮动ip，这个操作需要选择资源池，即先前我们创建的公网网段中选择。



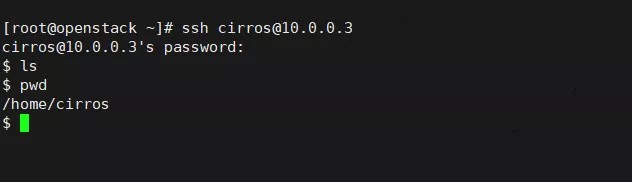
选择后，会自动分配一个ip地址，即浮动ip给创建的实例VM

至此，完整的网络拓扑如下图：

在postman中请求主机详情字段中：

"OS-EXT-IPS:type": "fixed" 是 vpc 的 ip

"OS-EXT-IPS:type": "floating" 是浮动ip，对外暴露的连接口。



至此，我们在宿主机中去ssh连接VM，

可以从外部去访问VM实例。最后注意浮动ip不是公网ip，二者是不同的。

定义是不是公网的ip就要看整个internet，全球任何一个其他的ip节点都能访问到的IP才是公网IP，公网ip 需要企业注册，外网属于内部员工使用。

而外网只是相对于某一个“内网”而言，也可能好几层内网包着，外面的那层相对里面包着的都是外网，但是 这些外网仍然不是公网。

# 总结

通过allinone的方式部署，是体验OpenStack的一个不错的方法，在VPC配置方面，从外部访问虚拟机内部的网络，需要绑定浮动ip，即指定 router 上应用 iptalbes 的 NAT 规则，使得外部可以通过并访问虚拟机，以上就是OpenStack创建VPC网络并连接实例的基本过程。