搭建OpenStack私有云环境

1. 环境准备

基于之前的kvm环境，做好centos系统的模板机

卸载掉NetworkManager、firewalld等，设置好Selinux

[root@room8pc205 images]# ls .centos\_tem.\*

.centos\_tem.qcow2 .centos\_tem.xml

我这里显示的是模板机的虚拟磁盘文件和xml文件

1. 准备镜像

[root@room8pc205 ~]# ls /ISO/

CentOS7-1708.iso RHEL7-extras.iso RHEL7OSP-10.iso rhel-8.0-x86\_64-dvd.iso rhel-server-7.5-x86\_64-dvd.iso

#需要准备标红的这三个镜像文件

1. 宿主机配置转发dns服务

#找到宿主机所用的dns服务器IP地址

[root@room8pc205 ~]# cat /etc/resolv.conf

# Generated by NetworkManager

search localdomain tedu.cn

nameserver 192.168.100.2

nameserver 192.168.32.1

#安装dns服务

[root@room8pc205 ~]# yum -y install bind bind-chroot

#配置dns服务

[root@room8pc205 ~]# vim /etc/named.conf

[root@room8pc205 ~]# cat /etc/named.conf

options {

listen-on port 53 { 192.168.1.254; };

allow-query { any; };

forwarders { 192.168.32.1; };

recursion yes;

dnssec-enable no;

dnssec-validation no;

};

[root@room8pc205 ~]# systemctl start named

#测试缓存dns服务器

[root@room8pc205 ~]# nslookup

> server 192.168.1.254

Default server: 192.168.1.254

Address: 192.168.1.254#53

> set q=a

> www.baidu.com

Server: 192.168.1.254

Address: 192.168.1.254#53

Non-authoritative answer:

www.baidu.com canonical name = www.a.shifen.com.

Name: www.a.shifen.com

Address: 182.61.200.7

Name: www.a.shifen.com

Address: 182.61.200.6

> exit

[root@room8pc205 ~]#

1. 搭建本地ntp时间同步服务器

#安装chrony包

[root@room8pc205 ~]# yum -y install chrony

#修改配置文件

[root@room8pc205 ~]# vim /etc/chrony.conf

[root@room8pc205 ~]# cat /etc/chrony.conf | grep -v ^# | grep -v ^$

server ntp1.aliyun.com iburst

stratumweight 10

driftfile /var/lib/chrony/drift

rtcsync

makestep 10 3

bindcmdaddress 127.0.0.1

bindcmdaddress ::1

bindacqaddress 0.0.0.0

allow 0/0

cmdallow 127.0.0.1

keyfile /etc/chrony.keys

commandkey 1

generatecommandkey

noclientlog

logchange 0.5

logdir /var/log/chrony

#启动服务并验证结果

[root@room8pc205 ~]# systemctl restart chronyd

[root@room8pc205 ~]# chronyc sources -v

210 Number of sources = 1

.-- Source mode '^' = server, '=' = peer, '#' = local clock.

/ .- Source state '\*' = current synced, '+' = combined , '-' = not combined,

| / '?' = unreachable, 'x' = time may be in error, '~' = time too variable.

|| .- xxxx [ yyyy ] +/- zzzz

|| Reachability register (octal) -. | xxxx = adjusted offset,

|| Log2(Polling interval) --. | | yyyy = measured offset,

|| \ | | zzzz = estimated error.

|| | | \

MS Name/IP address Stratum Poll Reach LastRx Last sample

===============================================================================

^\* 120.25.115.20 2 6 33 3 -543us[-1180us] +/- 28ms

[root@room8pc205 ~]#

1. 挂载OpenStack所需镜像

#修改fstab，永久挂载镜像

[root@room8pc205 ~]# vim /etc/fstab

[root@room8pc205 ~]# tail -3 /etc/fstab

/ISO/CentOS7-1708.iso /var/ftp/pub/iso/centos iso9660 defaults 0 0

/ISO/RHEL7-extras.iso /var/ftp/pub/iso/extras iso9660 defaults 0 0

/ISO/RHEL7OSP-10.iso /var/ftp/pub/iso/osp iso9660 defaults 0 0

[root@room8pc205 ~]# tail -3 /etc/fstab

/ISO/CentOS7-1708.iso /var/ftp/pub/iso/centos iso9660 defaults 0 0

/ISO/RHEL7-extras.iso /var/ftp/pub/iso/extras iso9660 defaults 0 0

/ISO/RHEL7OSP-10.iso /var/ftp/pub/iso/osp iso9660 defaults 0 0

[root@room8pc205 ~]# mkdir -p /var/ftp/pub/iso/{centos,extras,osp}

[root@room8pc205 ~]# mount -a

mount: /dev/loop0 写保护，将以只读方式挂载

mount: /dev/loop2 写保护，将以只读方式挂载

mount: /dev/loop3 写保护，将以只读方式挂载

#创建repo源文件

#由于之前以前做好centos的源，所以此处制作extras和ops的源

[root@room8pc205 ~]# ls -d /var/ftp/pub/iso/osp/rhel-7-server-\* > 1.txt

[root@room8pc205 ~]# ls -d /var/ftp/pub/iso/extras >> 1.txt

[root@room8pc205 ~]# vim 1.txt

[root@room8pc205 ~]# cat 1.txt

pub/iso/osp/rhel-7-server-openstack-10-devtools-rpms

pub/iso/osp/rhel-7-server-openstack-10-optools-rpms

pub/iso/osp/rhel-7-server-openstack-10-rpms

pub/iso/osp/rhel-7-server-openstack-10-tools-rpms

pub/iso/osp/rhel-7-server-rhceph-2-mon-rpms

pub/iso/osp/rhel-7-server-rhceph-2-osd-rpms

pub/iso/osp/rhel-7-server-rhceph-2-tools-rpms

pub/iso/osp/rhel-7-server-rhscon-2-agent-rpms

pub/iso/osp/rhel-7-server-rhscon-2-installer-rpms

pub/iso/osp/rhel-7-server-rhscon-2-main-rpms

pub/iso/extras

[root@room8pc205 ~]# for i in `cat 1.txt`; do

cat >> openstack.repo << EOF

[${i##\*/}]

name=$i

baseurl=ftp://192.168.1.254/$i

enabled=1

gpgcheck=0

EOF

done

[root@room8pc205 ~]# cat openstack.repo

[rhel-7-server-openstack-10-devtools-rpms]

name=pub/iso/osp/rhel-7-server-openstack-10-devtools-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-openstack-10-devtools-rpms

enabled=1

gpgcheck=0

[rhel-7-server-openstack-10-optools-rpms]

name=pub/iso/osp/rhel-7-server-openstack-10-optools-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-openstack-10-optools-rpms

enabled=1

gpgcheck=0

[rhel-7-server-openstack-10-rpms]

name=pub/iso/osp/rhel-7-server-openstack-10-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-openstack-10-rpms

enabled=1

gpgcheck=0

[rhel-7-server-openstack-10-tools-rpms]

name=pub/iso/osp/rhel-7-server-openstack-10-tools-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-openstack-10-tools-rpms

enabled=1

gpgcheck=0

[rhel-7-server-rhceph-2-mon-rpms]

name=pub/iso/osp/rhel-7-server-rhceph-2-mon-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-rhceph-2-mon-rpms

enabled=1

gpgcheck=0

[rhel-7-server-rhceph-2-osd-rpms]

name=pub/iso/osp/rhel-7-server-rhceph-2-osd-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-rhceph-2-osd-rpms

enabled=1

gpgcheck=0

[rhel-7-server-rhceph-2-tools-rpms]

name=pub/iso/osp/rhel-7-server-rhceph-2-tools-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-rhceph-2-tools-rpms

enabled=1

gpgcheck=0

[rhel-7-server-rhscon-2-agent-rpms]

name=pub/iso/osp/rhel-7-server-rhscon-2-agent-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-rhscon-2-agent-rpms

enabled=1

gpgcheck=0

[rhel-7-server-rhscon-2-installer-rpms]

name=pub/iso/osp/rhel-7-server-rhscon-2-installer-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-rhscon-2-installer-rpms

enabled=1

gpgcheck=0

[rhel-7-server-rhscon-2-main-rpms]

name=pub/iso/osp/rhel-7-server-rhscon-2-main-rpms

baseurl=ftp://192.168.1.254/pub/iso/osp/rhel-7-server-rhscon-2-main-rpms

enabled=1

gpgcheck=0

[extras]

name=pub/iso/extras

baseurl=ftp://192.168.1.254/pub/iso/extras

enabled=1

gpgcheck=0

[root@room8pc205 ~]# scp openstack.repo 192.168.1.11:/etc/yum.repos.d/

root@192.168.1.11's password:

openstack.repo 100% 1955 1.0MB/s 00:00

[root@room8pc205 ~]#

#算上centos的源共12个源

1. 准备两台虚拟机，我这里由于电脑内存限制，达不到讲师讲课的要求，所以OpenStack和Nova的内存均设置成2G

6.1、创建OpenStack虚拟机

#虚拟机1要求：内存2G，两块网卡（我这里三块），分别接入vbr，private1，设置静态IP地址，默认网关192.168.1.254，两块硬盘，其中vda50G，vdb20G

#创建虚拟磁盘和xml文件

[root@room8pc205 ~]# cd /var/lib/libvirt/images/

[root@room8pc205 images]# ls .centos\_tem.\*

.centos\_tem.qcow2 .centos\_tem.xml

[root@room8pc205 images]# qemu-img create -f qcow2 -b .centos\_tem.qcow2 node11.qcow2 50G

Formatting 'node11.qcow2', fmt=qcow2 size=53687091200 backing\_file='.centos\_tem.qcow2' encryption=off cluster\_size=65536 lazy\_refcounts=off

[root@room8pc205 images]# qemu-img create -f qcow2 node11-1.qcow2 20G

Formatting 'node11-1.qcow2', fmt=qcow2 size=21474836480 encryption=off cluster\_size=65536 lazy\_refcounts=off

[root@room8pc205 images]# cat .centos\_tem.xml > node11.xml

[root@room8pc205 images]# vim node11.xml

[root@room8pc205 images]# cat node11.xml

<domain type='kvm'>

<name>node11</name>

<memory unit='KB'>2097152</memory>

<currentMemory unit='KB'>2097152</currentMemory>

<vcpu placement='static'>2</vcpu>

<os>

<type arch='x86\_64' machine='pc'>hvm</type>

<boot dev='hd'/>

<bootmenu enable='yes'/>

<bios useserial='yes'/>

</os>

<features>

<acpi/>

<apic/>

</features>

<cpu mode='host-passthrough'>

</cpu>

<clock offset='localtime'/>

<on\_poweroff>destroy</on\_poweroff>

<on\_reboot>restart</on\_reboot>

<on\_crash>restart</on\_crash>

<devices>

<emulator>/usr/libexec/qemu-kvm</emulator>

<disk type='file' device='disk'>

<driver name='qemu' type='qcow2'/>

<source file='/var/lib/libvirt/images/node11.qcow2'/>

<target dev='vda' bus='virtio'/>

</disk>

<disk type='file' device='disk'>

<driver name='qemu' type='qcow2'/>

<source file='/var/lib/libvirt/images/node11-1.qcow2'/>

<target dev='vdb' bus='virtio'/>

</disk>

<interface type='bridge'>

<source bridge='vbr'/>

<model type='virtio'/>

</interface>

<interface type='bridge'>

<source bridge='private1'/>

<model type='virtio'/>

</interface>

<channel type='unix'>

<target type='virtio' name='org.qemu.guest\_agent.0'/>

</channel>

<serial type='pty'></serial>

<console type='pty'>

<target type='serial'/>

</console>

<memballoon model='virtio'></memballoon>

</devices>

</domain>

[root@room8pc205 images]# virsh define node11.xml

定义域 node11（从 node11.xml）

[root@room8pc205 images]# virsh start --console node11

#登录console控制台，查看动态获取IP地址

localhost login: root

Password:

Last login: Tue May 28 14:11:07 on ttyS0

[root@localhost ~]# ifconfig eth0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

inet 192.168.1.116 netmask 255.255.255.0 broadcast 192.168.1.255

ether 52:54:00:c8:d0:09 txqueuelen 1000 (Ethernet)

RX packets 30 bytes 2429 (2.3 KiB)

RX errors 0 dropped 12 overruns 0 frame 0

TX packets 6 bytes 896 (896.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

#退出ssh接入

[root@room8pc205 images]# ssh 192.168.1.116

[root@localhost ~]# hostnamectl set-hostname openstack

[root@localhost ~]# cd /etc/sysconfig/network-scripts/

[root@localhost network-scripts]# cp ifcfg-eth0 ifcfg-eth1

[root@localhost network-scripts]# vim ifcfg-eth0

[root@localhost network-scripts]# vim ifcfg-eth1 root@localhost network-scripts]# cat ifcfg-eth0

# Generated by dracut initrd

DEVICE="eth0"

ONBOOT="yes"

IPV6INIT="no"

IPV4\_FAILURE\_FATAL="no"

NM\_CONTROLLED="no"

TYPE="Ethernet"

BOOTPROTO="none"

IPADDR=192.168.1.11

PREFIX=24

GATEWAY=192.168.1.254

[root@localhost network-scripts]# cat ifcfg-eth1

# Generated by dracut initrd

DEVICE="eth1"

ONBOOT="yes"

IPV6INIT="no"

IPV4\_FAILURE\_FATAL="no"

NM\_CONTROLLED="no"

TYPE="Ethernet"

BOOTPROTO="none"

IPADDR=192.168.4.11

PREFIX=24

[root@localhost network-scripts]# reboot

[root@room8pc205 images]# ssh 192.168.1.11

Warning: Permanently added '192.168.1.11' (ECDSA) to the list of known hosts.

root@192.168.1.11's password:

Permission denied, please try again.

root@192.168.1.11's password:

Last failed login: Tue May 28 15:38:18 CST 2019 from 192.168.1.254 on ssh:notty

There was 1 failed login attempt since the last successful login.

Last login: Tue May 28 15:29:46 2019 from 192.168.1.254

[root@openstack ~]# ip a s | grep 192.168

inet 192.168.1.11/24 brd 192.168.1.255 scope global eth0

inet 192.168.4.11/24 brd 192.168.4.255 scope global eth1

[root@openstack ~]#

[root@openstack ~]# df -h

文件系统 容量 已用 可用 已用% 挂载点

/dev/vda1 50G 1.1G 1001M 51% /

devtmpfs 887M 0 887M 0% /dev

tmpfs 896M 0 896M 0% /dev/shm

tmpfs 896M 8.3M 888M 1% /run

tmpfs 896M 0 896M 0% /sys/fs/cgroup

tmpfs 180M 0 180M 0% /run/user/0

[root@openstack ~]# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

vda 253:0 0 50G 0 disk

└─vda1 253:1 0 2G 0 part /

vdb 253:16 0 20G 0 disk

[root@openstack ~]# nslookup

> server 192.168.1.254

Default server: 192.168.1.254

Address: 192.168.1.254#53

> set q=a

> www.baidu.com

Server: 192.168.1.254

Address: 192.168.1.254#53

Non-authoritative answer:

www.baidu.com canonical name = www.a.shifen.com.

Name: www.a.shifen.com

Address: 182.61.200.6

Name: www.a.shifen.com

Address: 182.61.200.7

> exit

[root@openstack ~]#

#至此虚拟机1部署完成

6.2、创建Nova虚拟机，

#虚拟机2要求：内存2G，两块网卡，分别接入vbr，private1，设置静态IP地址，默认网关192.168.1.254，两块硬盘，其中vda50G，vdb20G

#创建虚拟机2的xml文件和磁盘文件

[root@room8pc205 images]# qemu-img create -f qcow2 -b .centos\_tem.qcow2 node12.qcow2 50G

Formatting 'node12.qcow2', fmt=qcow2 size=53687091200 backing\_file='.centos\_tem.qcow2' encryption=off cluster\_size=65536 lazy\_refcounts=off

[root@room8pc205 images]# cat .centos\_tem.xml > node12.xml

[root@room8pc205 images]# vim node12.xml

[root@room8pc205 images]# cat node12.xml

<domain type='kvm'>

<name>node12</name>

<memory unit='KB'>2097152</memory>

<currentMemory unit='KB'>2097152</currentMemory>

<vcpu placement='static'>2</vcpu>

<os>

<type arch='x86\_64' machine='pc'>hvm</type>

<boot dev='hd'/>

<bootmenu enable='yes'/>

<bios useserial='yes'/>

</os>

<features>

<acpi/>

<apic/>

</features>

<cpu mode='host-passthrough'>

</cpu>

<clock offset='localtime'/>

<on\_poweroff>destroy</on\_poweroff>

<on\_reboot>restart</on\_reboot>

<on\_crash>restart</on\_crash>

<devices>

<emulator>/usr/libexec/qemu-kvm</emulator>

<disk type='file' device='disk'>

<driver name='qemu' type='qcow2'/>

<source file='/var/lib/libvirt/images/node12.qcow2'/>

<target dev='vda' bus='virtio'/>

</disk>

<interface type='bridge'>

<source bridge='vbr'/>

<model type='virtio'/>

</interface>

<interface type='bridge'>

<source bridge='private1'/>

<model type='virtio'/>

</interface>

<channel type='unix'>

<target type='virtio' name='org.qemu.guest\_agent.0'/>

</channel>

<serial type='pty'></serial>

<console type='pty'>

<target type='serial'/>

</console>

<memballoon model='virtio'></memballoon>

</devices>

</domain>

[root@room8pc205 images]# virsh define node12.xml

定义域 node12（从 node12.xml）

[root@room8pc205 images]# virsh start --console node12

#获取动态dhcp地址，然后ssh接入修改配置

localhost login: root

Password:

Last login: Tue May 28 14:11:07 on ttyS0

[root@localhost ~]# ifconfig eth0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

inet 192.168.1.167 netmask 255.255.255.0 broadcast 192.168.1.255

ether 52:54:00:9f:2a:4a txqueuelen 1000 (Ethernet)

RX packets 39 bytes 2867 (2.7 KiB)

RX errors 0 dropped 13 overruns 0 frame 0

TX packets 6 bytes 896 (896.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@localhost ~]#

[root@room8pc205 images]# ssh 192.168.1.167

Warning: Permanently added '192.168.1.167' (ECDSA) to the list of known hosts.

root@192.168.1.167's password:

Last login: Tue May 28 16:06:46 2019

[root@localhost ~]# hostnamectl set-hostname nova

[root@localhost ~]# cd /etc/sysconfig/network-scripts/

[root@localhost network-scripts]# cp ifcfg-eth0 ifcfg-eth1

[root@localhost network-scripts]# vim ifcfg-eth0 ★ --虚拟机--★ root@localhost network-scripts]# vim ifcfg-eth1

[root@localhost network-scripts]# cat ifcfg-eth0 ifcfg-eth1

# Generated by dracut initrd

DEVICE="eth0"

ONBOOT="yes"

IPV6INIT="no"

IPV4\_FAILURE\_FATAL="no"

NM\_CONTROLLED="no"

TYPE="Ethernet"

BOOTPROTO="none"

IPADDR=192.168.1.12

PREFIX=24

GATEWAY=192.168.1.254

# Generated by dracut initrd

DEVICE="eth1"

ONBOOT="yes"

IPV6INIT="no"

IPV4\_FAILURE\_FATAL="no"

NM\_CONTROLLED="no"

TYPE="Ethernet"

BOOTPROTO="none"

IPADDR=192.168.4.12

PREFIX=24

[root@localhost network-scripts]# reboot

#验证结果

[root@room8pc205 images]# ssh 192.168.1.12

Warning: Permanently added '192.168.1.12' (ECDSA) to the list of known hosts. ★ --虚拟机--★ oot@192.168.1.12's password: ★ --虚拟机--★ ast login: Tue May 28 16:07:26 2019 from 192.168.1.254

[root@nova ~]# ip a s | grep 192.168

inet 192.168.1.12/24 brd 192.168.1.255 scope global eth0

inet 192.168.4.12/24 brd 192.168.4.255 scope global eth0

[root@nova ~]# route -n

Kernel IP routing table

Destination Gateway Genmask Flags Metric Ref Use Iface

0.0.0.0 192.168.1.254 0.0.0.0 UG 0 0 0 eth0

192.168.1.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0

192.168.4.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0

[root@nova ~]# df -h

文件系统 容量 已用 可用 已用% 挂载点

/dev/vda1 50G 1.1G 1001M 51% /

devtmpfs 887M 0 887M 0% /dev

tmpfs 896M 0 896M 0% /dev/shm

tmpfs 896M 8.3M 888M 1% /run

tmpfs 896M 0 896M 0% /sys/fs/cgroup ★ --虚拟机--★ mpfs 180M 0 180M 0% /run/user/0

[root@nova ~]# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

vda 253:0 0 50G 0 disk

└─vda1 253:1 0 2G 0 part /

[root@nova ~]# nslookup

> server 192.168.1.254

Default server: 192.168.1.254

Address: 192.168.1.254#53

> set q=a

> www.baidu.com

Server: 192.168.1.254

Address: 192.168.1.254#53

Non-authoritative answer:

www.baidu.com canonical name = www.a.shifen.com.

Name: www.a.shifen.com

Address: 182.61.200.7

Name: www.a.shifen.com

Address: 182.61.200.6

> exit

[root@nova ~]#

#至此，两天虚拟机部署完毕，注意虚拟机的磁盘大小，是否扩容成功

1. 测试yum源，配置虚拟机的时间同步服务，域名解析

#7.1、配置yum源，共12个

[root@room8pc205 ~]# scp openstack.repo 192.168.1.11:/etc/yum.repos.d/

root@192.168.1.11's password:

openstack.repo 100% 1955 1.0MB/s 00:00

[root@room8pc205 ~]# scp openstack.repo 192.168.1.12:/etc/yum.repos.d/

root@192.168.1.12's password:

openstack.repo 100% 1955 2.4MB/s 00:00

[root@room8pc205 ~]#

[root@openstack ~]# yum repolist | tail -1

repolist: 10,731

[root@nova ~]# yum repolist | tail -1

repolist: 10,731

#设置域名解析

[root@openstack ~]# vim /etc/hosts

[root@openstack ~]# tail -4 /etc/hosts

192.168.1.11 openstack

192.168.4.11 openstack

192.168.1.12 nova

192.168.4.12 nova

[root@openstack ~]# scp /etc/hosts

hosts hosts.allow hosts.deny

[root@openstack ~]# scp /etc/hosts 192.168.1.12:/etc/

root@192.168.1.12's password:

hosts 100% 233 96.4KB/s 00:00

[root@openstack ~]#

[root@openstack ~]# ping -c 2 openstack

PING openstack (192.168.1.11) 56(84) bytes of data.

64 bytes from openstack (192.168.1.11): icmp\_seq=1 ttl=255 time=0.101 ms

64 bytes from openstack (192.168.1.11): icmp\_seq=2 ttl=255 time=0.023 ms

--- openstack ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1000ms

rtt min/avg/max/mdev = 0.023/0.062/0.101/0.039 ms

[root@openstack ~]# ping -c 2 nova

PING nova (192.168.1.12) 56(84) bytes of data.

64 bytes from nova (192.168.1.12): icmp\_seq=1 ttl=255 time=1.45 ms

64 bytes from nova (192.168.1.12): icmp\_seq=2 ttl=255 time=1.17 ms

--- nova ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1003ms

rtt min/avg/max/mdev = 1.179/1.318/1.458/0.144 ms

#设置时间同步服务

[root@openstack ~]# yum -y install chrony

[root@openstack ~]# vim /etc/chrony.conf

[root@openstack ~]# cat /etc/chrony.conf | grep -v ^$ | grep -v ^#

server 192.168.1.254 iburst

cmdallow 127.0.0.1

driftfile /var/lib/chrony/drift

makestep 1.0 3

rtcsync

logdir /var/log/chrony

[root@openstack ~]# systemctl restart chronyd

[root@openstack ~]# chronyc sources -v

210 Number of sources = 1

.-- Source mode '^' = server, '=' = peer, '#' = local clock.

/ .- Source state '\*' = current synced, '+' = combined , '-' = not combined,

| / '?' = unreachable, 'x' = time may be in error, '~' = time too variable.

|| .- xxxx [ yyyy ] +/- zzzz

|| Reachability register (octal) -. | xxxx = adjusted offset,

|| Log2(Polling interval) --. | | yyyy = measured offset,

|| \ | | zzzz = estimated error.

|| | | \

MS Name/IP address Stratum Poll Reach LastRx Last sample

===============================================================================

^\* gateway 3 6 17 2 +14us[ +38us] +/- 29ms

[root@openstack ~]#

[root@nova ~]# yum -y install chrony

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

[root@nova ~]# cat /etc/chrony.conf | grep -v ^$ | grep -v ^#

server 192.168.1.254 iburst

cmdallow 127.0.0.1

driftfile /var/lib/chrony/drift

makestep 1.0 3

rtcsync

logdir /var/log/chrony

[root@nova ~]# systemctl restart chronyd

[root@nova ~]# chronyc sources -v

210 Number of sources = 1

.-- Source mode '^' = server, '=' = peer, '#' = local clock.

/ .- Source state '\*' = current synced, '+' = combined , '-' = not combined,

| / '?' = unreachable, 'x' = time may be in error, '~' = time too variable.

|| .- xxxx [ yyyy ] +/- zzzz

|| Reachability register (octal) -. | xxxx = adjusted offset,

|| Log2(Polling interval) --. | | yyyy = measured offset,

|| \ | | zzzz = estimated error.

|| | | \

MS Name/IP address Stratum Poll Reach LastRx Last sample

===============================================================================

^\* gateway 3 6 7 2 -2690ns[ -95us] +/- 29ms

[root@nova ~]#

1. 部署OpenStack

8.1、安装OpenStack依赖包

#创建卷组，卷组名必须是cinder-volumes

[root@openstack ~]# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

vda 253:0 0 50G 0 disk

└─vda1 253:1 0 50G 0 part /

vdb 253:16 0 20G 0 disk

[root@openstack ~]# yum -y install lvm2

[root@openstack ~]# pvcreate /dev/vdb

Physical volume "/dev/vdb" successfully created.

[root@openstack ~]# vgcreate cinder-volumes /dev/vdb

Volume group "cinder-volumes" successfully created

[root@openstack ~]# vgs

VG #PV #LV #SN Attr VSize VFree

cinder-volumes 1 0 0 wz--n- <20.00g <20.00g

[root@openstack ~]#

#安装依赖包

[root@openstack ~]# yum -y install qemu-kvm libvirt-client libvirt-daemon libvirt-daemon-driver-qemu python-setuptools

[root@nova ~]# yum -y install qemu-kvm libvirt-client libvirt-daemon libvirt-daemon-driver-qemu python-setuptools

8.2、安装packstack

[root@openstack ~]# yum -y install openstack-packstack

[root@openstack ~]# packstack --gen-answer-file answer.ini

Packstack changed given value to required value /root/.ssh/id\_rsa.pub

[root@openstack ~]# vim answer.ini

11 CONFIG\_DEFAULT\_PASSWORD=redhat

#配置默认密码

42 CONFIG\_SWIFT\_INSTALL=n

#设置禁用swift对象存储

75 CONFIG\_NTP\_SERVERS=192.168.1.254

#NTP服务器地址

98 CONFIG\_COMPUTE\_HOSTS=192.168.1.11

#计算节点的IP地址

102 CONFIG\_NETWORK\_HOSTS=192.168.1.11

#需要配置vxlan网络的IP地址

554 CONFIG\_CINDER\_VOLUMES\_CREATE=n

#禁止自动创建cinder卷组

840 CONFIG\_NEUTRON\_ML2\_TYPE\_DRIVERS=flat,vxlan

#设置网络支持协议

876 CONFIG\_NEUTRON\_ML2\_VXLAN\_GROUP=239.1.1.5

#设置组播地址

910 CONFIG\_NEUTRON\_OVS\_BRIDGE\_MAPPINGS=physnet1:br-ex

#设置虚拟交换机

921 CONFIG\_NEUTRON\_OVS\_BRIDGE\_IFACES=br-ex:eth0

#设置虚拟交换机连接的物理网卡

936 CONFIG\_NEUTRON\_OVS\_TUNNEL\_IF=eth1

#设置隧道网络使用的网卡

1179 CONFIG\_PROVISION\_DEMO=n

#禁止测试案例

#修改10行，其中98和102行不需要修改

8.3、安装OpenStack

[root@openstack ~]# packstack --answer-file=answer.ini

#报错如下

192.168.1.11\_controller.pp: [ ERROR ]

Applying Puppet manifests [ ERROR ]

ERROR : Error appeared during Puppet run: 192.168.1.11\_controller.pp

Error: Could not set 'present' on ensure: Cannot allocate memory - fork(2) at 70:/var/tmp/packstack/82f1096200ae4451834f159c01e6feac/modules/openstacklib/manifests/db/mysql.pp

#原因是虚拟机内存不足，重启虚拟机，还不行的话增加虚拟机内存分配值

##我这里由于是2G内存，所以报不足

#亲测，需要把openstack虚拟机的内存调整到5G

\*\*\*\* Installation completed successfully \*\*\*\*\*\*

Additional information:

\* File /root/keystonerc\_admin has been created on OpenStack client host 192.168.1.11. To use the command line tools you need to source the file.

\* To access the OpenStack Dashboard browse to http://192.168.1.11/dashboard .

Please, find your login credentials stored in the keystonerc\_admin in your home directory.

\* The installation log file is available at: /var/tmp/packstack/20190529-153642-zO4ICx/openstack-setup.log

\* The generated manifests are available at: /var/tmp/packstack/20190529-153642-zO4ICx/manifests

[root@openstack ~]#

##见到此内容，安装完成

[root@openstack ~]# ifconfig

br-ex: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

inet 192.168.1.11 netmask 255.255.255.0 broadcast 192.168.1.255

ether 32:42:b2:02:d7:47 txqueuelen 1000 (Ethernet)

RX packets 2539 bytes 24344871 (23.2 MiB)

RX errors 0 dropped 0 overruns 0 frame 0 TX packets 2428 bytes 185738 (181.3 KiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500

ether 52:54:00:71:e9:78 txqueuelen 1000 (Ethernet)

RX packets 4465 bytes 35174400 (33.5 MiB)

RX errors 0 dropped 210 overruns 0 frame 0

TX packets 3546 bytes 282262 (275.6 KiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

#eth0网卡绑定到br-ex网桥

1. OpenStack基础使用

#修复bug，修改httpd配置文件

[root@openstack ~]# vim /etc/httpd/conf.d/15-horizon\_vhost.conf

[root@openstack ~]# tail -3 /etc/httpd/conf.d/15-horizon\_vhost.conf

WSGIApplicationGroup %{GLOBAL}

WSGIScriptAlias /dashboard "/usr/share/openstack-dashboard/openstack\_dashboard/wsgi/django.wsgi"

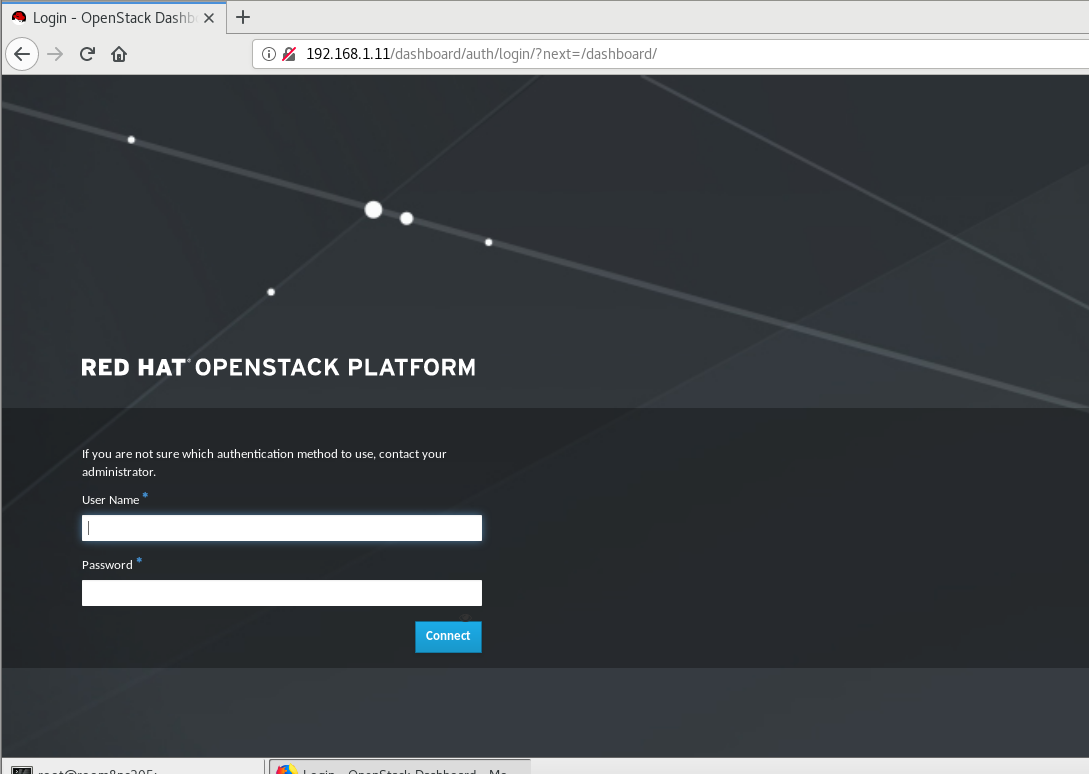
</VirtualHost>

[root@openstack ~]# apachectl graceful

[root@openstack ~]#

#向配置文件中写入标红色的一条，然后重载apache服务

#访问OpenStack的http服务



#见到此页面即为成功

#获取登录密码

[root@openstack ~]# cat keystonerc\_admin

unset OS\_SERVICE\_TOKEN

export OS\_USERNAME=admin

export OS\_PASSWORD=f2d8f2732bd54ba9

export OS\_AUTH\_URL=http://192.168.1.11:5000/v2.0

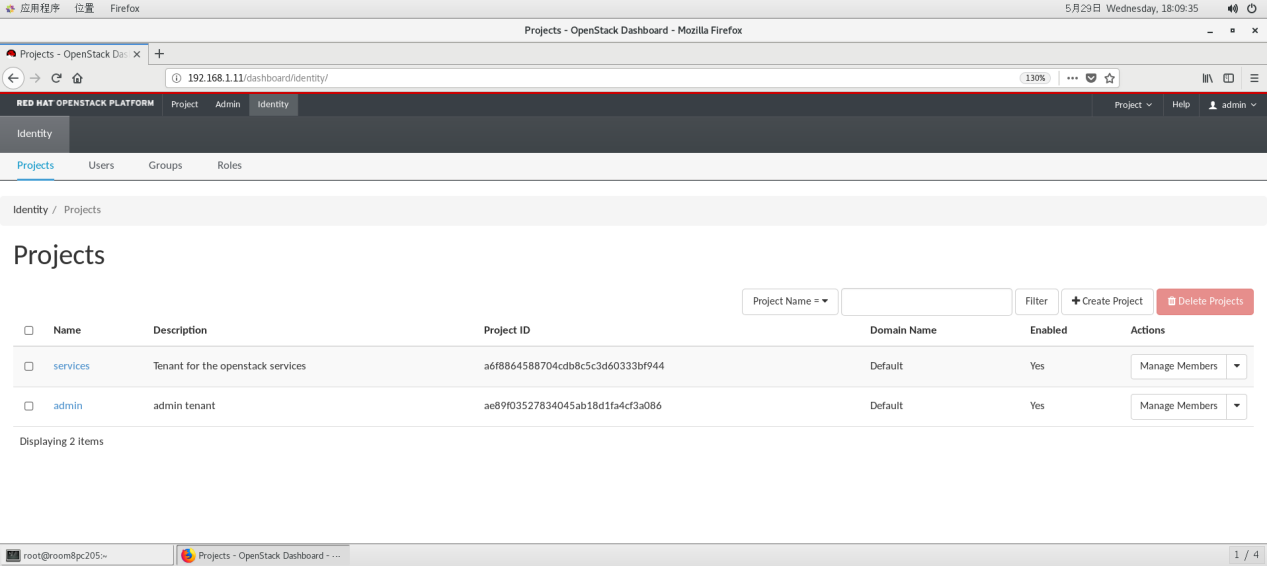
export PS1='[\u@\h \W(keystone\_admin)]\$ '

export OS\_TENANT\_NAME=admin

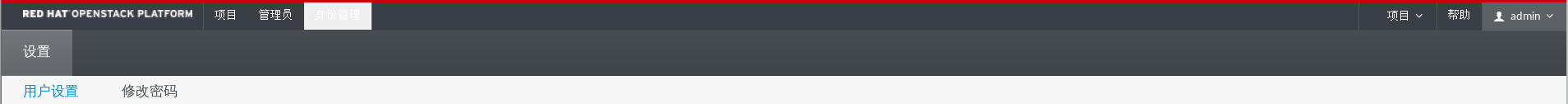
export OS\_REGION\_NAME=RegionOne

#用户名admin，密码是加密的那一串

#输入用户名密码，登录到管理页面



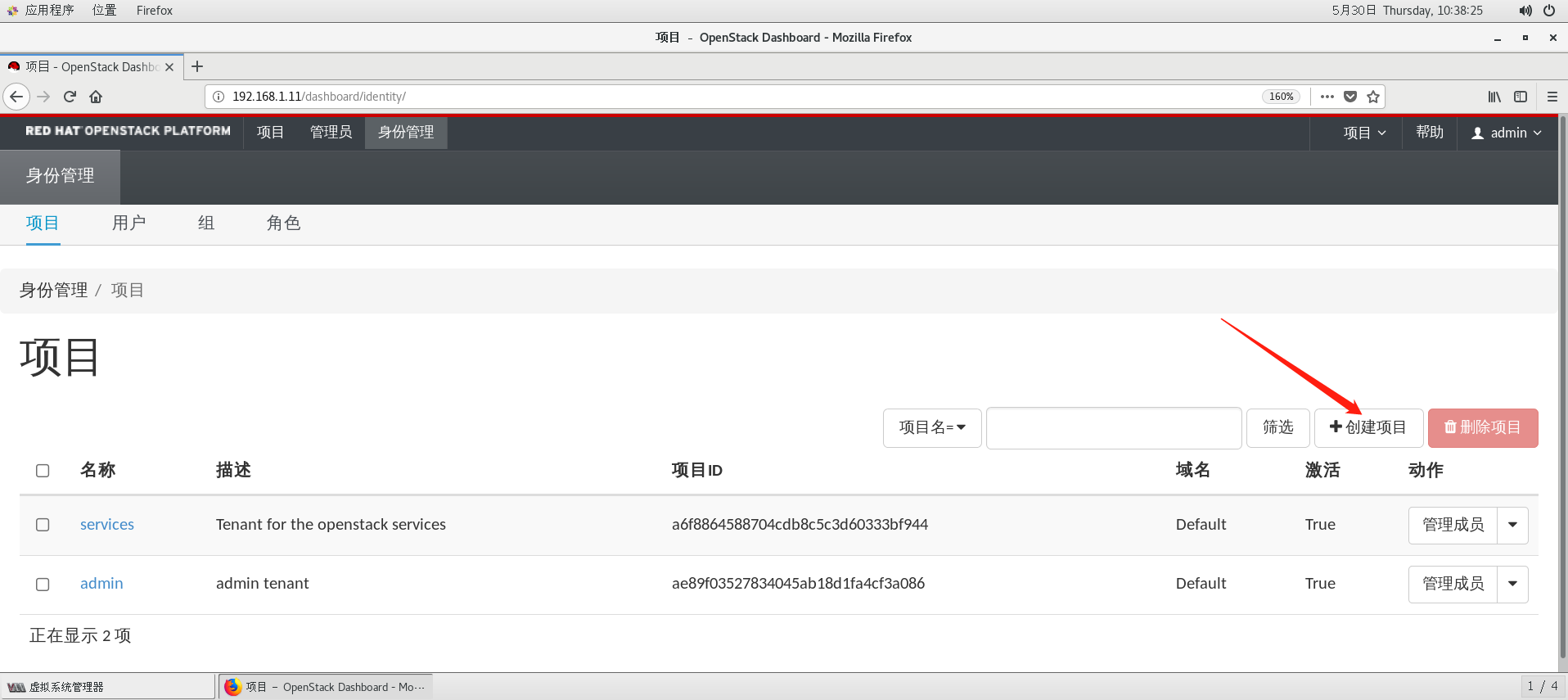
#修改界面语言，点击右上角admin，settings，languages，选择中文，然后点击save保存

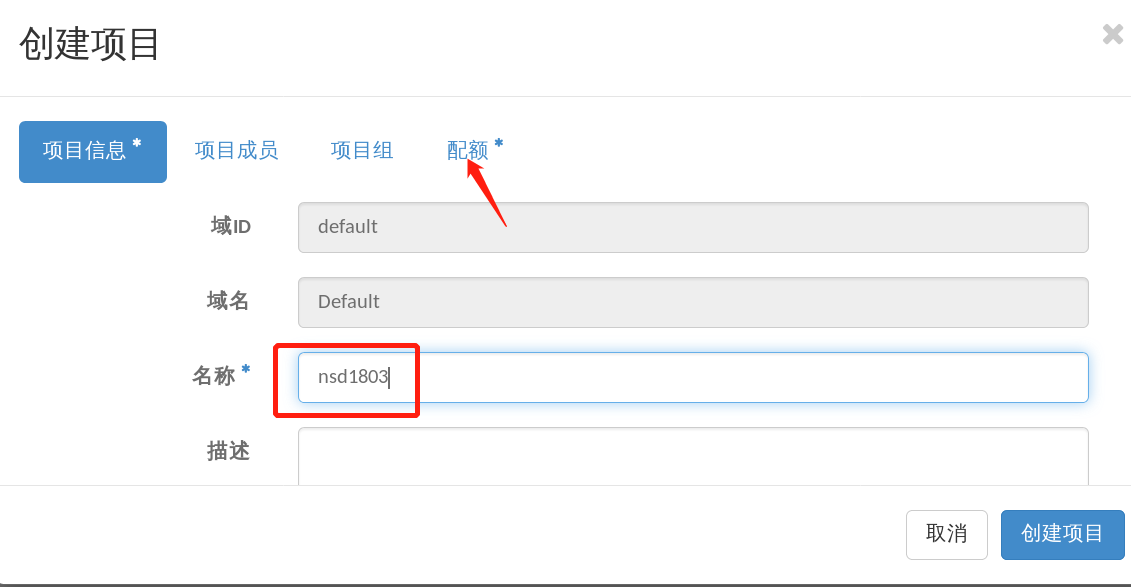


#界面语言变成中文

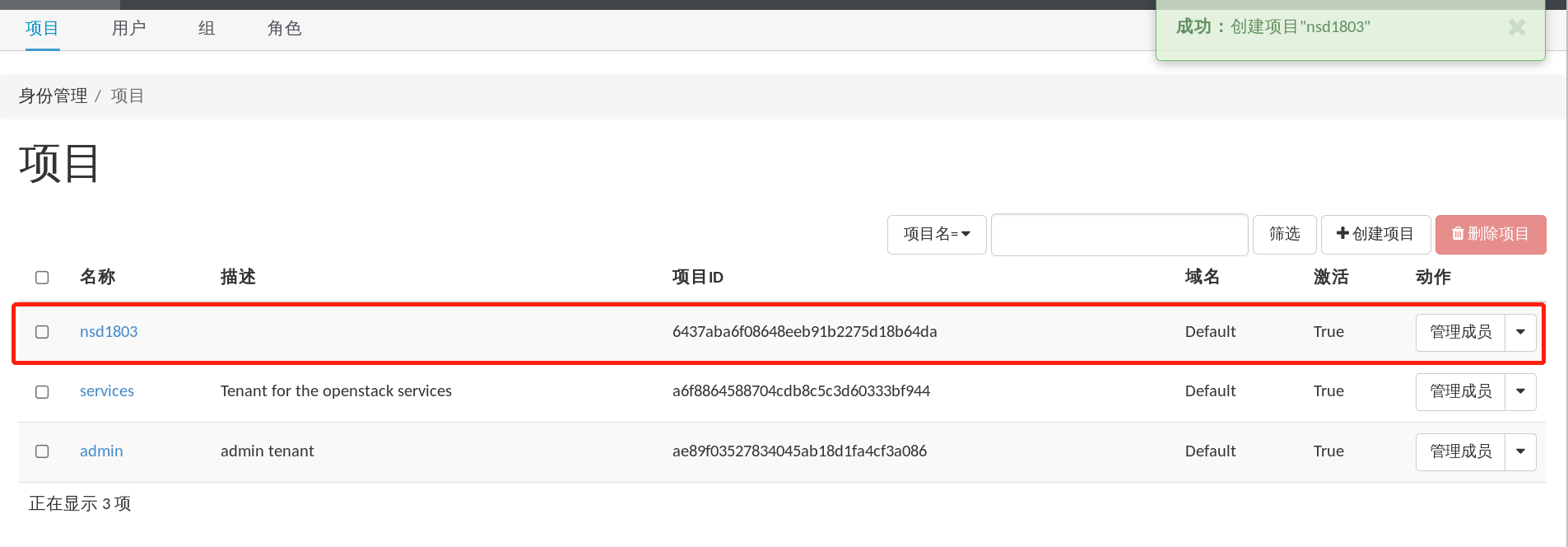
##项目管理

#创建项目，带\*的是必须设置的条目





#点击到配额页面，（我这里由于内存小，所以把内存修改的小一点）

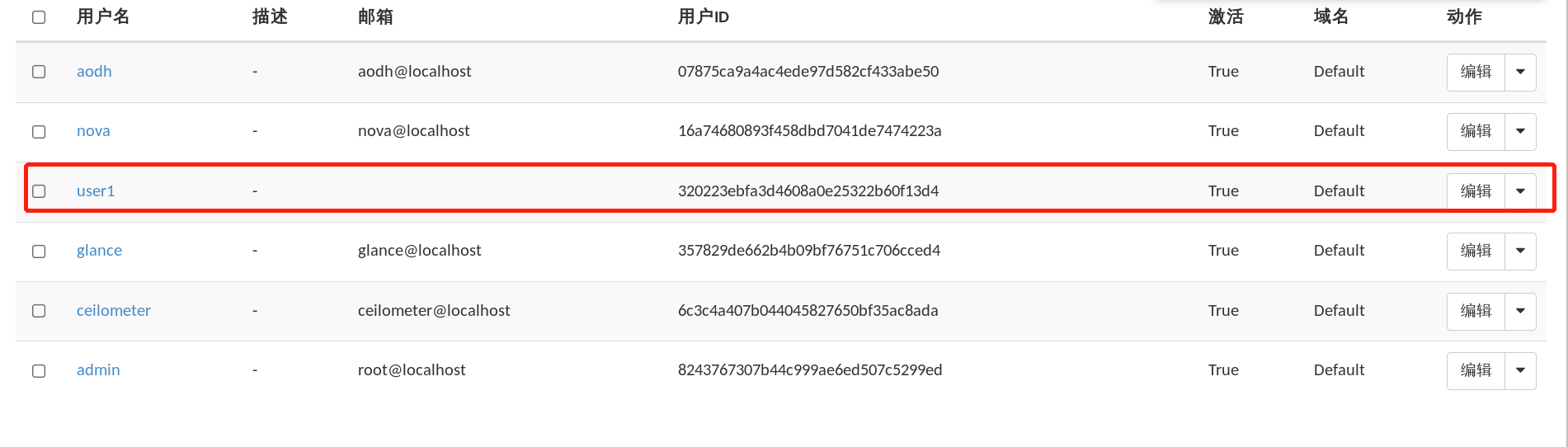


#创建出nsd1803项目

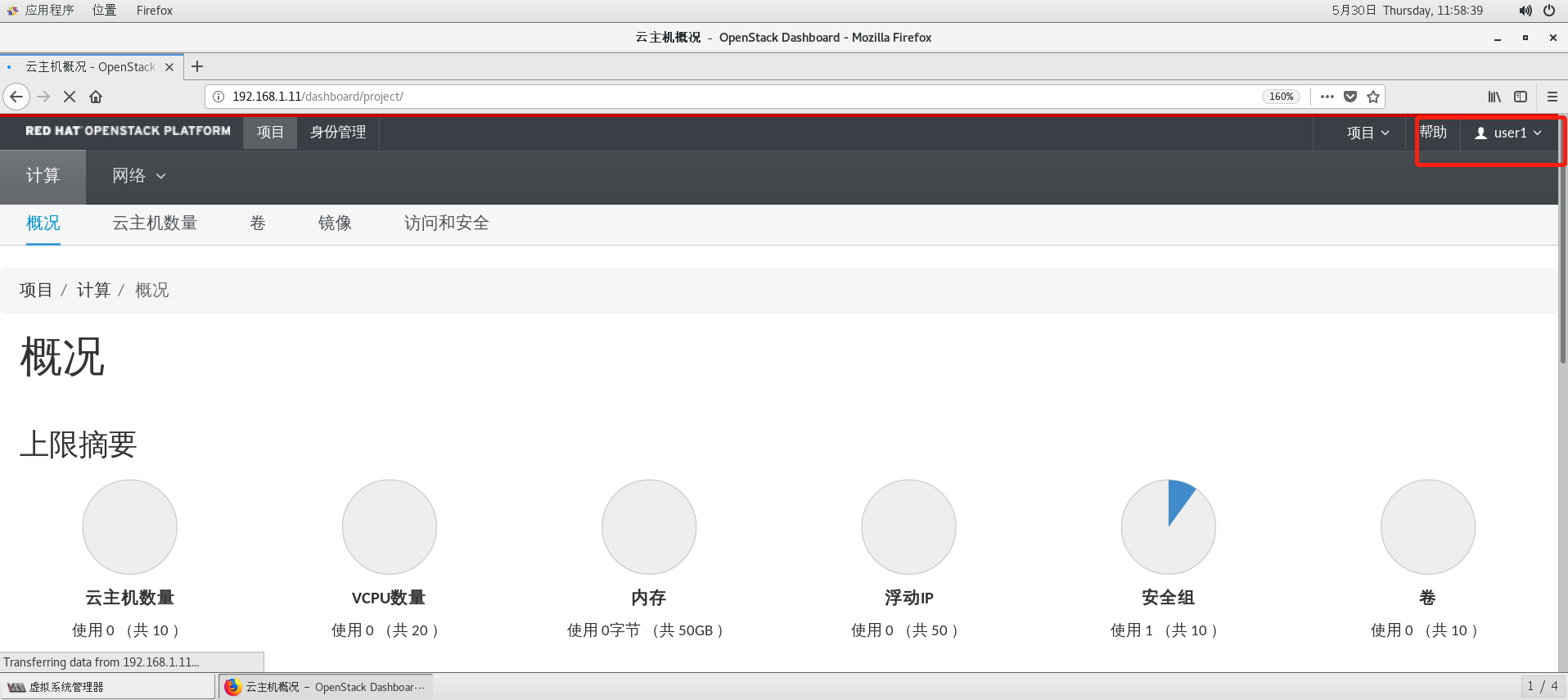
#创建项目用户，注意不要修改已有的用户，不然会造成无法通信



#用户名user1，密码123，项目设置nsd1803，角色为member



#user1用户创建成功，然后web页面点击右上角，退出admin用户，用user1用户登录



#成功登录user1用户管理界面

#命令行控制项目，用户，测试命令

[root@openstack ~]# source ~/keystonerc\_admin

[root@openstack ~(keystone\_admin)]# env | grep OS

HOSTNAME=openstack

OS\_REGION\_NAME=RegionOne

OS\_PASSWORD=f2d8f2732bd54ba9

OS\_AUTH\_URL=http://192.168.1.11:5000/v2.0

OS\_USERNAME=admin

OS\_TENANT\_NAME=admin

[root@openstack ~(keystone\_admin)]# openstack project list

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

| ID | Name |

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

| 6437aba6f08648eeb91b2275d18b64da | nsd1803 |

| a6f8864588704cdb8c5c3d60333bf944 | services |

| ae89f03527834045ab18d1fa4cf3a086 | admin |

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

[root@openstack ~(keystone\_admin)]# openstack user list

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

| ID | Name |

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

| 8243767307b44c999ae6ed507c5299ed | admin |

| 9fa4187c16b74c048509fdec59936c63 | neutron |

| bb091e0f45da47dd8bf45d969ab1a760 | gnocchi |

| 07875ca9a4ac4ede97d582cf433abe50 | aodh |

| 16a74680893f458dbd7041de7474223a | nova |

| 357829de662b4b09bf76751c706cced4 | glance |

| 6c3c4a407b044045827650bf35ac8ada | ceilometer |

| 9e02c5a21b4d4c13a4d10bb3d5c06959 | cinder |

| 320223ebfa3d4608a0e25322b60f13d4 | user1 |

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

[root@openstack ~(keystone\_admin)]# nova quota-defaults

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

| Quota | Limit |

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

| instances | 10 |

| cores | 20 |

| ram | 51200 |

| metadata\_items | 128 |

| injected\_files | 5 |

| injected\_file\_content\_bytes | 10240 |

| injected\_file\_path\_bytes | 255 |

| key\_pairs | 100 |

| server\_groups | 10 |

| server\_group\_members | 10 |

#不能直接在keystone文件里修改密码，可以从web页面上修改，修改完成以后修改keystone文件

#安装完成后，如果没有修改过keystone密码，可以通过在answer应答文件里边，找到默认安装密码323 CONFIG\_KEYSTONE\_ADMIN\_PW=f2d8f2732bd54ba9

##用户配额管理

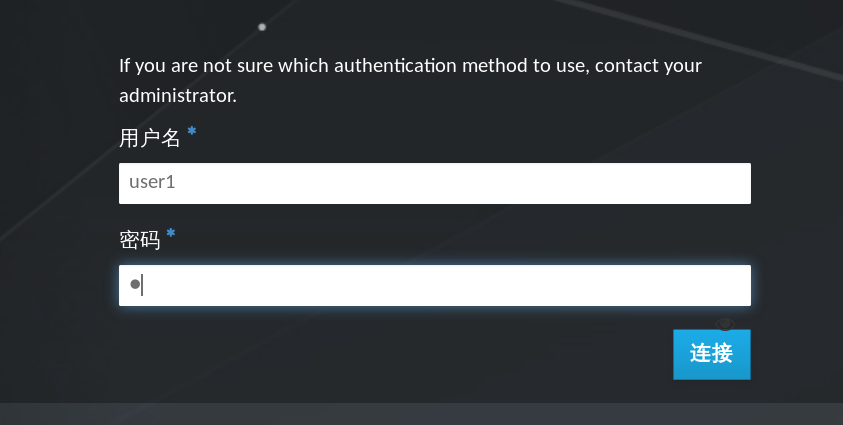
#修改用户密码，用admin用户修改项目用户密码，跟Linux的root用户一样，直接修改

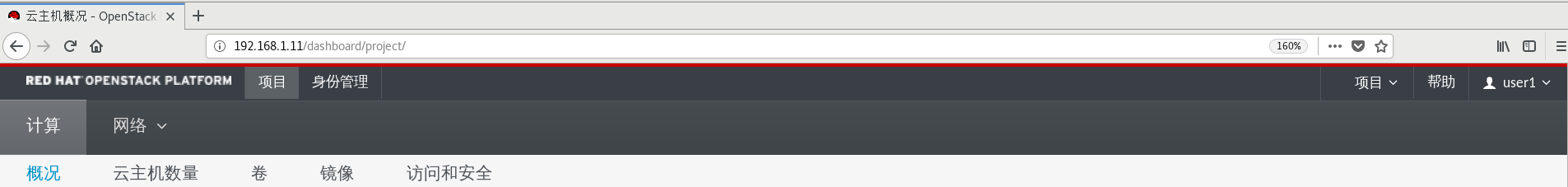
[root@openstack ~]# source keystonerc\_admin

[root@openstack ~(keystone\_admin)]# openstack user set --password a user1

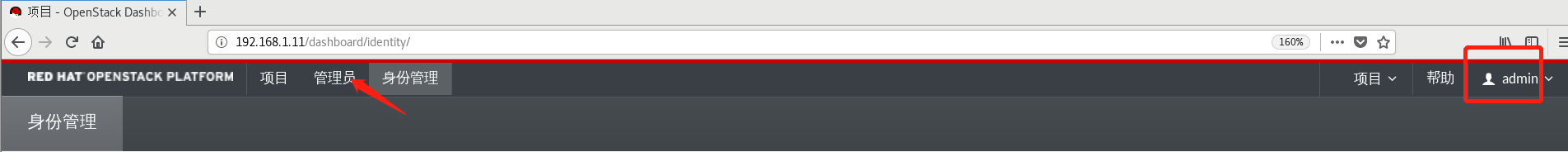
[root@openstack ~(keystone\_admin)]#

#将user1用户的密码从123修改成a

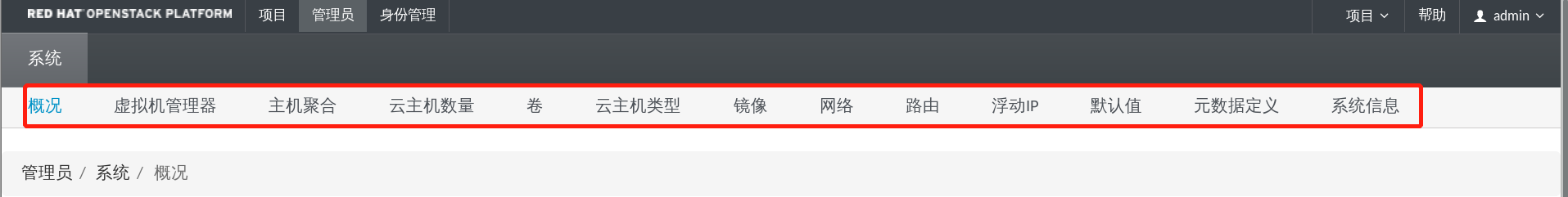




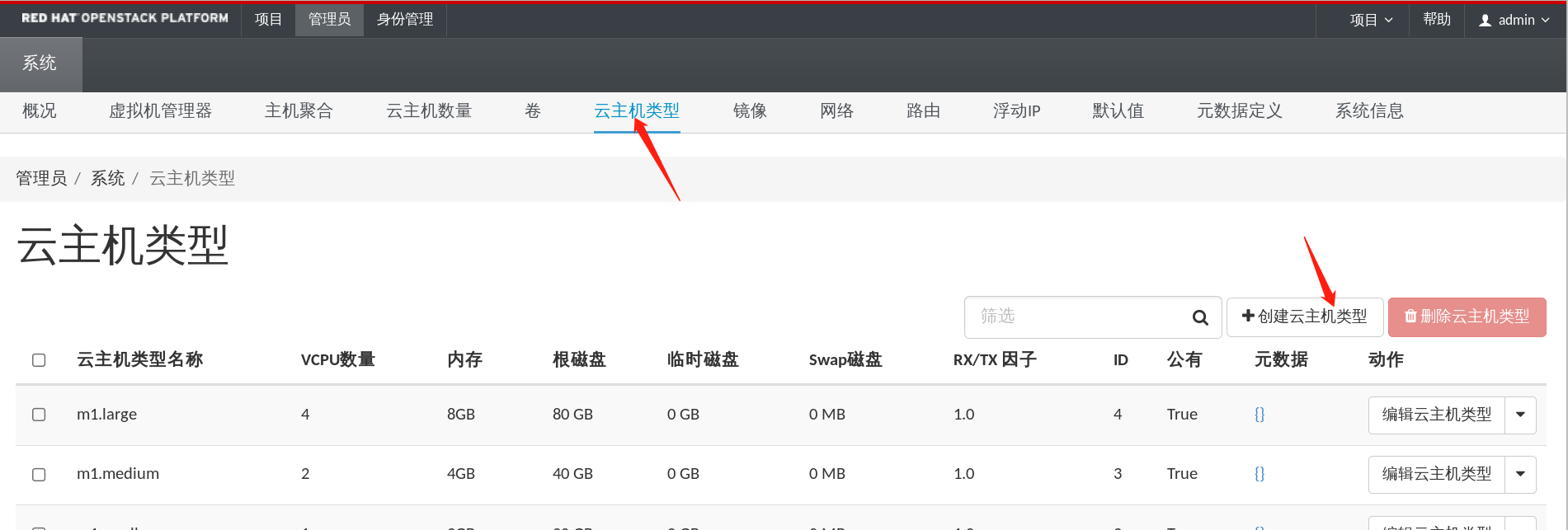
#退出user1用户，用admin登录管理页面



#设置条目



#点击云主机类型，自定义云主机类型（类似于修改virt-manager的xml文件）

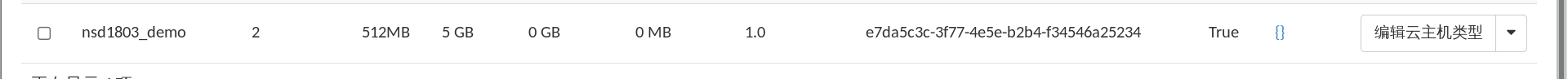


#填写带\*的条目，其他默认





#然后点击创建云主机类型，右上角弹出成功，面板中出现自定义的类型



#镜像管理（virt-manager的后端磁盘）

#准备镜像，官网下载small.img

[root@openstack ~(keystone\_admin)]# qemu-img info small.img

image: small.img

file format: qcow2

virtual size: 1.0G (1073741824 bytes)

disk size: 102M

cluster\_size: 65536

Format specific information:

compat: 1.1

lazy refcounts: false

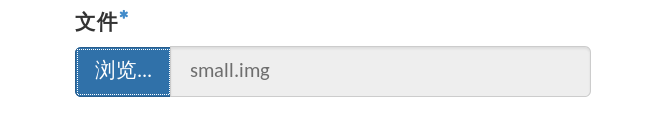
refcount bits: 16

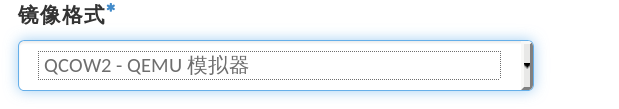
corrupt: false

#导入镜像



镜像名称可随意

该文件从真机选择

类型一定选择正确

#点击创建镜像，等待small.img上传，等待30s后手动刷新页面，如下即成功



#可用命令行验证镜像状态

[root@openstack ~(keystone\_admin)]# openstack image list

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

| ID | Name | Status |

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

| 21bafc84-58d7-4cc4-9e36-658c0e276f85 | small | active |

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

[root@openstack ~(keystone\_admin)]#

#备份镜像

[root@openstack ~(keystone\_admin)]# openstack image save --file /tmp/rhel6.img small

[root@openstack ~(keystone\_admin)]# ls /tmp/rhel6.img

/tmp/rhel6.img

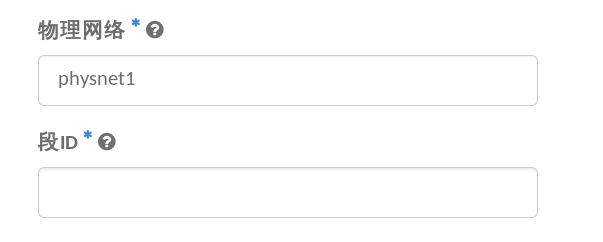
[root@openstack ~(keystone\_admin)]#

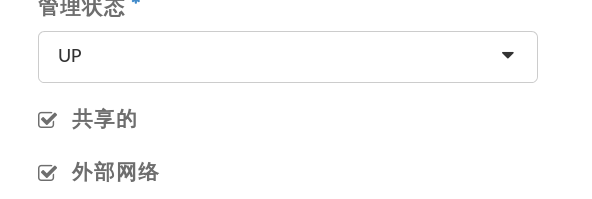
#网络管理

##创建外部网络

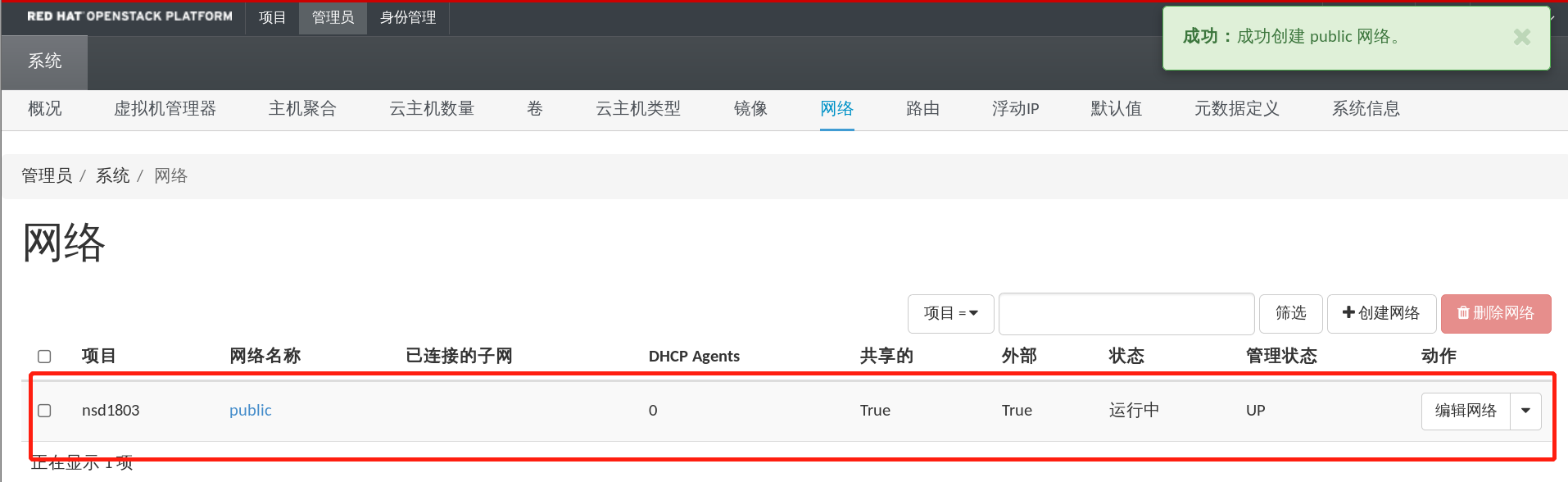




段ID自动生成

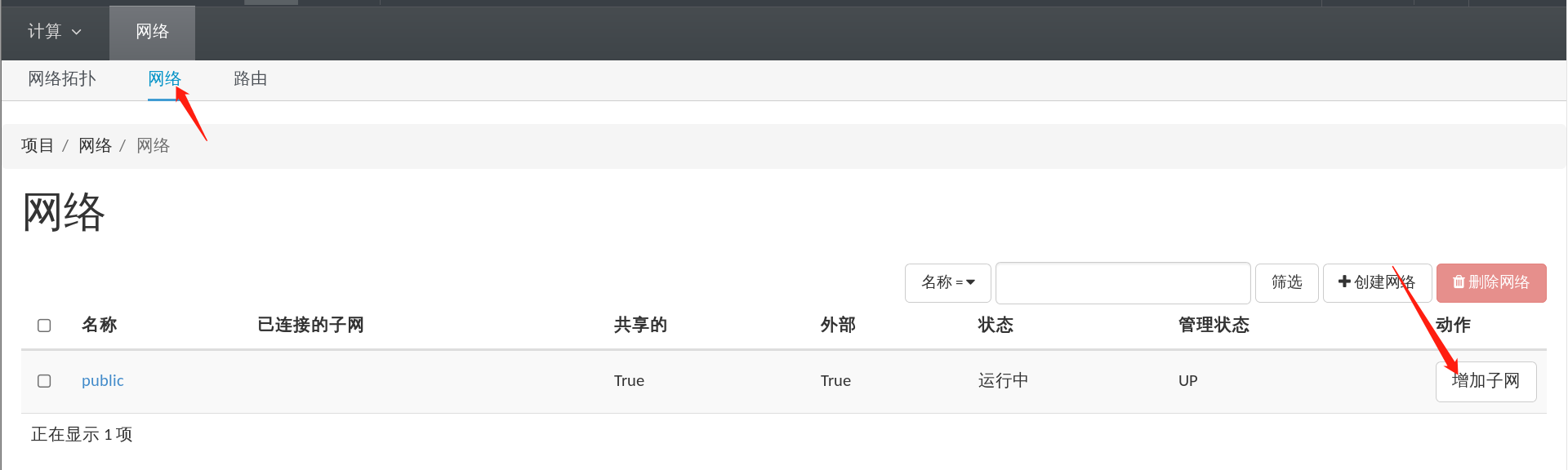


#点击提交，创建外部网络

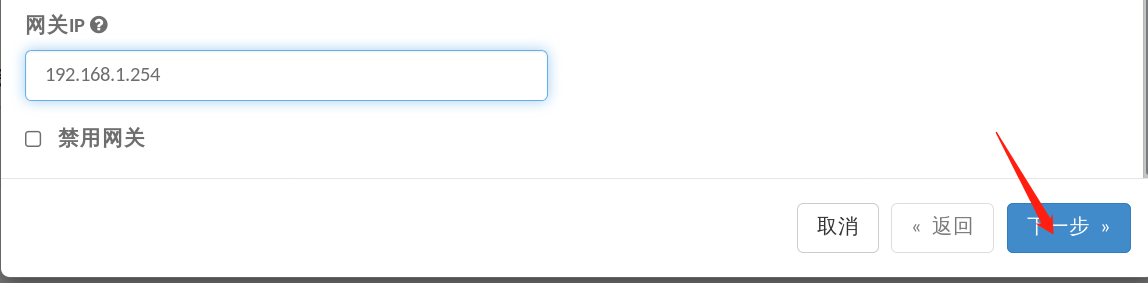


#见到此条目，外部网络创建完毕，退出admin用户，登录user1，创建内部网络









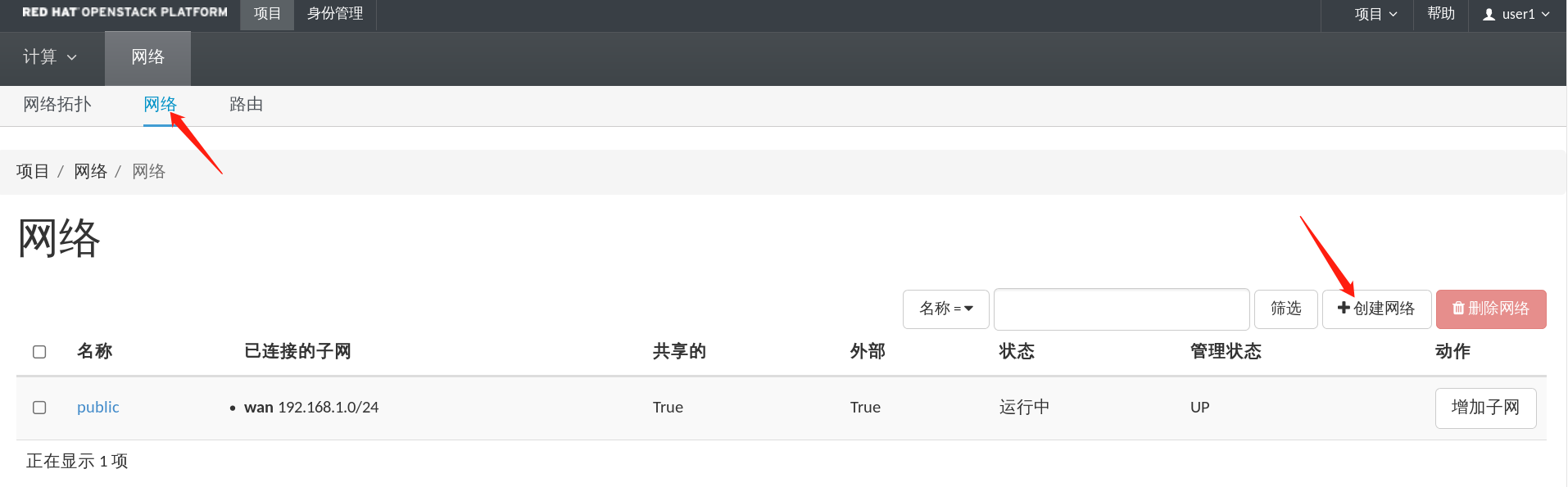
#取消勾选激活dhcp，然后点击创建





#至此，public网段创建完成

#然后点击网络，创建网络

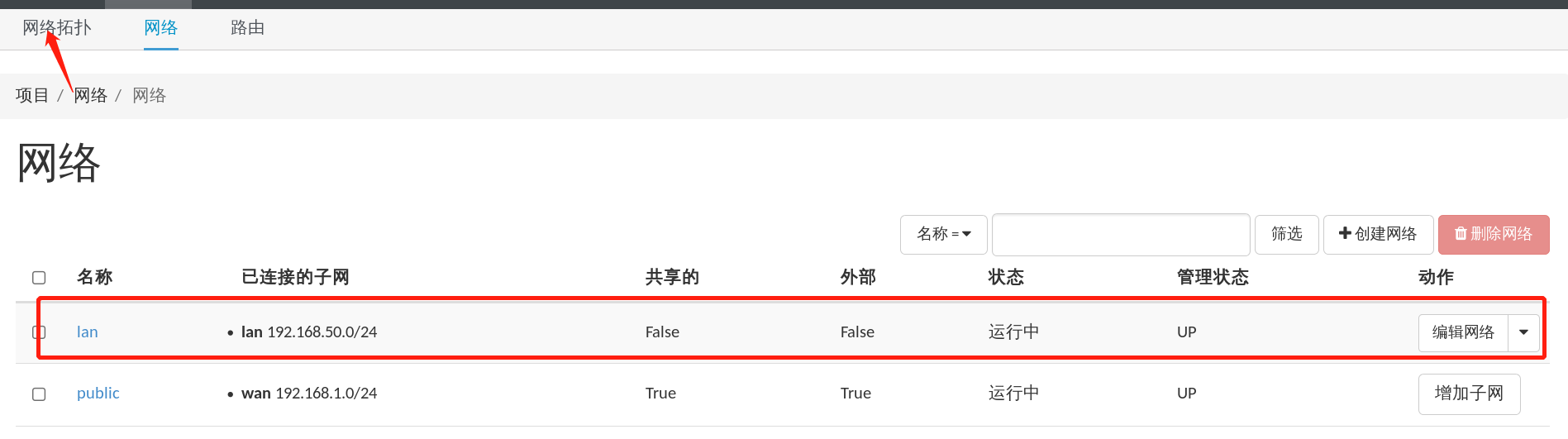








#点击已创建，



#然后点击网络拓扑



#然后点击路由

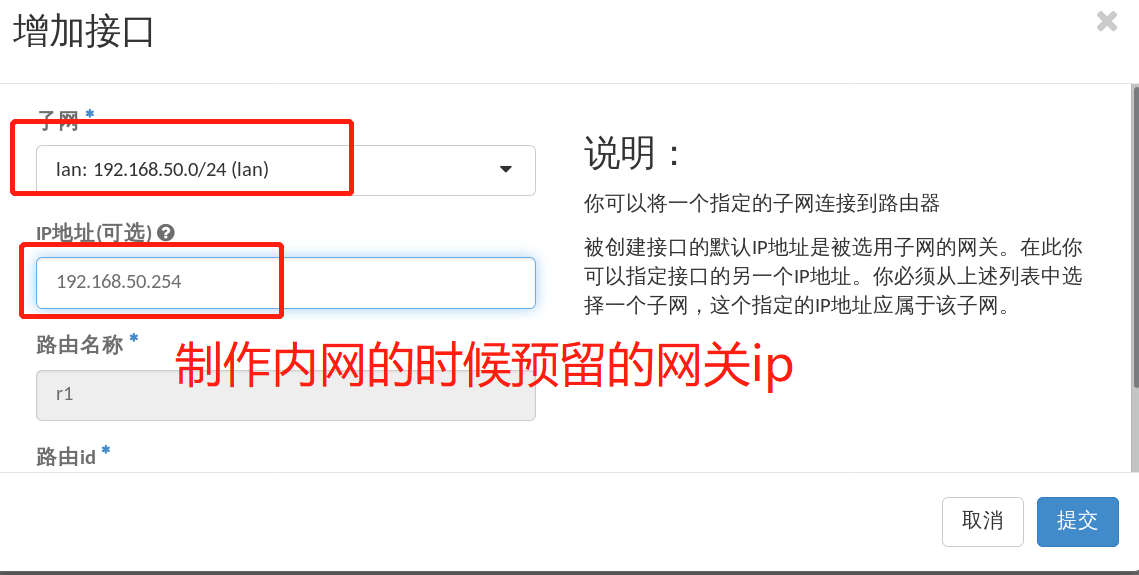






#外部路由设置成功，进入r1内部，设置内部路由







#点击网络拓扑，查看内外网络拓扑图，已经连接成功

##云主机管理

#创建云主机





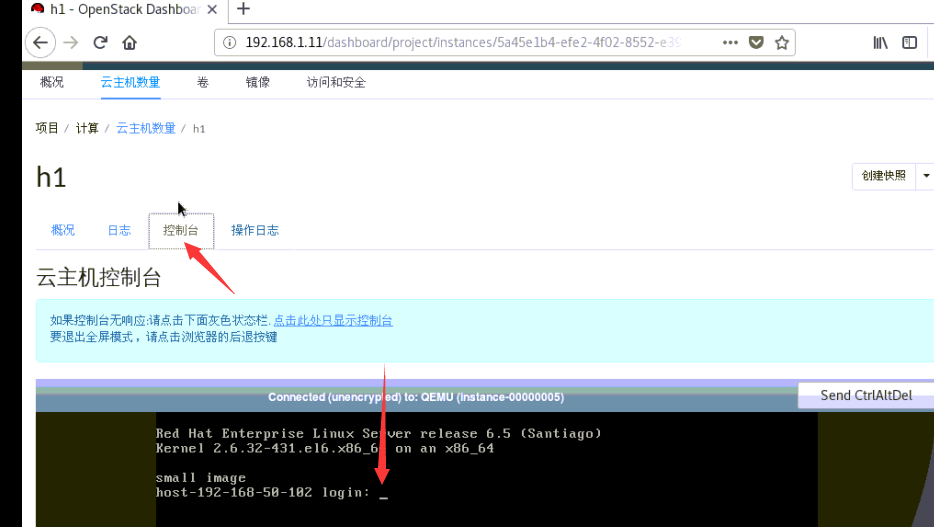




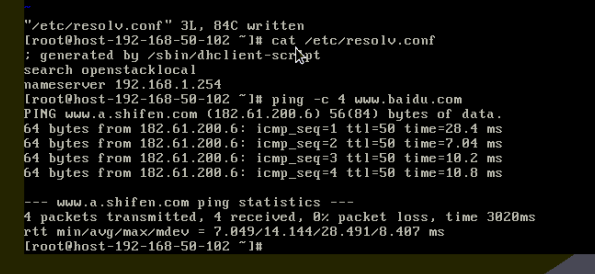
#点击创建云主机



#点击h1，进入云主机控制页面



#点击控制台，见到Linux命令行，输入用户名root，密码redhat



#登录控制台，修改resolv文件的nameserver地址，然后测试ping百度，接入外网

#如果此处报错，可以尝试登录admin用户，把云主机类型调整到最小配置

#再创建一台云主机







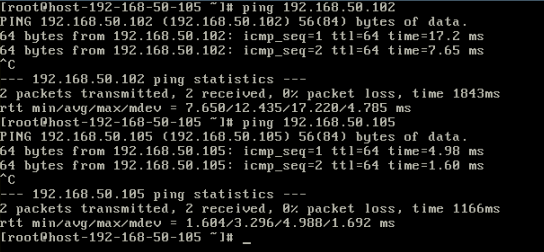


#点击创建云主机

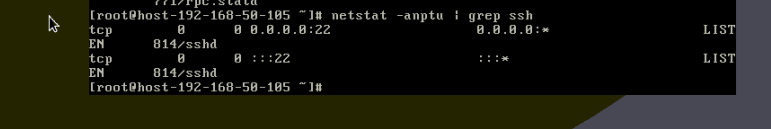


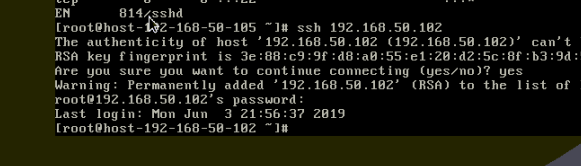
#创建出两个云主机，登录虚拟机控制台，测试两个云主机均可用

#登录h2云主机的控制台



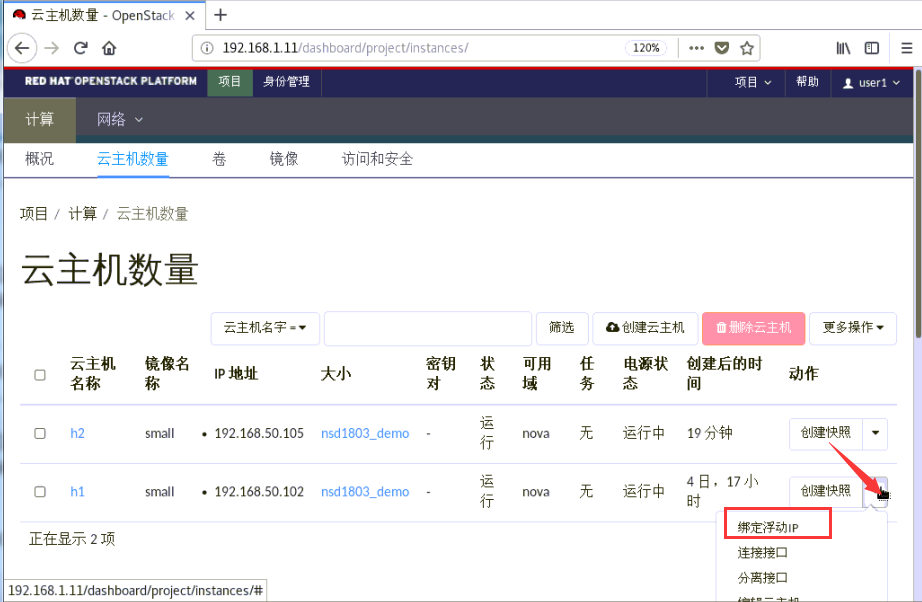
#云主机之间可以互相通信

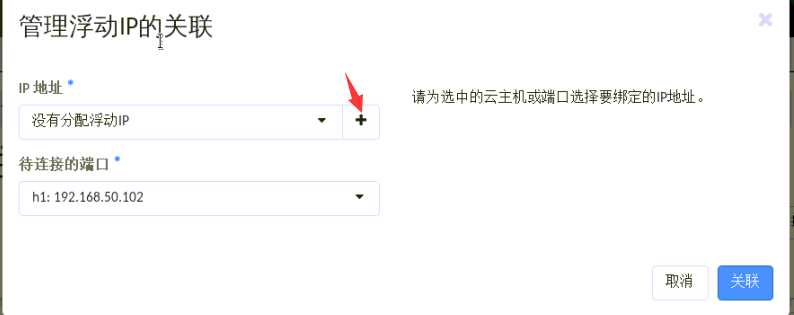




#且云主机已经开启sshd服务，可以在内部网络执行ssh远程管理

##将云主机发布到公网



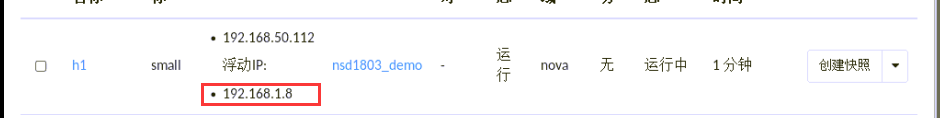






#注意，千万不要跟OpenStack这台机器的br-ex网桥的ip冲突，然后点击关联

#我这里虚拟机死机了一下，删除到原有云主机h1，重新创建



#成功分配192.168.1.8这个ip地址，此时还不能通过公网来访问这个云主机，需要设置安全组

#创建安全组







#去掉ipv6规则



#添加ping的规则



[root@openstack ~]# ping 192.168.1.8

PING 192.168.1.8 (192.168.1.8) 56(84) bytes of data.

^C

--- 192.168.1.8 ping statistics ---

4 packets transmitted, 0 received, 100% packet loss, time 3003ms

#此时还不能ping通1.8这个ip地址

#修改安全组规则





#左边是安全组仓库，右边是云主机应用，点击保存

[root@openstack ~]# ping 192.168.1.8

PING 192.168.1.8 (192.168.1.8) 56(84) bytes of data.

64 bytes from 192.168.1.8: icmp\_seq=1 ttl=63 time=14.8 ms

64 bytes from 192.168.1.8: icmp\_seq=2 ttl=63 time=1.52 ms

^C

--- 192.168.1.8 ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1001ms

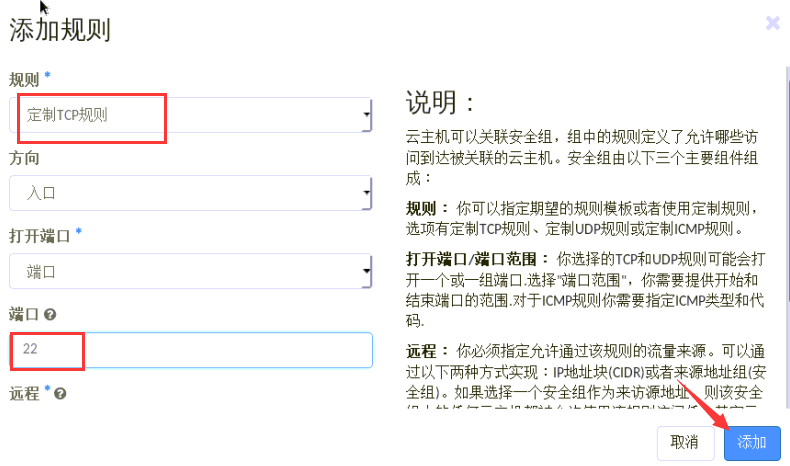
rtt min/avg/max/mdev = 1.523/8.202/14.881/6.679 ms

#此时1.8这个ip地址已经可以ping通

[root@openstack ~]# ssh 192.168.1.8

ssh: connect to host 192.168.1.8 port 22: No route to host

#此时还不能ssh，继续修改安全组规则





[root@openstack ~]# ssh 192.168.1.8

root@192.168.1.8's password:

[root@host-192-168-50-112 ~]# ifconfig eth0 | head -2

eth0 Link encap:Ethernet HWaddr FA:16:3E:B1:49:09

inet addr:192.168.50.112 Bcast:192.168.50.255 Mask:255.255.255.0

[root@host-192-168-50-112 ~]#

#已经可以ssh远程到这台云主机

#设置通用规则







#此时这台云主机已经开放到公网，裸奔中

[root@room8pc205 ~]# ping 192.168.1.8

PING 192.168.1.8 (192.168.1.8) 56(84) bytes of data.

64 bytes from 192.168.1.8: icmp\_seq=1 ttl=63 time=1.66 ms

^C

--- 192.168.1.8 ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 1.669/1.669/1.669/0.000 ms

[root@room8pc205 ~]# ssh 192.168.1.8

The authenticity of host '192.168.1.8 (192.168.1.8)' can't be established.

RSA key fingerprint is SHA256:hr5EbbrtCFv0XfFCRunl0fMTpr8ade6DhpQfDX29/f4.

RSA key fingerprint is MD5:e9:90:c7:53:4a:24:63:1d:1d:25:48:a2:f9:f2:2c:19.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '192.168.1.8' (RSA) to the list of known hosts.

root@192.168.1.8's password:

Last login: Mon Jun 3 23:28:51 2019 from 192.168.1.11

[root@host-192-168-50-112 ~]# ifconfig | head -2

eth0 Link encap:Ethernet HWaddr FA:16:3E:B1:49:09

inet addr:192.168.50.112 Bcast:192.168.50.255 Mask:255.255.255.0

##nova使用

#检测OpenStack主机和Nova主机互相通信，两个网卡都测试

[root@openstack ~]# ping -c 2 192.168.1.12

PING 192.168.1.12 (192.168.1.12) 56(84) bytes of data.

64 bytes from 192.168.1.12: icmp\_seq=1 ttl=255 time=1.37 ms

64 bytes from 192.168.1.12: icmp\_seq=2 ttl=255 time=0.689 ms

--- 192.168.1.12 ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1001ms

rtt min/avg/max/mdev = 0.689/1.031/1.373/0.342 ms

[root@openstack ~]# ping -c 2 192.168.4.12

PING 192.168.4.12 (192.168.4.12) 56(84) bytes of data.

64 bytes from 192.168.4.12: icmp\_seq=1 ttl=255 time=1.18 ms

64 bytes from 192.168.4.12: icmp\_seq=2 ttl=255 time=0.851 ms

--- 192.168.4.12 ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1001ms

rtt min/avg/max/mdev = 0.851/1.015/1.180/0.167 ms

[root@openstack ~]#

#检测nova主机状态

[root@nova ~]# sestatus

SELinux status: disabled

[root@nova ~]# rpm -qa | grep firewall

[root@nova ~]# rpm -qa | grep Network

[root@nova ~]# yum repolist | tail -1

repolist: 10,731

[root@nova ~]# chronyc sources -v | tail -1

^\* gateway 3 6 377 9 +11us[-8754ns] +/- 55ms

[root@nova ~]# nslookup www.baidu.com 192.168.1.254

Server: 192.168.1.254

Address: 192.168.1.254#53

Non-authoritative answer:

www.baidu.com canonical name = www.a.shifen.com.

Name: www.a.shifen.com

Address: 182.61.200.7

Name: www.a.shifen.com

Address: 182.61.200.6

[root@nova ~]# ping -c 2 openstack

PING openstack (192.168.1.11) 56(84) bytes of data.

64 bytes from openstack (192.168.1.11): icmp\_seq=1 ttl=255 time=1.37 ms

64 bytes from openstack (192.168.1.11): icmp\_seq=2 ttl=255 time=0.492 ms

--- openstack ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1002ms

rtt min/avg/max/mdev = 0.492/0.935/1.379/0.444 ms

[root@nova ~]#

#在openstack主机修改answer.ini文件

[root@openstack ~]# vim answer.ini

[root@openstack ~]# sed -n '98p;102p' answer.ini

CONFIG\_COMPUTE\_HOSTS=192.168.1.11,192.168.1.12

CONFIG\_NETWORK\_HOSTS=192.168.1.11,192.168.1.12

#重新执行安装命令

[root@openstack ~]# packstack --answer-file=answer.ini

#输入nova主机的root密码，然后自动部署

#如果中途报错内存不够，可以先删除掉之前创建的h1和h2云主机

Finalizing [ DONE ]

\*\*\*\* Installation completed successfully \*\*\*\*\*\*

见到此字样的情况，则安装成功

#修改apache配置文件

[root@openstack ~]# vim /etc/httpd/conf.d/15-horizon\_vhost.conf

[root@openstack ~]# cat /etc/httpd/conf.d/15-horizon\_vhost.conf | tail -3

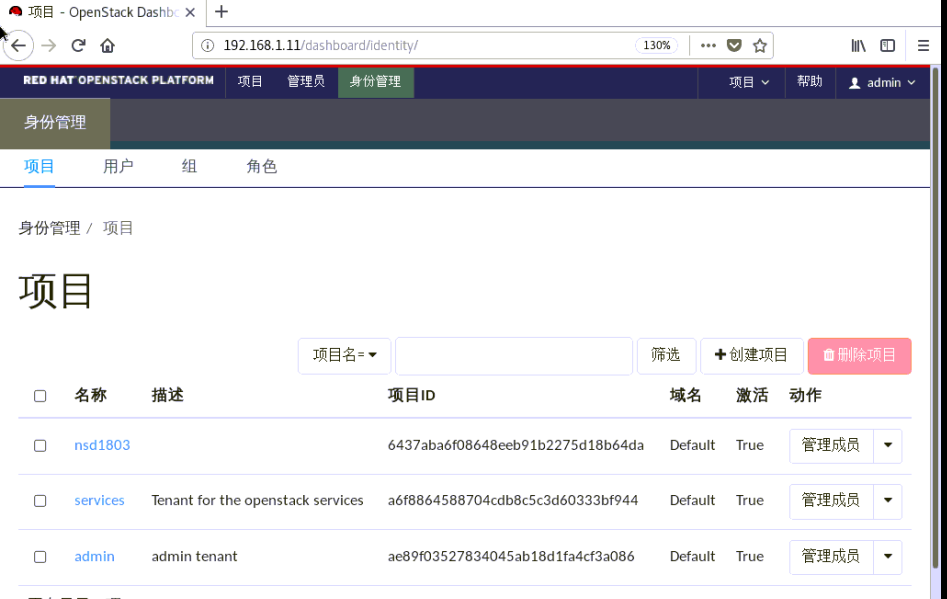
WSGIApplicationGroup %{GLOBAL}

WSGIScriptAlias /dashboard "/usr/share/openstack-dashboard/openstack\_dashboard/wsgi/django.wsgi"

</VirtualHost>

[root@openstack ~]# apachectl graceful

#修改配置文件加入标红的行，完成后重载配置文件



#openstack页面可以重新访问

##数据管理主机热迁移



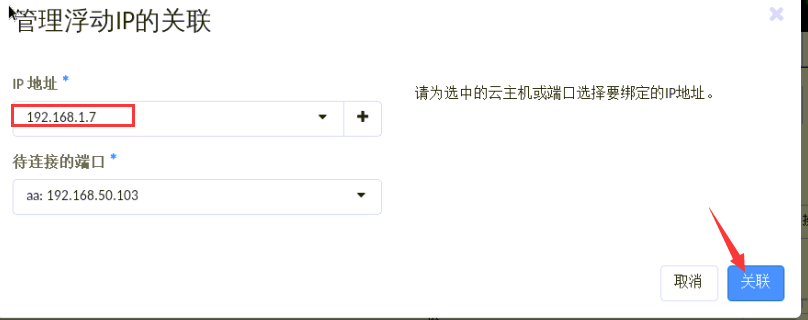
#可以看到两个节点

#退出admin用户，登录user1用户，创建一台云主机

#相对之前的步骤，创建云主机的时候直接设置好free安全组



#绑定浮动ip





#测试浮动ip的安全组

[root@room8pc205 ~]# ping -c 2 192.168.1.7

PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.

64 bytes from 192.168.1.7: icmp\_seq=1 ttl=63 time=15.2 ms

64 bytes from 192.168.1.7: icmp\_seq=2 ttl=63 time=1.03 ms

--- 192.168.1.7 ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1000ms

rtt min/avg/max/mdev = 1.033/8.151/15.270/7.119 ms

[root@room8pc205 ~]# ssh 192.168.1.7

The authenticity of host '192.168.1.7 (192.168.1.7)' can't be established.

RSA key fingerprint is SHA256:Isu3olBHuijQT7ZKrZ3nOFF8hjuMgOs/zn1QPZ292gQ.

RSA key fingerprint is MD5:78:5f:65:a1:aa:95:d3:06:96:69:09:8f:10:c6:16:d4.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '192.168.1.7' (RSA) to the list of known hosts.

root@192.168.1.7's password:

[root@host-192-168-50-103 ~]# ifconfig eth0 | head -2

eth0 Link encap:Ethernet HWaddr FA:16:3E:6E:C4:F1

inet addr:192.168.50.103 Bcast:192.168.50.255 Mask:255.255.255.0

[root@host-192-168-50-103 ~]#

#退出user1用户，用admin用户登录



#可以查看到刚刚user1用户创建的aa云主机，且该云主机在openstack这台机器上

[root@openstack ~]# virsh list

Id 名称 状态

----------------------------------------------------

3 instance-0000000a running

[root@nova ~]# virsh list

Id 名称 状态

----------------------------------------------------

[root@nova ~]#

#热迁移云主机





#在此过程中，可以开一个终端，一直ping云主机的浮动ip，检查是否有中断



#页面显示迁移成功

#命令行检测

[root@openstack ~]# virsh list --all

Id 名称 状态

----------------------------------------------------

[root@openstack ~]#

[root@nova ~]# virsh list --all

Id 名称 状态

----------------------------------------------------

1 instance-0000000a running

[root@nova ~]#

#aa云主机实例已经迁移到nova主机