

# **TTS 10.0 COOKBOOK**

## **( NSD CLOUD DAY02 )**

版本编号 10.0

2018-08

达内 IT 培训集团

## NSD CLOUD DAY02

### 1. 案例 1：配置 yum 仓库

#### • 问题

本案例要求把三个镜像配置 yum 源：

- CentOS7-1708 光盘内容作为仓库源
- 配置 RHEL7-extras 内容加入仓库源
- RHEL7OSP-10 光盘中包含多个目录，每个目录都是仓库源（可以使用脚本生成）

#### • 方案

此实验的整体方案需要三台机器，openstack 作为主节点，nova 作为额外节点，DNSNTP 做为 DNS 和 NTP 的服务器，提供域名解析和时间同步服务，具体情况如表-1 所示：

U-1		
主机名	内存大小	IP
openstack.tedu.cn	8G	192.168.1.1
nova.tedu.cn	6G	192.168.1.2
DNSNTP	2G	192.168.1.3

#### • 步骤

实现此案例需要按照如下步骤进行。

##### 步骤一：配置 yum 仓库

1) 使用模板文件创建三台新的虚拟机，由于昨天已经讲过怎样用模板文件创建虚拟机，这里不再列出步骤

2) 配置 CentOS7-1708，RHEL7-extras，RHEL7OSP-10 的 yum 源（共 12 个，真机操作）

**注意：由于 yum 配置的第一个源为 gpgcheck=1 需要导入公钥，但昨天做的虚拟机模板公钥已经导入，这里不再重复导入公钥**

```
[root@zrj ~]# mkdir /var/ftp/system
[root@zrj ~]# mkdir /var/ftp/extras
[root@zrj ~]# mkdir /var/ftp/HEL7OSP
[root@zrj ~]# mount CentOS7-1708.iso /var/ftp/system
[root@zrj ~]# mount RHEL7-extras.iso /var/ftp/extras
[root@zrj ~]# mount RHEL7OSP-10.iso /var/ftp/HEL7OSP
[root@zrj ~]# vim /etc/yum.repos.d/local.repo
[local_repo]
name=CentOS-$releasever - Base
```

```
baseurl="ftp://192.168.1.254/system"
enabled=1
gpgcheck=1

[local_extras]
name=extras
baseurl="ftp://192.168.1.254/extras"
enabled=1
gpgcheck=0

[1local_devtools-rpms]
name=devtools-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-openstack-10-devtools-rpms"
enabled=1
gpgcheck=0

[2local_optools-rpms]
name=optools-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-openstack-10-optools-rpms"
enabled=1
gpgcheck=0

[3local_rpms]
name=rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-openstack-10-rpms"
enabled=1
gpgcheck=0

[4local_tools-rpms]
name=tools-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-openstack-10-tools-rpms"
enabled=1
gpgcheck=0

[5local_mon-rpms]
name=mon-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhceph-2-mon-rpms"
enabled=1
gpgcheck=0

[6local_osd-rpms]
name=osd-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhceph-2-osd-rpms"
enabled=1
gpgcheck=0

[7local_rhceph-2-tools-rpms]
name=rhceph-2-tools-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhceph-2-tools-rpms"
enabled=1
gpgcheck=0

[8local_agent-rpms]
name=agent-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhscon-2-agent-rpms"
enabled=1
gpgcheck=0

[9local_installer-rpms]
name=installer-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhscon-2-installer-rpms"
enabled=1
gpgcheck=0

[10local_rhscon-2-main-rpms]
```

```
name=rhscn-2-main-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhscn-2-main-rpms"
enabled=1
gpgcheck=0
```

## 2. 案例 2：配置 DNS 服务器：

### • 问题

本案例要求掌握 DNS 服务器的配置：

- 允许 DNS 服务器为所有的客户端提供服务
- 解析域名 openstack.tedu.cn
- 解析域名 nova.tedu.cn

### • 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一：三台虚拟机配置静态 ip

备注：三台主机同样操作，改一下 ip 就可以了

```
[root@localhost ~]# echo openstack.tedu.cn > /etc/hostname
[root@localhost ~]# hostname openstack.tedu.cn //另外两台主机改名为 nova.tedu.cn 和
dnsntp
[root@openstack ~]# vim /etc/sysconfig/network-scripts/ifcfg-eth0
# Generated by dracut initrd
DEVICE="eth0"
ONBOOT="yes"
IPV6INIT="no"
IPV4_FAILURE_FATAL="no"
NM_CONTROLLED="no"
TYPE="Ethernet"
BOOTPROTO="static"
IPADDR="192.168.1.1"
PREFIX=24
GATEWAY=192.168.1.254
[root@docker1 ~]# systemctl restart network
[root@zrj ~]# scp /etc/yum.repos.d/local.repo \
192.168.1.3:/etc/yum.repos.d/ //拷贝给 dnsntp 这台主机
```

#### 步骤二：配置 DNS

```
[root@dnsntp ~]# yum -y install bind bind-chroot
[root@dnsntp ~]# vim /etc/named.conf
options {
    listen-on port 53 { 192.168.1.3; }; //修改 ip
    allow-query { any; }; //允许所有
    recursion yes;
    forwarders { 172.40.1.10; }; //转发 dns,真机的服务器地址

    dnssec-enable no;
```

```
dnssec-validation no;
};
[root@dnsntp ~]# systemctl restart named
```

### 步骤三：域名解析

```
[root@openstack ~]# vim /etc/hosts    //在 openstack.tedu.cn 和 nova.tedu.cn 主机上
面操作
192.168.1.1 openstack.tedu.cn
192.168.1.2 nova.tedu.cn
```

测试能否 ping 通，如图-1 所示：

```
[root@openstack ~]# ping nova.tedu.cn
PING nova.tedu.cn (192.168.1.2) 56(84) bytes of data.
64 bytes from nova.tedu.cn (192.168.1.2): icmp_seq=1 ttl=255 time=0.464 ms
64 bytes from nova.tedu.cn (192.168.1.2): icmp_seq=2 ttl=255 time=0.281 ms
^C
--- nova.tedu.cn ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.281/0.372/0.464/0.093 ms
[root@openstack ~]#

[root@nova network-scripts]# ping openstack.tedu.cn
PING openstack.tedu.cn (192.168.1.1) 56(84) bytes of data.
64 bytes from openstack.tedu.cn (192.168.1.1): icmp_seq=1 ttl=255 time=0.470 ms
64 bytes from openstack.tedu.cn (192.168.1.1): icmp_seq=2 ttl=255 time=0.261 ms
^C
--- openstack.tedu.cn ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.261/0.365/0.470/0.106 ms
[root@nova network-scripts]#
```

图-1

## 3. 案例 3：配置 NTP 服务器

### • 问题

本案例要求配置 NTP 时间同步服务器：

- 将 NTP 服务与 DNS 服务部署在同一台主机上
- 确认 NTP 服务器的时区是东八区
- 确认 NTP 服务器的时间准确
- 计划安装 openstack 的服务器与 NTP 服务器进行时间校正

### • 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一：配置 NTP 时间同步

```
[root@dnsntp ~]# yum -y install chrony
[root@dnsntp ~]# vim /etc/chrony.conf
```

```
server ntp1.aliyun.com iburst
bindaddress 0.0.0.0
allow 0/0 //允许所有人使用我的时间服务器
cmdallow 127.0.0.1 //控制指令
[root@dnsntp ~]# systemctl restart chronyd
[root@dnsntp ~]# netstat -antup | grep chronyd
udp          0            0  0.0.0.0:123          0.0.0.0:*
23036/chronyd
udp          0            0  127.0.0.1:323       0.0.0.0:*
23036/chronyd
[root@dnsntp ~]# chronyc sources -v //出现*号代表 NTP 时间可用
^* 120.25.115.20      2  6  17  62  -753us[-7003us] +/- 24ms
```

## 4. 案例 4：环境准备

### • 问题

本案例要求准备基础环境，为安装 openstack 做准备：

- 准备 openstack 的基础环境

### • 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一：准备基础环境

##### 1) 配置 yum 源

```
[root@openstack ~]# vim /etc/yum.repos.d/local.repo
[local_repo]
name=CentOS-$releasever - Base
baseurl="ftp://192.168.1.254/system"
enabled=1
gpgcheck=1

[local_extras]
name=extras
baseurl="ftp://192.168.1.254/extras"
enabled=1
gpgcheck=0

[1local_devtools-rpms]
name=devtools-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-openstack-10-devtools-rpms"
enabled=1
gpgcheck=0

[2local_optools-rpms]
name=optools-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-openstack-10-optools-rpms"
enabled=1
gpgcheck=0

[3local_rpms]
name=rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-openstack-10-rpms"
enabled=1
gpgcheck=0
```

```
[4local_tools-rpms]
name=tools-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-openstack-10-tools-rpms"
enabled=1
gpgcheck=0

[5local_mon-rpms]
name=mon-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhceph-2-mon-rpms"
enabled=1
gpgcheck=0

[6local_osd-rpms]
name=osd-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhceph-2-osd-rpms"
enabled=1
gpgcheck=0

[7local_rhceph-2-tools-rpms]
name=rhceph-2-tools-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhceph-2-tools-rpms"
enabled=1
gpgcheck=0

[8local_agent-rpms]
name=agent-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhscon-2-agent-rpms"
enabled=1
gpgcheck=0

[9local_installer-rpms]
name=installer-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhscon-2-installer-rpms"
enabled=1
gpgcheck=0

[10local_rhscon-2-main-rpms]
name=rhscon-2-main-rpms
baseurl="ftp://192.168.1.254/HEL70SP/rhel-7-server-rhscon-2-main-rpms"
enabled=1
gpgcheck=0
[root@openstack ~]# scp /etc/yum.repos.d/local.repo \
    192.168.1.2:/etc/yum.repos.d/ //拷贝给 nova.tedu.cn 这台主机，之前 dnsttp 主机已
    经拷贝过 yum 源，这里不再重复拷贝
```

## 步骤二：配置 ip

备注：配置 eth0 为公共网络，网络地址 192.168.1.0/24

配置 eth1 为隧道接口，网络地址 192.168.2.0/24

1) 给 openstack.tedu.cn 主机添加 eth1 网卡

```
[root@zrj networks]# virsh -c qemu:///system attach-interface openstack bridge
private2 --model virtio
Interface attached successfully //添加成功
[root@openstack ~]# cd /etc/sysconfig/network-scripts
[root@openstack network-scripts]# cp ifcfg-eth0 ifcfg-eth1
[root@openstack network-scripts]# vim ifcfg-eth1
# Generated by dracut initrd
DEVICE="eth1"
ONBOOT="yes"
```

```
IPV6INIT="no"
IPV4_FAILURE_FATAL="no"
NM_CONTROLLED="no"
TYPE="Ethernet"
BOOTPROTO="static"
IPADDR="192.168.2.1"
PREFIX=24
GATEWAY=192.168.1.254
[root@openstack network-scripts]# systemctl restart network
```

## 2) 给 nova.tedu.cn 主机添加 eth1 网卡

```
[root@zrj networks]# virsh -c qemu:///system attach-interface nova bridge private2
--model virtio
Interface attached successfully //添加成功
[root@nova ~]# cd /etc/sysconfig/network-scripts
[root@nova network-scripts]# cp ifcfg-eth0 ifcfg-eth1
[root@nova network-scripts]# vim ifcfg-eth1
# Generated by dracut initrd
DEVICE="eth1"
ONBOOT="yes"
IPV6INIT="no"
IPV4_FAILURE_FATAL="no"
NM_CONTROLLED="no"
TYPE="Ethernet"
BOOTPROTO="static"
IPADDR="192.168.2.2"
PREFIX=24
GATEWAY=192.168.1.254
[root@openstack network-scripts]# systemctl restart network
```

## 3) 配置卷组 (openstack 主机上面操作)

```
[root@zrj images]# qemu-img create -f qcow2 disk.img 50G
Formatting 'disk.img', fmt=qcow2 size=53687091200 encryption=off cluster_size=65536
lazy_refcounts=off
[root@zrj networks]# virsh -c qemu:///system attach-disk openstack \
/var/lib/libvirt/images/disk.img vdb --subdriver qcow2 --sourcetype file
Disk attached successfully //添加成功
[root@openstack ~]# yum install lvm2
[root@openstack ~]# pvcreate /dev/vdb
[root@openstack ~]# vgcreate cinder-volumes /dev/vdb
```

## 5) 安装 openstack 的依赖包 (openstack.tedu.cn 和 nova.tedu.cn 主机上面操作)

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

## 5. 案例 5 : 部署 Openstack :

### • 问题

本案例要求通过 packstack 完成以下配置 :

- 通过 packstack 部署 Openstack



- 根据相关日志文件进行排错

## • 步骤

实现此案例需要按照如下步骤进行。

### 步骤一：安装 packstack

```
[root@openstack ~]# yum install -y openstack-packstack
[root@openstack ~]# packstack --gen-answer-file answer.ini //answer.ini 与
answer.txt 是一样的，只是用 vim 打开 answer.ini 文件有颜色
Packstack changed given value to required value /root/.ssh/id_rsa.pub
[root@openstack ~]# vim answer.ini
 11 CONFIG_DEFAULT_PASSWORD=redhat //密码
 42 CONFIG_SWIFT_INSTALL=n
 75 CONFIG_NTP_SERVERS=192.168.1.3 //时间服务器的地址
554 CONFIG_CINDER_VOLUMES_CREATE=n //创建卷,已经手动创建过了
840 CONFIG_NEUTRON_ML2_TYPE_DRIVERS=flat,vxlan //驱动类型
876 CONFIG_NEUTRON_ML2_VXLAN_GROUP=239.1.1.5 //设置组播地址,最后一个随意不能为 0 和
255,其他固定
910 CONFIG_NEUTRON_OVS_BRIDGE_MAPPINGS=physnet1:br-ex //物理网桥的名称
921 CONFIG_NEUTRON_OVS_BRIDGE_IFACES=br-ex:eth0 //br-ex 桥的名称与 eth0 连接,管理
eth0,网桥与哪个物理网卡连接
936 CONFIG_NEUTRON_OVS_TUNNEL_IF=eth1
1179 CONFIG_PROVISION_DEMO=n //DEMO 是否测试
[root@openstack ~]# packstack --answer-file=answer.ini
**** Installation completed successfully **** //出现这个为成功
```

## 6. 案例 6：网络管理

### • 问题

本案例要求运用 OVS 完成以下配置：

- 配置外部 OVS 网桥及其端口
- 验证 OVS 配置

### • 步骤

实现此案例需要按照如下步骤进行。

### 步骤一：配置外部 ovs 网桥

#### 1) 配置 br-ex 为外部 OVS 网桥

```
[root@openstack ~]# cat /etc/sysconfig/network-scripts/ifcfg-br-ex
ONBOOT="yes"
NM_CONTROLLED="no"
IPADDR="192.168.1.1"
PREFIX=24
GATEWAY=192.168.1.254
```

```
DEVICE=br-ex
NAME=br-ex
DEVICETYPE=ovs
OVSBOOTPROTO="static"
TYPE=OVSBridge
```

## 2) 配置 eth0 为外部 OVS 网桥的端口

```
[root@nova ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
NAME=eth0
DEVICETYPE=ovs
TYPE=OVSPort
OVS_BRIDGE=br-ex
ONBOOT=yes
BOOTPROTO=none
```

## 3) 验证 OVS 配置

```
[root@nova ~]# ovs-vsctl show
Bridge br-ex
  Controller "tcp:127.0.0.1:6633"
    is_connected: true
  fail_mode: secure
  Port br-ex
    Interface br-ex
      type: internal
  Port phy-br-ex
    Interface phy-br-ex
      type: patch
      options: {peer=int-br-ex}
  Port "eth0"
    Interface "eth0"
  ovs_version: "2.5.0"
```

# 7. 案例 7：管理项目

## • 问题

本案例要求通过 Horizon 完成以下操作：

- 创建名为 myproject 的项目
- 查看项目信息
- 更新 vcpu 配额为 30
- 删除 myproject
- 

## • 步骤

实现此案例需要按照如下步骤进行。

### 步骤一：浏览器访问 openstack

#### 1) 浏览器访问，不出现页面，改配置文件

```
[root@openstack conf.d]# firefox 192.168.1.1
[root@openstack ~]# cd /etc/httpd/conf.d/
```

```
[root@openstack conf.d]# vi 15-horizon_vhost.conf
35  WSGIProcessGroup apache
36  WSGIApplicationGroup %{GLOBAL}    //添加这一行
[root@openstack conf.d]# apachectl graceful //重新载入配置文件
```

2) 浏览器访问，出现页面，如图-2 所示：

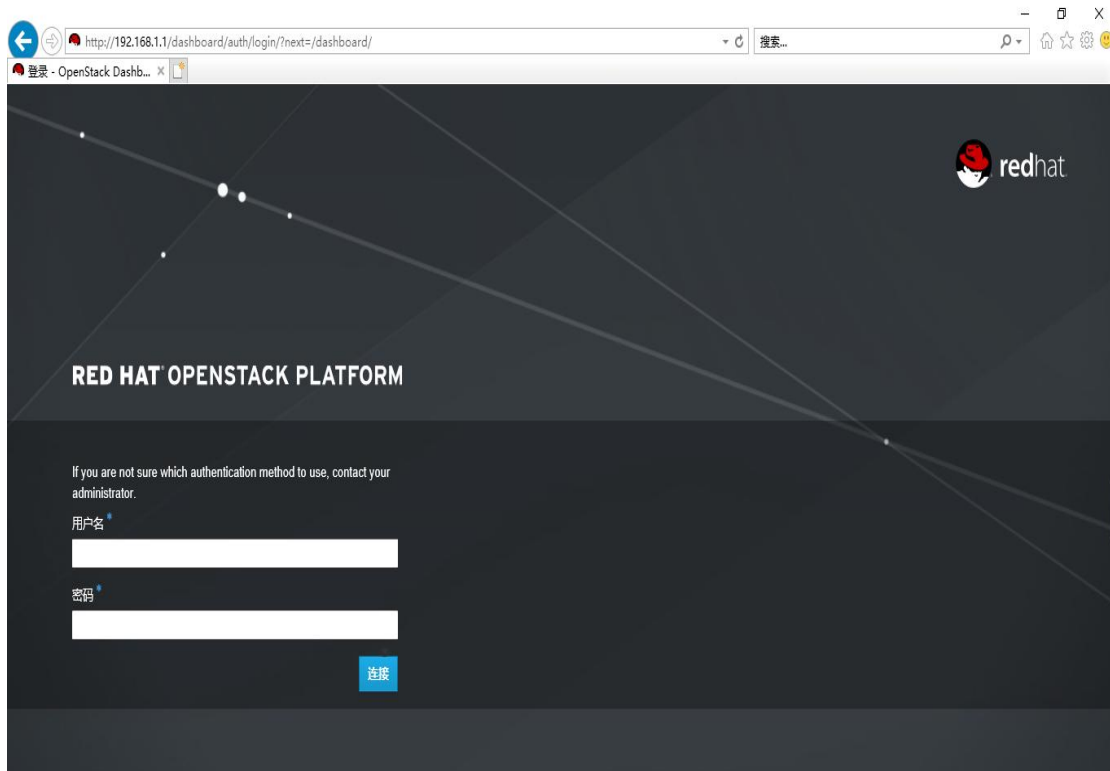


图-2

3) 输入用户名和密码，如图-2 所示：

```
[root@openstack conf.d]# firefox 192.168.1.1
[root@openstack conf.d]# cd
[root@openstack ~]# ls
answer.ini  keystone_admin //keystone_admin 生成的文件，里面有用户名和密码
[root@openstack ~]# cat keystone_admin
unset OS_SERVICE_TOKEN
export OS_USERNAME=admin //用户名
export OS_PASSWORD=1bb4c987345c45ba //密码
export OS_AUTH_URL=http://192.168.1.1:5000/v2.0
export PS1='[\u@\h \W(keystone_admin)]\$ '

export OS_TENANT_NAME=admin
export OS_REGION_NAME=RegionOne
```

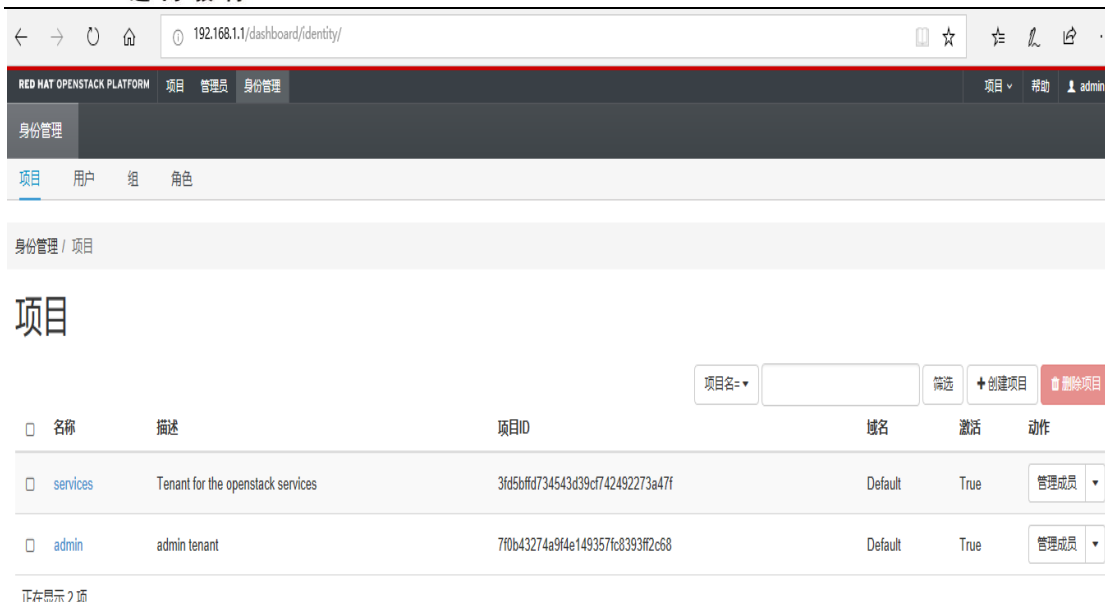


图-2

#### 4) 创建名为 myproject 的项目

```
[root@openstack ~(keystone_admin)]# openstack project create myproject
+-----+
| Field | Value |
+-----+
| description | None |
| enabled | True |
| id | f73949e2701346328faf5c1272414cd0 |
| name | myproject |
+-----+
```

#### 5) 查看项目信息

```
[root@openstack ~(keystone_admin)]# openstack project list
+-----+
| ID | Name |
+-----+
| 3fd5bffd734543d39cf742492273a47f | services |
| 7f0b43274a9f4e149357fc8393ff2c68 | admin |
| f73949e2701346328faf5c1272414cd0 | myproject |
+-----+
```

#### 6) 更新 vcpu 配额为 30

```
[root@openstack ~(keystone_admin)]# nova quota-update --cores 30 myproject
```

#### 7) 删除 myproject

```
[root@openstack ~(keystone_admin)]# openstack project delete myproject
```