**openstack概述**

云计算简介

什么是云计算

基于互联网的相关服务的增加、使用和交付模式

这种模式提供可用的、便捷的、按需的网络访问，进入可配置的计算资源共享池（资源包括网络，服务器，存储，应用软件，服务）

这些资源能够被快速提供，只需投入很少的管理工作，或与服务供应商进行很少的交互

通常涉及通过互联网来提供动态易扩展且经常是虚拟化的资源

IaaS

IaaS(infrastruvture as a service)，即基础设施即服务

提供给消费者的服务是对所有计算基础设施的利用，包括处理CPU、内存、网络和其他基本的计算资源，用户能够部署和运行任意软件，包括操作系统和应用程序

消费者不管理或控制任何云计算基础设施，但能控制操作系统的选择、存储的空间、部署的应用

IaaS通常分为三种用法：公有云共的和、私有云和混合云

PaaS

PaaS是Platfrom-as-a-Service的缩写，意思是平台即服务

云计算时代相应的服务器平台或者开发环境作为服务进行提供就成为了PaaS

PaaS运营商所需提供的服务，不仅仅是单纯的基础平台，而且包括针对该平台的技术支持服务，甚至针对该平台而进行的应用系统开发、优化等服务

简单的说，PaaS平台就是指云环境中的应用基础设施服务，也可以说是中间件即服务

SaaS

SaaS是software-as-a-service（软件即服务）的简称

它是一种通过internet提供软件的模式，厂商将应用软件统一部署到自己的服务器上，客户可以根据自己实际需求，通过互联网向厂商订购所需的应用软件服务

用户不用再购买软件，而改用向提供商租用基于web的软件，来管理企业经营活动，且无需对软件进行维护，服务提供商会全权管理和维护软件，软件厂商在向客户提供互联网应用的同时，也提供软件的离线操作和本地数据存储，让用户随时随地的可以使用其定购的软件和服务

OpenStack简介

什么是OpenStack

OpenStack是一个由NASA（美国国家航空航天局）和rackspace合作研发并发起的项目

OpenStack是一套IaaS解决方案

OpenStack是一个开源的云计算管理平台

以apache许可证为授权

OpenStack主要组件

Horizon

用于管理openstack各种服务的、基于web的管理接口

通过图形用户界面实现创建用户、管理服务、启动实例等操作

keystone

为其他服务提供认证和集中身份管理服务

也提供了集中的目录服务

支持多种身份认证模式，如密码认证、令牌认证、以及AWS（亚马逊WEB服务）登录

为用户和其他服务提供了SSO认证服务

Neutron

一种软件定义网络服务

用于创建网络、子网、路由器、管理浮动IP地址

可以实现虚拟交换机、虚拟路由器

可以在项目中创建VPN

Cinder

为虚拟机管理存储卷的服务

为运行在Nova（托管节点）中的实例提供永久的块存储

可以通过快照进行数据备份

经常应用在实例存储环境中，如数据库文件

nova

在节点上用于管理虚拟机的服务

nova是一个分布式的服务，能够与keystone交互实现认证，与glance交互实现镜像管理

nova被设计成在标准硬件上能够水平扩展

启动实例时，如果有需要则下载镜像

Glance

扮演虚拟机镜像注册角色

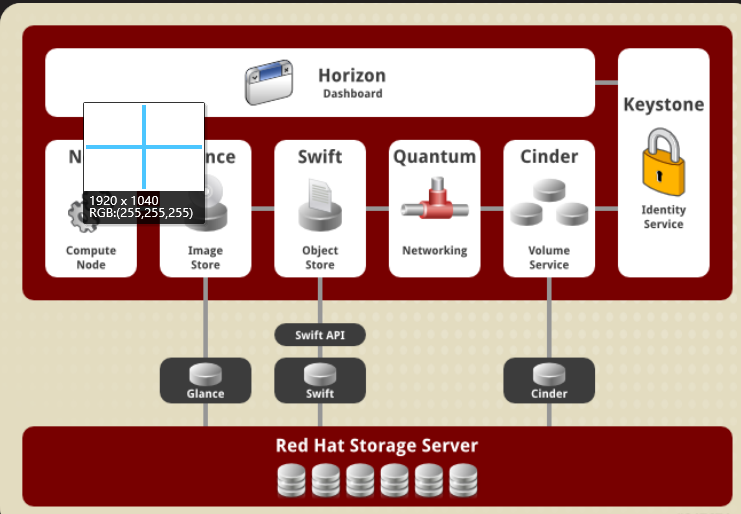
允许用户为直接存储拷贝服务器镜像

这些镜像可以用于新建虚拟机的模版

OpenStack版本



OpenStack结构



**部署安装环境**

yum仓库及名称解析

配置yum仓库（12个yum源）

RHEL7光盘信息必须提供（1个）

RHEL7OSP光盘拥有众多目录，每个目录都是一个仓库，因此，需要把这些目录都配置为yum源（10个）

RHEL7-extras扩展包光盘在安装的时候也是需要的（1个）

安装完成后可以看到12个源和10731个包

设置DNS服务器

系统环境准备

openstack安装时候需要使用外部dns来解析域名，并且还需要外部事件服务器来保证所有节点的时间保持一致

我们需要创建一个dns服务器，并且为我们的主机提供域名解析

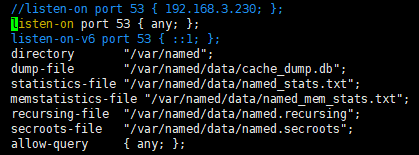
将openstack.tedu.cn域名对应的IP解析到对应的openstac的服务器上

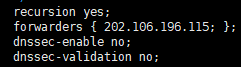
注：DNS服务器不能与OpenStack安装在同一台主机上

真机：

yum -y install bind-chroot

vim /etc/named.conf

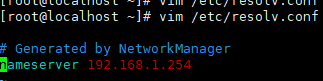




systemctl restart named

systemctl enable named

虚拟机：

改成物理机IP

NTP服务

时区

时区：由于同一时间点上面，整个地球的时间应该都不一样，为了解决这个问题，地球被分成了24个时区

地球被人类以经纬度坐标来进行定位，而精度为零的地点在英国（格林威治）这个城市所在纵剖面上

地球一圈是360度角，这360度角共分为24个时区，一个时区就是15度角

UTC时间

计算时间的时候，最准确的计算应该是使用（原子震荡周期）所计算的物理时钟了（atomic clock，也被称为原子种），这也被定义为标准时间

我们常常看见的UTC（协和标准时间）就是利用这种物理时钟为基准所定义出来的正确时间

UTC标准时间是以GMT这个时区为主，本地时间与UTC时间的时差就是本地时间与GMT时间的时差

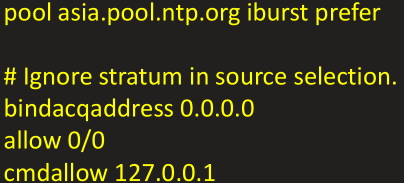
配置NTP服务

安装

yum -y install chrony

修改配置文件

vim /etc/chrony.conf



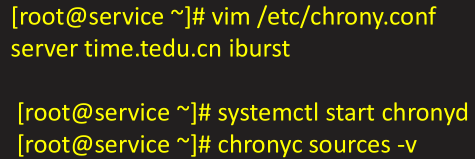
启动并测试

启动服务

systemctl start chronyd

systemctl enable chronyd

在另一台服务器上测试校准时间



真机安装服务

虚拟机里server那项配置为真机IP

**部署OpenStack**

基础环境准备

配置IP地址

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192.168.4.0/24

关闭networkmanager服务

禁用selinux

卸载firewalld

配置yum客户端

将RHEL7光盘作为基础的yum源

将RHEL7-extars光盘作为扩展的yum源

将RHEL7OSP光盘中所有目录作为yum源

配置卷组（管理节点安装）

OpenStack为虚拟机提供的云硬盘，本质上是本地的逻辑卷

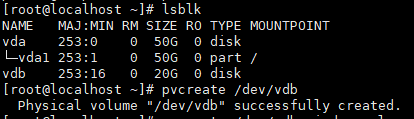
逻辑卷创建于名为cinder-volumes的卷组

没有物理卷可以使用loop文件代替

losetup loop0 /var/disk.img

pvcreate /dev/loop0

vgcreate cinder-volumes /dev/loop0







导入公钥（所有节点）

安装OpenStack期间，系统会要求密钥验证

手动导入系统密钥

密钥文件在光盘中提供

rpm --import PRM-GPG-KEY-CentOS-7

安装额外软件包（所有节点）

安装OpenStack期间，有些软件所依赖的软件包，并没有在安装过程中安装

这些软件包需提前安装

本地RPM包也可以通过yum进行安装

yum install -y qemu-kvm libvirt-client libvirt-daemon libvirt-daemon-driver-qemu python-setuptools

安装OpenStack

配置packstack

安装packstack（管理节点）

yum -y install openstack-packstack

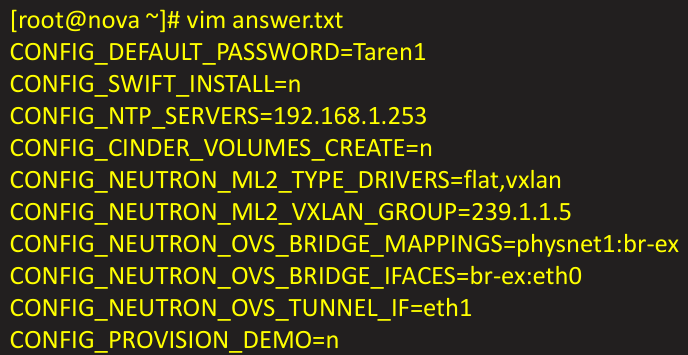


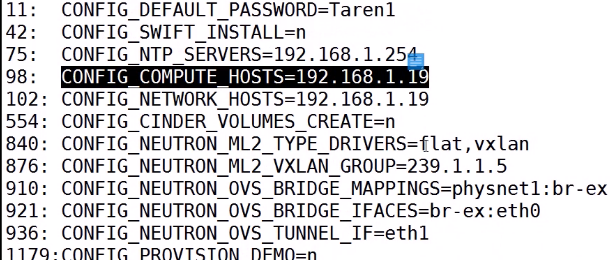
packstack --gen-answer-file answer.txt





修改应答文件





98行是计算节点（nova）的IP地址

102行是配置主机的网络，哪些主机要变成这个网络

设定密码

禁用swift存储

配置NTP服务器

禁止自动创建卷，默认会从第一个硬盘的第一个设备创建

扁平网络（外网，flat），出口网卡，内网vxlan

主播地址，vxlan设备互联互通

配置虚拟机（网桥），br-ex是ifconfig可以看见的设备，用于和外网进行通信

虚拟交换机跟eth0连接，出外网用

承载多个vlan

测试用的demo，不用测试，直接使用

一键部署OpenStack

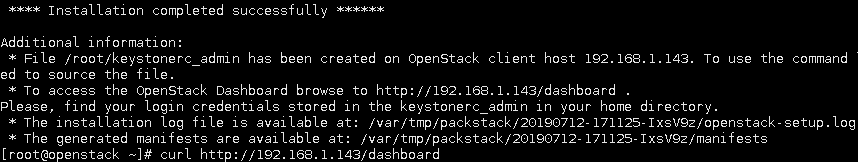
如果前期环境准备无误，只需耐心等待安装结束即可

根据主机配置不同，安装过程需要20分钟或更久

如果出现错误，根据屏幕上给出的日志文件进行排错

packstack --answer-file answer.txt

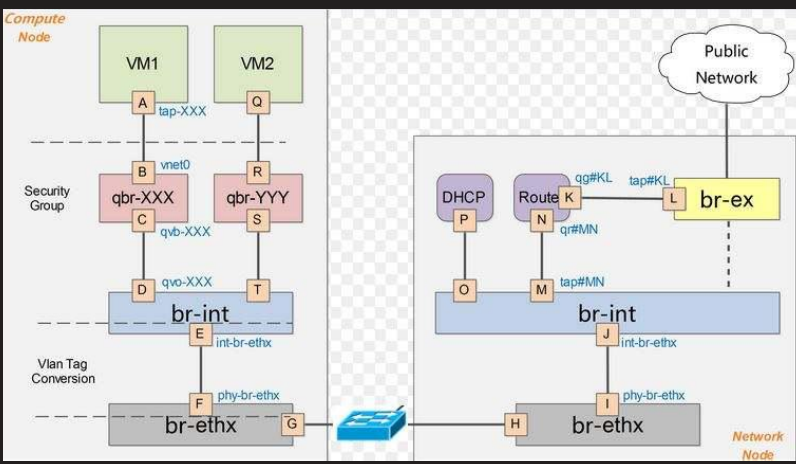




网络配置

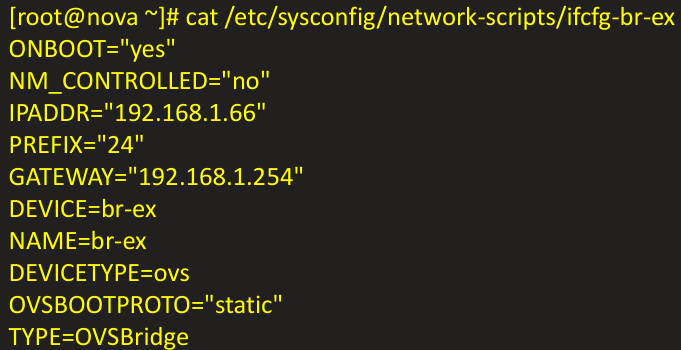
网络拓扑

多计算节点的拓扑如下所示



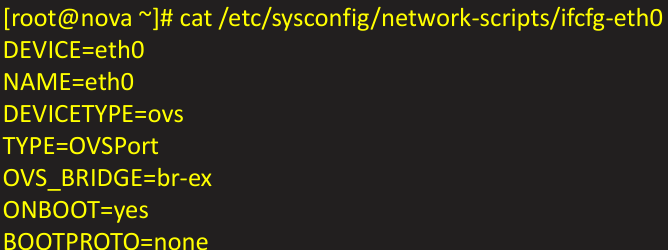
配置外部OVS网桥

配置br-ex为外部OVS网桥



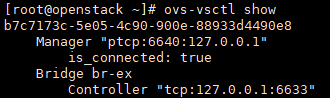
配置外部OVS网桥接口

配置eth0为外部OVS网桥接口

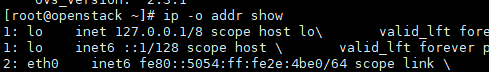


验证OVS配置

ovs-vsctl show



ip -o addr show



**OpenStack操作基础**

horizon配置

概述

horizon是一个用以管理、控制OpenStack服务的web控制面板，也称之为dashboard仪表盘

可以管理实例、镜像、创建密钥对，对实例添加卷、操作swift容器等。除此之外，用户还可以在控制面板中使用终端或VNC直接访问实例

基于python的django web框架进行开发

功能与特点

实例管理：创建、终止实例，查看终端日志，VNC连接，添加终端等

访问与安全管理：创建安全群组，管理密钥对。设置浮动IP等

偏好设置：对虚拟硬件模版可以进行不同偏好设定

镜像管理：编辑或删除镜像

用户管理：创建用户等

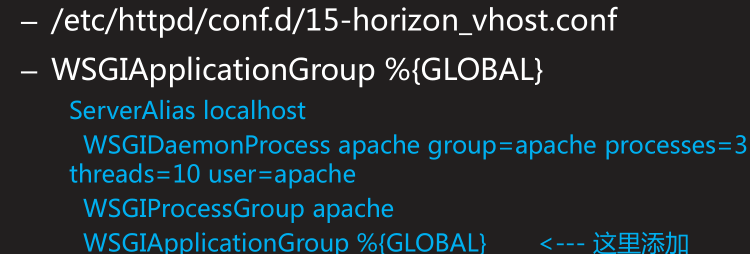
卷管理：创建卷和快照

对象存储管理：创建、删除容器和对象

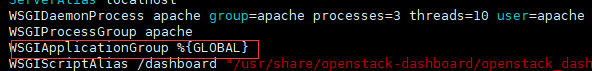
horizon

horizon bug处理

安装虽然没有报错，但默认无法打开horizon，这是一个软件的配置bug



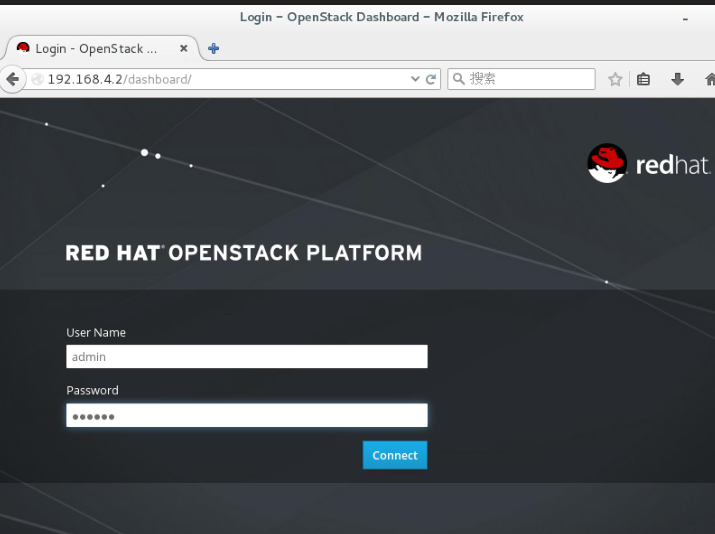


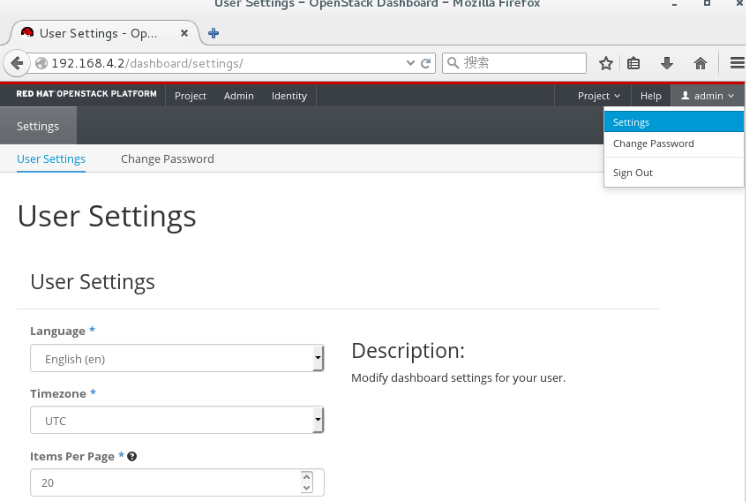


重新载入配置文件apache graceful



基本配置





项目管理

基本概念

项目：一组隔离的资源和对象。由一组关联的用户进行管理

在老的版本里，也是租户来表示

根据配置的需求，项目对应一个组织、一个公司或是一个客户等

项目中可以有多个用户，项目中的用户可以在该项目创建、管理虚拟资源

具有admin角色的用户可以创建项目

项目相关信息保存到mariadb中

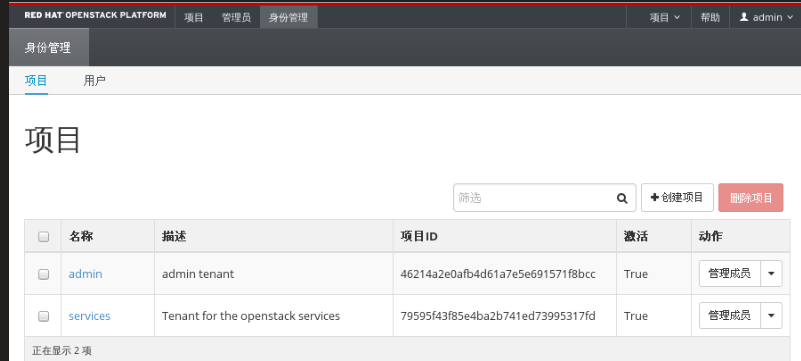
默认情况下，packstack安装的OpenStack中有两个独立的项目

admin：为admin账户创建的项目

services：与安装的各个服务相关联

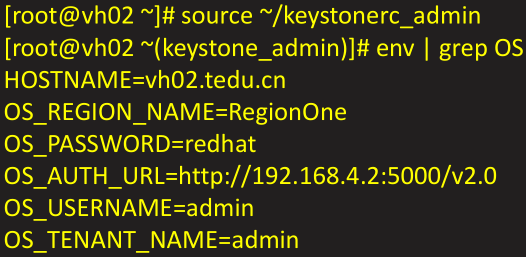
在horizon中管理项目

通过horizon可以进行项目的创建和删除



命令接口基础

初始化环境变量



使用帮助



通过命令行管理项目

创建名为myproject项目

openstack project create myproject

列出所有项目

openstack project list

查看myproject详细信息

openstack project show myproject

禁用与激活项目

openstack project set --disable myproject

openstack project set --enable myproject

查看项目配额

nova quto-show --tenant myproject

更新可用vcpu数目为30

nova quota-update --cores 30

删除myprojects

openstack project delete myproject