

基础

一 OpenStack 概述-云计算简介

1.1 什么是云计算

- a 基于互联网的相关服务的增加\使用和交互模式
- b 这种模式是提供可用的\便捷的\按需的网络访问,进入可配置的计算资源共享池
- c 这些资源能够被快速提供,只需投入很少的管理工作,与服务供应商进行很少的交互
- d 通常设计通过互联网来提供动态易扩展且经常是虚拟化的资源

1.2 IaaS 虚拟机

Infrastructure as a Service,基础设置即服务

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\存储\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

IaaS 通常分为 3 种用法:共有云,私有云,混合云

1.3 PaaS 代表:京东平台,淘宝平台

Platform as a Service,平台即服务

以服务器平台或者开发环境作为服务进行提供,就称为了 PaaS

PaaS 运营商所需提供的服务,不仅仅是单纯的基础平台,还提供针对该平台的技术支持服务,甚至针对该平台而进行应用系统开发\优化等服务

简单地说,PaaS 平台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

1.4 SaaS APPSTORE

Software as a Service,软件即服务

是一种通过 Internet 提供软件的模式,厂商将应用软件统一部署在自己的服务器上,

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,而是向提供商租用基于 web 的软件,来管理企业经营活动,不用

对软件进行维护,提供商会全权管理和维护软件,同时也提供软件的离线操作和本地数

据存储

二 OpenStack 概述-OpenStack 简介

2.1 什么是 OpenStack

是一个由 NASA 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

2.2 OpenStack 主要组件

Horizon: 用于管理 OpenStack 各种服务的\基于 web 的管理接口

[hə'raɪzn] 通过图形界面实现创建用户\管理网络\启动实例等操作

Keystone: 为其他服务提供认证和授权的集中身份管理服务

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 登录

为用户和其他服务提供了 SSO 认证服务

Neutron: quantum['kwɒntəm] 一种软件定义网络服务

['nju:trɒn]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

可用于在项目中创建 VPN

Cinder: 为虚拟机管理存储卷的服务

['sɪndə(r)]为运行在 Nova 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

Glance: 扮演虚拟机镜像注册的角色

[glɑ:ns] 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板

Nova: 在节点上用于管理虚拟机的服务

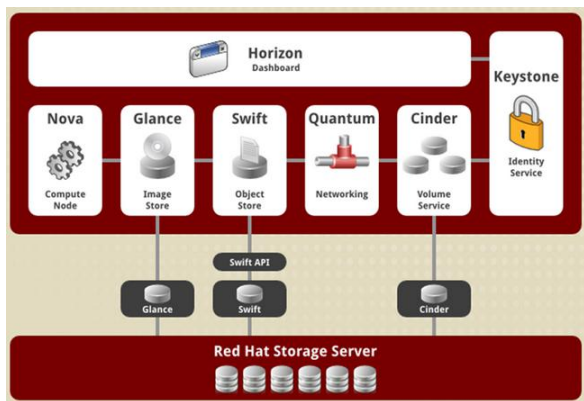
['nəʊvə] Nova 是一个分布式的服务,能够与 Keystone 交互实现认证,与 Glance 交互实现镜像管理

Nova 被设计成在标准硬件上能够进行水平扩展

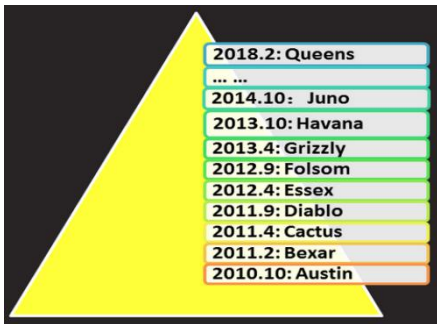
启动实例时,如果有则需要下载镜像

swift 提供对象存储,在不同的虚拟机间共享数据

2.3 OpenStack 结构图



2.4 OpenStack 版本



三 部署安装环境-openstack 环境准备

3.1 准备虚拟机



3.1.1 创建虚拟机

Openstack 管理主机：

2cpu, 6G 内存, 50G 硬盘, 静态 IP: 192.168.1.10

nova1 nova2 计算节点:

2cpu,4.5G 内存,100G 硬盘,静态 IP:192.168.1.11\12

room9pc01 ~]\$ base-vm openstack nova1 nova2

3.1.2 修改虚拟机内存\硬盘大小

room9pc01 ~]\$ virsh edit openstack\nova1\nova2 #修改内存

room9pc01 ~]\$ virsh blockresize \

--path /var/lib/libvirt/images/openstack\nova1\nova2.img \

--size 50G\100G\100G openstack\nova1\nova2 #扩展磁盘

localhost ~]\$ /usr/bin/growpart /dev/vda 1 #扩展分区

localhost ~]\$ /usr/sbin/xfs_growfs / #扩展根分区

3.1.3 DNS 域名解析设置(3 台相同设置)

openstack ~]# ping www.baidu.com

openstack ~]# vim /etc/resolv.conf 删除 search localhost 行

openstack ~]# vim /etc/hosts 添加以下三行

192.168.1.10 openstack

192.168.1.11 nova1

192.168.1.12 nova2

3.1.4 NTP 时间服务器设置(3 台相同设置)

penstack ~]# vim /etc/chrony.conf

7 server 192.168.1.254 iburst

```
openstack ~]# systemctl restart chronyd
```

```
openstack ~]# chronyc sources -v
```

```
^* gateway 3 6 17 16 -9212ns[ -25us] +/- 17ms #^*表示设置成功
```

3.2 配置 yum 仓库(3 台相同设置)

yum makecache 更改 yum 源后创建新的缓存

```
room9pc01 ~]$ cd /var/ftp
```

```
room9pc01 ftp]$ mkdir extras RHEL70PS
```

```
room9pc01 ftp]$ mount /linux-soft/04/openstack/RHEL7-extras.iso
```

```
/var/ftp/extras/
```

```
room9pc01 ftp]$ mount /linux-soft/04/openstack/RHEL70SP-10.iso
```

```
/var/ftp/RHEL70PS/
```

```
openstack ~]# vim /etc/yum.repos.d/local.repo
```

```
[local_repo]
```

```
name=CentOS-$releasever - Base
```

```
baseurl="ftp://192.168.1.254/centos-1804"
```

```
enabled=1
```

```
gpgcheck=0
```

[extras]

name=extras

baseurl="ftp://192.168.1.254/extras"

enabled=1

gpgcheck=0

[openstack-10-rpms]

name=penstack-10-rpms

baseurl="ftp://192.168.1.254/RHEL70PS/rhel-7-server-openstack-
10-rpms"

enabled=1

gpgcheck=0

[openstack-10-devtools-rpms]

name=penstack-10-devtools-rpms

baseurl="ftp://192.168.1.254/RHEL70PS/rhel-7-server-openstack-
10-devtools-rpms"

enabled=1

gpgcheck=0

四 部署 Openstack-基础环境准备

4.1 安装额外软件包

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

4.2 检查 openstack 环境部署(3 机相同)

检查 firewall 和 networkmanager 是否卸载

```
openstack ~]# rpm -qa | grep netmanager*  
openstack ~]# rpm -qa | grep firewall*
```

检查主机网络参数 BOOTPROTO="static", 静态 IP

```
openstack ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
```

检查主机互 ping 主机名

```
openstack ~]# ping -c 2 openstack\nova1\nova2  
nova1 ~]# ping -c 2 openstack\nova1\nova2  
nova2 ~]# ping -c 2 openstack\nova1\nova2
```

检查 yum 源

```
openstack ~]# yum clean all  
openstack ~]# yum repolist      #repolist: 10,670
```

检查时间同步是否可用

```
openstack ~]# chronyc sources -v      # ^* gateway
```

检查/etc/resolv.conf 不能有 serach 开头的行

```
openstack ~]# cat /etc/resolv.conf
```

4.3 nova1 nova2 上启动 libvirtd,并设置开机启动

```
nova1 ~]# systemctl start\status\enable libvirtd
```

五 部署 Openstack-安装 Openstack

5.1 配置 packstack

5.1.1 安装 openstack-packstack

```
openstack ~]# yum -y install openstack-packstack
```

5.1.2 使用 packstack 创建通用应答文件

```
openstack ~]# packstack --gen-answer-file=answer.ini
```

Packstack changed given value to required value /root/.ssh/id_rsa.pub

5.1.3 修改应答文件

```
openstack ~]# vim answer.ini
```

```
42 CONFIG_SWIFT_INSTALL=n          #是否使用对象存储 SWIFT
```

```
45 CONFIG_CEILOMETER_INSTALL=n     #计费相关模块,关闭或不安装
```

```
49 CONFIG_AODH_INSTALL=n           #计费相关模块,关闭或不安装
```

```
53 CONFIG_GNOCCHI_INSTALL=n        #计费相关模块,关闭或不安装
```

```
75 CONFIG_NTP_SERVERS=192.168.1.254 #时间服务器地址
```

```
98 CONFIG_COMPUTE_HOSTS=192.168.1.11 #哪台机器安装 nova 组件
102 CONFIG_NETWORK_HOSTS=192.168.1.10,192.168.1.11
    #哪台机器安装 neutron
333 CONFIG_KEYSTONE_ADMIN_PW=a          #管理员密码
840 CONFIG_NEUTRON_ML2_TYPE_DRIVERS=flat,vxlan #驱动类型
910 CONFIG_NEUTRON_OVS_BRIDGE_MAPPINGS=physnet1:br-ex
    #物理网桥名称,将 eth0 变成此网桥的一个端口
921 CONFIG_NEUTRON_OVS_BRIDGE_IFACES=br-ex:eth0
    #br-ex 桥的名称与 eth0 连接,管理 eth0,网桥与那个物理网卡连接
1179 CONFIG_PROVISION_DEMO=n    #样例 demo 是否测试
```

5.2 一建部署 OpenStack

如果前期环境准备无误,只要耐心等待安装接收即可

根据主机配置不同,安装过程需要 20 分钟左右或更久

如果出现错误,根据屏幕上给出的日志文件进行排错

```
openstack ~]# packstack --answer-file=answer.ini
```

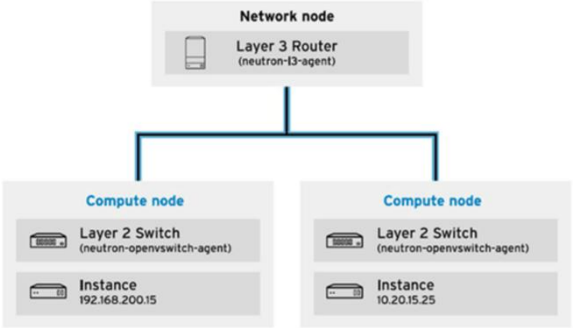
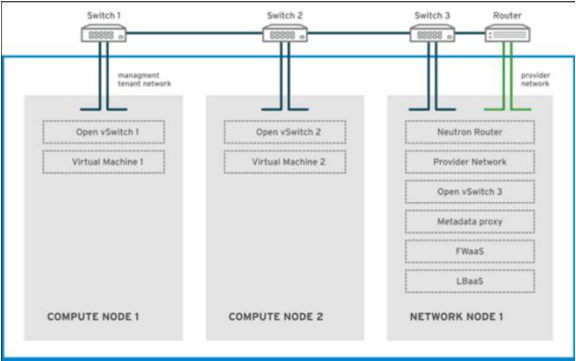
根据提示输入 10 和 11 的 root 的密码

最后的提示: **** Installation completed successfully ****

六 部署 Openstack-网络配置

6.1 网络拓扑

多计算节点的拓扑



6.2 查看外部 OVS 网桥

br-ex 为外部 OVS 网桥

```
openstack ~]# cat /etc/sysconfig/network-scripts/ifcfg-br-ex
ONBOOT="yes"

NM_CONTROLLED="no"

IPADDR="192.168.1.10"

NETMASK="255.255.255.0"

GATEWAY="192.168.1.254"

DEVICE=br-ex

NAME=br-ex

DEVICETYPE=ovs

OVSBOOTPROTO="static"

TYPE=OVSBridge
```

6.3 查看外部 OVS 网桥端口

eth0 为外部 OVS 网桥的端口

```
openstack ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0

DEVICE=eth0

NAME=eth0

DEVICETYPE=ovs
```

TYPE=OVSPort

OVS_BRIDGE=br-ex

ONBOOT=yes

BOOTPROTO=none

6.4 验证 OVS 配置: `ovs-vsctl show`

```
openstack ~]# ovs-vsctl show
```

七 OpenStack 操作基础-Horizon 配置

7.1 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.

可以在控制面板中使用终端(console)或 VNC 直接访问实例

基于 python 的 django web 框架进行开发

7.2 功能与特点

实例管理: 创建\种植实例,查看终端日志,VNC 连接,添加卷等

访问与安全管理: 创建安全群组,管理密钥对,设置浮动 IP 等

偏好设定: 对虚拟硬件模板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

对象存储处理：创建\删除容器和对象

7.3 Horizon

Horizon BUG 处理

```
openstack ~]# vim /etc/httpd/conf.d/15-horizon_vhost.conf
```

```
35 WSGIProcessGroup apache
```

```
36 WSGIApplicationGroup %{GLOBAL}    #35 行下添加此行语句
```

```
openstack ~]# apachectl -t          #检查 apache 配置文件语法
```

Syntax OK

```
openstack ~]# apachectl graceful    #重新载入配置文件
```

```
openstack ~]# systemctl reload httpd #重新加载 httpd 服务
```

7.4 基本配置

浏览器地址栏 <http://192.168.4.10/>

用户名 **admin**, 密码 **a**

八 OpenStack 操作基础-项目管理

8.1 基本管理

8.2 在 Horizon 中管理项目

8.3 命令行接口基础

初始化环境变量

```
openstack ~]# source keystone_admin
```

```
openstack ~(keystone_admin)]# env | grep OS
```

```
HOSTNAME=openstack
```

```
OS_REGION_NAME=RegionOne
```

```
OS_PASSWORD=a
```

```
OS_AUTH_URL=http://192.168.1.10:5000/v2.0
```

```
OS_USERNAME=admin
```

```
OS_TENANT_NAME=admin
```

使用帮助

```
openstack ~(keystone_admin)]# openstack help
```

8.4 通过命令行管理项目