

# 01.系统环境准备

参考任务时间： 60min

(原教材第 4 章节，因为 CentOS、Openstack 版本更新的缘故，教材中部分配置不再适用)

(!! 文档内较长命令会换行，请在输入时自行修正为一行!!)

系统环境准备 CentOS7.5 + OpenStack Rocky

注意：原教材部署架构使用 3 节点架构方案，总需 5 台物理主机；我们实验使用 1 台虚拟机(virtualbox+CentOS)，Allinone 模式，即所有服务部署在一个机器上。

考官方文档：

<https://docs.openstack.org/install-guide/openstack-services.html>

## 1.0. 实验步骤概述

### 1.0.系统环境

#### 1.1.配置域名解析

- 1) 配置主机名
- 2) 配置主机名解析

#### 1.2.关闭防火墙和 selinux

- 1) 关闭 iptables
- 2) 关闭 selinux

#### 1.3.配置时间同步

- 1) 在控制端配置时间同步服务
- 2) 编辑配置文件确认有以下配置
- 3) 重启 ntp 服务，并配置自启动
- 4) 同步时间

#### 1.4.配置相关 yum 源

- 1) 配置阿里云的 base 和 epel 源

- 2) 安装 **openstack-rocky** 的仓库
- 3) 更新软件包
- 4) 安装 **openstack** 客户端相关软件

#### 1.5.在控制节点安装数据库

- 1) 安装 **mariadb** 相关软件包
- 2) 创建 **openstack** 的数据库配置文件
- 3) 启动数据库设置开机启动
- 4) 初始化数据库并重新启动
- 5) 创建 **openstack** 相关数据库

#### 1.6.在控制节点安装消息队列 **rabbitmq**

- 1) 安装 **rabbitmq-server**
- 2) 启动 **rabbitmq**，并配置自启动
- 3) 创建消息队列中 **openstack** 账号及密码

#### 1.7.在控制节点上安装 **Memcached**

- 1) 安装 **Memcached** 用于缓存令牌
- 2) 修改 **memcached** 配置文件
- 3) 启动 **memcached** 并设置开机自启动

#### 1.8.在控制节点上安装 **Etcd** 服务

- 1) 安装 **etcd** 服务
- 2) 修改 **etcd** 配置文件
- 3) 启动 **etcd** 并设置开机自启动

## 1.0.系统环境

(1) 生产测试应用的服务器都是物理机，虚拟机 VirtualBox 仅可用于搭建测试及学习

(2) 系统选择是目前的最新版本：CentOS Linux 7.5 + Rocky

(3) 控制节点 Controller 、计算节点 Compute 均部署在一台计算机上， IP 地址：  
192.168.56.126

根据以下图形，理解实验部署与实际生产部署的差异：



## 1.1.配置域名解析

### 1) 配置主机名

主机名设置好后不可修改，较多组件使用主机名作为参数，若修改主机名则相应组件会出错；控制节点、计算节点配置相同，且均需要设置。

设置主机名

```
hostnamectl set-hostname controller
```

添加主机映射

```
cat << EOF >> /etc/hosts  
  
192.168.56.126 controller  
  
EOF
```

添加主机映射（另一方法）

```
vi /etc/hosts  
  
-----  
  
192.168.56.126    controller  
  
-----
```

配置主机名和集群内部角色名称，方便后续配置

## 1.2.关闭防火墙和 selinux

1) 关闭 iptables

# 在 CentOS7 上面是 firewalld

```
systemctl stop firewalld.service  
  
systemctl disable firewalld.service  
  
systemctl status firewalld.service
```

2) 关闭 selinux

```
setenforce 0  
  
getenforce  
  
sed -i 's#SELINUX=enforcing#SELINUX=disabled#g' /etc/sysconfig/selinux grep SELINUX=disabled /etc/sysconfig/selinux
```

## 1.3.配置时间同步

1) 在控制端配置时间同步服务

```
yum install chrony -y
```

2) 编辑配置文件确认有以下配置

```
vi /etc/chrony.conf
```

```
-----  
server ntp1.aliyun.com iburst  
server ntp2.aliyun.com iburst  
allow 192.168.0.0/16  
-----
```

3) 重启 ntp 服务，并配置开机自启动

```
systemctl restart chronyd.service
```

```
systemctl status chronyd.service
```

```
systemctl enable chronyd.service
```

```
systemctl list-unit-files |grep chronyd.service
```

4) 设置时区，同步时间

```
timedatectl set-timezone Asia/Shanghai
```

```
chronyc sources
```

```
timedatectl status
```

# 配置完成，示例如下

```
[root@controller ~]# chronyc sources
```

```
210 Number of sources = 6
```

MS Name/IP address	Stratum	Poll	Reach	LastRx	Last sample
--------------------	---------	------	-------	--------	-------------

```
=====
```

^ 85.199.214.101	1	8	373	185	+36ms[ +36ms] +/- 184ms
------------------	---	---	-----	-----	-------------------------

^ ntp8.flashdance.cx	2	8	304	571	+56ms[ +56ms] +/- 265ms
----------------------	---	---	-----	-----	-------------------------

^- ntp7.flashdance.cx	2	8	10	840	+42ms[ +41ms] +/-	286ms
^- ns1.do.steersnet.org.uk	2	7	357	117	+61ms[ +61ms] +/-	212ms
^* 120.25.115.20	2	8	377	1	+5665us[+6357us] +/-	41ms
^+ 203.107.6.88	2	8	377	2	-93us[ +598us] +/-	57ms

```
[root@controller ~]# timedatectl status
```

```
Local time: Thu 2019-02-14 15:14:30 CST
```

```
Universal time: Thu 2019-02-14 07:14:30 UTC
```

```
RTC time: Thu 2019-02-14 07:15:56
```

```
Time zone: Asia/Shanghai (CST, +0800)
```

```
NTP enabled: yes
```

```
NTP synchronized: yes
```

```
RTC in local TZ: no
```

```
DST active: n/a
```

## 1.4.配置相关 yum 源

1) 配置阿里云的 base 和 epel 源

```
mv -f /etc/yum.repos.d/CentOS-Base.repo
/etc/yum.repos.d/CentOS-Base.repo.backup
```

```
wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-7.repo
```

```
mv /etc/yum.repos.d/epel.repo /etc/yum.repos.d/epel.repo.backup
```

```
wget -O /etc/yum.repos.d/epel.repo http://mirrors.aliyun.com/repo/epel-7.repo
```

2) 安装 openstack-rocky 的仓库

```
yum install centos-release-openstack-rocky -y
```

```
yum clean all
```

```
yum makecache
```

# 手动 OpenStack 的阿里云 yum 源地址

```
vi /etc/yum.repos.d/CentOS-OpenStack-Rocky.repo
```

-----

[centos-openstack-rocky]

name=CentOS-7 - OpenStack rocky

baseurl=http://mirrors.aliyun.com/centos/7/cloud/\$basearch/openstack-rocky/

gpgcheck=1

enabled=1

gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-SIG-Cloud

[centos-openstack-rocky-test]

name=CentOS-7 - OpenStack rocky Testing

baseurl=http://mirrors.aliyun.com/centos/7/cloud/\$basearch/openstack-rocky/

gpgcheck=0

enabled=0

[centos-openstack-rocky-debuginfo]

name=CentOS-7 - OpenStack rocky - Debug

baseurl=http://mirrors.aliyun.com/centos/7/cloud/\$basearch/

gpgcheck=1

enabled=0

gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-SIG-Cloud

[centos-openstack-rocky-source]

name=CentOS-7 - OpenStack rocky - Source

baseurl=http://mirrors.aliyun.com/centos/7/cloud/\$basearch/openstack-rocky/

gpgcheck=1

enabled=0

gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-SIG-Cloud

[rdo-trunk-rocky-tested]

name=OpenStack rocky Trunk Tested

baseurl=http://mirrors.aliyun.com/centos/7/cloud/\$basearch/rdo-trunk-rocky-tested/

gpgcheck=0

enabled=0

-----

3) 更新软件包

```
yum update -y
```

4) 安装 openstack 客户端相关软件

```
yum install python-openstackclient openstack-selinux -y
```

## 1.5.在控制节点安装数据库

可以修改系统内核更改最大连接数和文件句柄数

4.修改文件句柄数量(避免 socket 连接错误)

```
ulimit -SHn 65536
```

```
vi /etc/security/limits.conf
```

添加如下内容

-----

\* hard nfile 65536

\* soft nfile 65536

-----

增加交换空间(有 swap 跳过此步)

```
mkdir /data
```

```
dd if=/dev/zero of=/data/swap bs=1024 count=4096000
```

```
mkswap /data/swap
```

```
swapon /data/swap
```



```
vi /etc/fstab
```

增加如下内容

```
-----  
/data/swap    swap        swap        defaults    0 0  
-----
```

1) 安装 mariadb 相关软件包

# CentOS7.5 默认数据库为 maraidb

```
yum install mariadb mariadb-server MySQL-python python2-PyMySQL -y
```

2) 创建 openstack 的数据库配置文件

```
vi /etc/my.cnf.d/mariadb_openstack.cnf
```

# 在[mysqld]添加以下配置

```
-----  
[mysqld]  
  
bind-address = 0.0.0.0  
  
default-storage-engine = innodb  
  
innodb_file_per_table = on  
  
max_connections = 4096  
  
collation-server = utf8_general_ci  
  
character-set-server = utf8  
  
init-connect = 'SET NAMES utf8'  
-----
```

3) 启动数据库设置开机启动

```
systemctl start mariadb.service
```

```
systemctl status mariadb.service
```

```
systemctl enable mariadb.service
```

#### 4) 初始化数据库并重新启动

# 设置密码，默认密码为空；然后输入密码 123456，一路 y 回车

```
/usr/bin/mysql_secure_installation
```

```
systemctl restart mariadb.service
```

#### 5) 创建 openstack 相关数据库

测试下数据库，相关的数据库在需要时单独创建

-----

```
mysql -u root -p123456
```

```
flush privileges;
```

```
show databases;
```

```
select user,host from mysql.user;
```

```
exit
```

-----

数据库配置完毕

## 1.6.在控制节点安装消息队列 RABBITMQ

消息队列（MQ）全称为 Message Queue，是一种应用程序对应用程序的通信方法，应用程序通过读写出入队列的消息来通信。

消息传递指的是程序之间通过在消息中发送数据进行通信，而不是通过直接调用彼此来通信，直接调用通常是用于诸如远程过程调用的技术。

RabbitMQ 是一个在 AMQP 基础上完整的，可复用的企业消息系统，遵循 Mozilla Public License 开源协议。

#### 1) 安装 rabbitmq-server

```
yum install rabbitmq-server -y
```

#### 2) 启动 rabbitmq，并配置自启动

```
systemctl start rabbitmq-server.service
```

```
systemctl status rabbitmq-server.service
```

```
systemctl enable rabbitmq-server.service
```

```
systemctl list-unit-files |grep rabbitmq-server.service
```

3) 创建消息队列中 **openstack** 账号及密码 (设置不成功, 通常是 **hostname** 配置问题, 请重启系统)

# 添加 **openstack** 用户和密码, 配置用户权限, 配置读, 写权限

```
rabbitmqctl add_user openstack openstack
```

```
rabbitmqctl set_permissions openstack ".*" ".*" ".*"
```

# rabbitmq 配置完毕

### 1.7.在控制节点上安装 Memcached

认证服务认证缓存使用 **Memcached** 缓存令牌。缓存服务 **memcached** 运行在控制节点。  
在生产部署中, 推荐启用防火墙、认证和加密保证它的安全。

1) 安装 **Memcached** 用于缓存令牌

```
yum install memcached python-memcached -y
```

2) 修改 **memcached** 配置文件

```
vi /etc/sysconfig/memcached
```

-----

```
OPTIONS="-l 127.0.0.1,controller"
```

-----

# 如果没有启用 **IPv6** 地址需要删掉::1 的地址绑定

3) 启动 **memcached** 并设置开机自启动

```
systemctl start memcached.service
```

```
systemctl status memcached.service
```

```
netstat -anptl | grep memcached
```

```
systemctl enable memcached.service
```

```
systemctl list-unit-files | grep memcached.service
```

# 至此, **memcached** 配置完毕

### 1.8.在控制节点上安装 Etcd 服务

# 这个 Etcd 服务是新加入的，用于自动化配置。

Etcd 是一个分布式一致性键值存储系统，用于共享配置和服务发现。

### 1) 安装 etcd 服务

```
yum install etcd -y
```

### 2) 修改 etcd 配置文件

```
vi /etc/etcd/etcd.conf
```

增加如下内容：

```
-----  
#[Member]  
ETCD_DATA_DIR="/var/lib/etcd/default.etcd"  
ETCD_LISTEN_PEER_URLS="http://192.168.56.126:2380"  
ETCD_LISTEN_CLIENT_URLS="http://192.168.56.126:2379"  
ETCD_NAME="controller"  
#[Clustering]  
ETCD_INITIAL_ADVERTISE_PEER_URLS="http://192.168.56.126:2380"  
ETCD_ADVERTISE_CLIENT_URLS="http://192.168.56.126:2379"  
ETCD_INITIAL_CLUSTER="controller=http://192.168.56.126:2380"  
ETCD_INITIAL_CLUSTER_TOKEN="etcd-cluster-01"  
ETCD_INITIAL_CLUSTER_STATE="new"  
-----
```

### 3) 启动 etcd 并设置开机自启动

```
systemctl start etcd.service  
systemctl status etcd.service  
netstat -anptl|grep etcd  
systemctl enable etcd.service  
systemctl list-unit-files | grep etcd.service
```

# 节点 controller 完成基础环境配置

完成实验