**Hdfs+Zookeeper+Mesos+Chronos部署指导手册**

目录

[一、部署准备： 2](#_Toc519606613)

[1.物理设备规划 2](#_Toc519606614)

[2.虚拟机配置（单台） 2](#_Toc519606615)

[3.软件信息 2](#_Toc519606616)

[4.部署规划 2](#_Toc519606617)

[二、环境装备 3](#_Toc519606618)

[1.修改主机名 3](#_Toc519606619)

[2.配置本地dns域名解析 3](#_Toc519606620)

[3.关闭防火墙 3](#_Toc519606621)

[4.关闭selinux 3](#_Toc519606622)

[5.给部署用户提升权限 4](#_Toc519606623)

[6.配置免秘钥登录 4](#_Toc519606624)

[三、部署hdfs 4](#_Toc519606625)

[1.安装hdfs 4](#_Toc519606626)

[2.修改hdfs配置文件 5](#_Toc519606627)

[3.初始化name node 6](#_Toc519606628)

[4.启动hdfs服务 6](#_Toc519606629)

[5.客户端访问验证 8](#_Toc519606630)

[四、部署Mesos、zookeeper 9](#_Toc519606631)

[1.安装依赖 9](#_Toc519606632)

[2.安装mesos 9](#_Toc519606633)

[3.配置Zookeeper 10](#_Toc519606634)

[4.配置master和slave节点 11](#_Toc519606635)

[5.启动mesos集群 13](#_Toc519606636)

[6.客户端访问验证 13](#_Toc519606637)

[五、部署chronos 14](#_Toc519606638)

[1.安装chronos 14](#_Toc519606639)

[2.客户端访问验证 15](#_Toc519606640)

# 一、部署准备：

## 1.物理设备规划

|  |  |  |
| --- | --- | --- |
| **设备** | **数量** | **操作系统** |
| 虚拟机 | 3 | ubuntu 18.04 |

## 2.虚拟机配置（单台）

|  |  |
| --- | --- |
| **设备** | **配置** |
| CPU | 核心数：2核\*4；频率：2600MHz； 缓存：15360KB；型号：Intel(R) Xeon(R) CPU E5-2630 v2 @ 2.60GHz |
| 内存 | 物理内存：5960M；交换内存：2047M |
| 硬盘 | 500G |
| 网卡 | 以太网卡1个 |

## 3.软件信息

|  |  |  |  |
| --- | --- | --- | --- |
| **软件** | **版本** | **来源** | **安装方式** |
| openjdk-8-jdk | 1.8.0\_171 | ubuntu18.04自带 | apt-get install |
| hadoop-3.0.3.tar.gz | 3.0.3 | http://hadoop.apache.org/releases.html | 压缩包解压 |
| mesos-1.6.0.tar.gz | 1.6.0 | http://mesos.apache.org/downloads/ | 源码包编译 |
| zookeeper-3.4.8 | 3.4.8 | mesos-1.6.0.tar.gz中包含，编译mesos时，也跟随编译 | 源码包编译 |
| chronos | 3.0.3 | https://github.com/mesos/chronos.git | 源码包编译 |

## 4.部署规划

|  |  |  |
| --- | --- | --- |
| **IP地址** | **主机名** | **安装服务** |
| 10.10.9.17 | master | hdfs-name dode，mesos-master，mesos-slave(mesos-agent)，zookeeper，chronos |
| 10.10.9.18 | slave1 | hdfs-data node，mesos-slave(mesos-agent) |
| 10.10.9.19 | slave2 | hdfs-data node，mesos-slave(mesos-agent) |

# 二、环境装备

## 1.修改主机名

# hostnamectl set-hostname master //10.10.9.17节点上执行

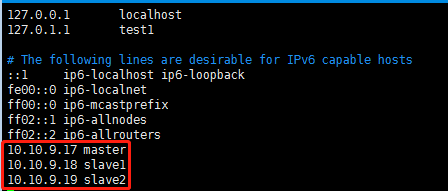
# hostnamectl set-hostname slave1 //10.10.9.18节点上执行

# hostnamectl set-hostname slave2 //10.10.9.19节点上执行

## 2.配置本地dns域名解析

（所有节点）

# vi /etc/hosts



检测本地dns域名解析：

在master、slave1、slave2三台机器上依次执行ping master; ping slave1; ping slave2。能够互相通信即可。

## 3.关闭防火墙

（所有节点）

# service iptables status //查看防火墙状态

# service iptables stop //关闭防火墙

# chkconfig iptables –list //查看防火墙开机启动状态

# chkconfig iptables off //关闭防火墙开机启动

## 4.关闭selinux

（所有节点）

#getenforce //查看selinux状态，0为关闭，1为开启

#setenforce 0 //临时关闭selinux，重启后失效

#vi /etc/selinux/config 将SELINUX=enforcing改为SELINUX=disabled，保存退出 //永久关闭selinux，需要重启生效，如果执行了setenforce 0则不需要重启。

## 5.给部署用户提升权限

（所有节点）

# su root

# vim /etc/sudoers //本次安装用root用户，如果是普通用户需要百度此方法

## 6.配置免秘钥登录

（在master节点上执行）

# cd /root/.ssh/

# ssh-keygen -t rsa （四个回车） //生成公、私钥

# ssh-copy-id localhost //拷贝秘钥到本机

# ssh-copy-id slave1 //拷贝秘钥到slave1

# ssh-copy-id slave2 //拷贝秘钥到slave2

# 三、部署hdfs

## 1.安装hdfs

（所有节点）

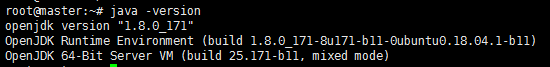
在Apache官网http://hadoop.apache.org/releases.html下载hadoop软件包，拷贝到各节点/root/目录下



# apt-get upgrade //更新包源

# apt-get install openjdk-8-jdk //安装Java8

# java –version //检查Java版本



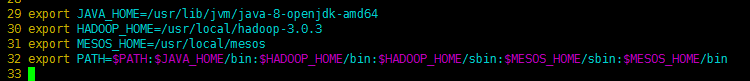
# cd /root/

# mkdir app

# tar -zxf hadoop-3.0.3.tar.gz -C app/

# mv app/hadoop-3.0.3 /usr/local/

# vi /etc/profile //在末尾增加环境变量



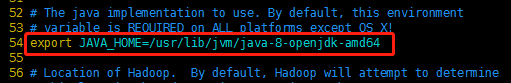
# source /etc/profile //加载环境变量

## 2.修改hdfs配置文件

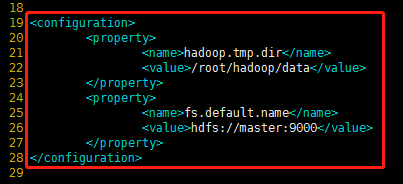
（所有节点）

# cd /usr/local/hadoop-3.0.3/

# vi etc/hadoop/hadoop-env.sh //添加Java目录



# vi etc/hadoop/core-site.xml //在末尾添加“指定hadoop运行时产生文件的存储目录”和“指定HADOOP所使用的文件系统schema（URI），HDFS的老大（NameNode）的地址”。



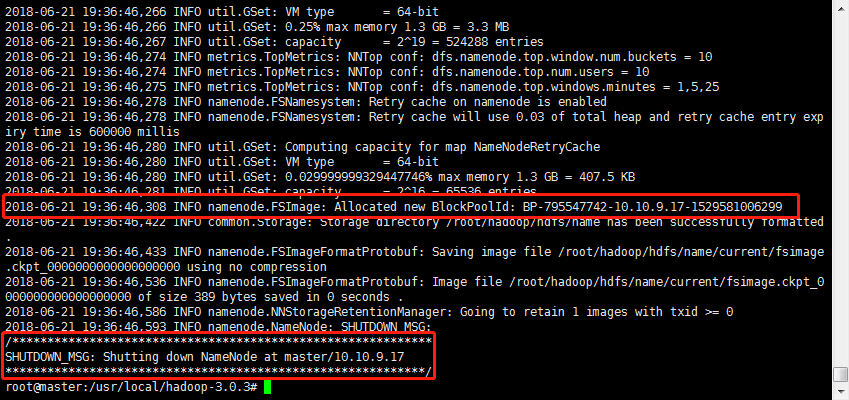
# vi etc/hadoop/hdfs-site.xml //在末尾添加“副本数”、“name node的工作目录”、 “data node的工作目录”、“name node的URL”和“name node备份secondary的URL”。



## 3.初始化name node

（仅在name node节点（即master节点）上执行）

# hdfs namenode -format

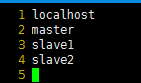


## 4.启动hdfs服务

（仅在name node节点（即master节点）上执行）

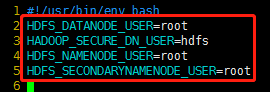
# cd /usr/local/hadoop-3.0.3/etc/hadoop/

# vi workers //添加集群主机名信息

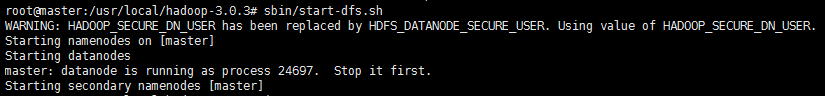


# cd /usr/local/hadoop-3.0.3/

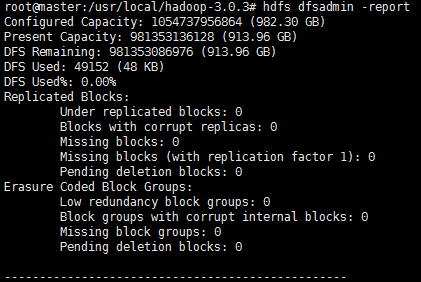
# vi sbin/start-dfs.sh //允许root身份运行

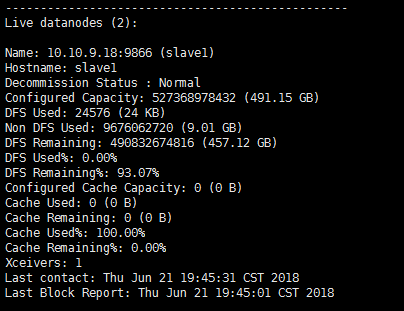


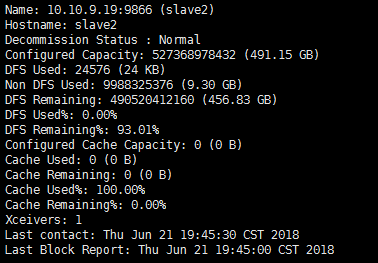
# sbin/start-dfs.sh //启动集群服务



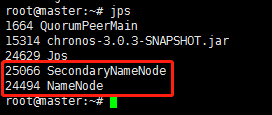
# hdfs dfsadmin –report //查看各节点服务状态







# jps //查看服务状态，在name node节点（master节点）执行

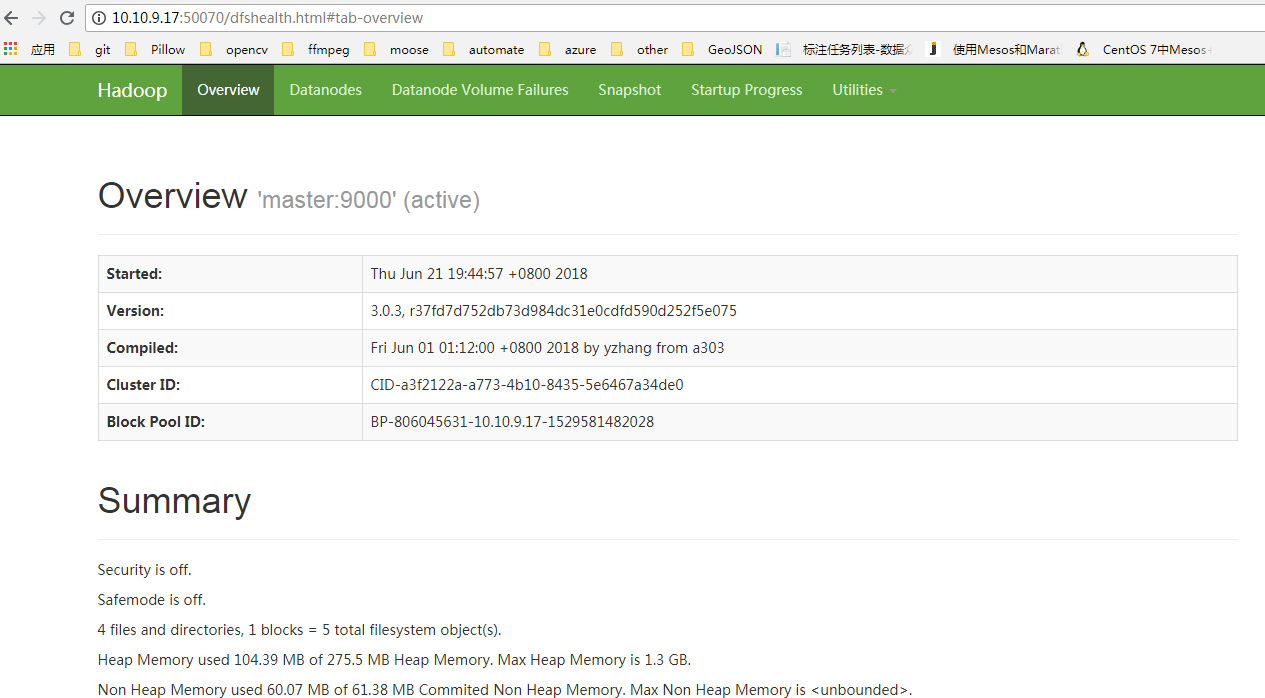


# jps //在data node节点（slave1和slave2节点）上执行



## 5.客户端访问验证

客户端访问地址：http://10.10.9.17:50070



可以看到hdfs集群信息，2个name节点均为active，至此hdfs集群部署完毕。

注：页面暂不支持上传，仅支持下载，下载时需要将slave1和slave2的域名手动改为IP地址，或者修改本地dns。

# 四、部署Mesos、zookeeper

## 1.安装依赖

（所有节点）

# sudo apt-get update

# sudo apt-get install -y openjdk-8-jdk //部署hdfs时已安装

# sudo apt-get -y install build-essential python-dev python-six python-virtualenv libcurl4-nss-dev libsasl2-dev libsasl2-modules maven libapr1-dev libsvn-dev zlib1g-dev iputils-ping python-pip

# pip install --upgrade pip //升级pip到10版本

## 2.安装mesos

（所有节点）

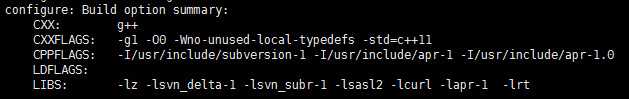
# cd /root

# tar -zxf mesos-1.6.0.tar.gz //在官网下载，放在/root/目录下

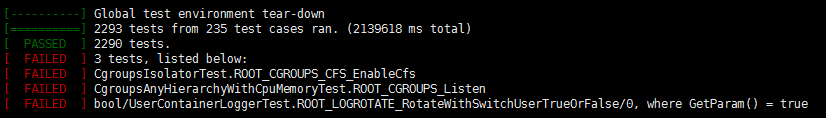
# cd mesos-1.6.0/

# mkdir build && cd $\_

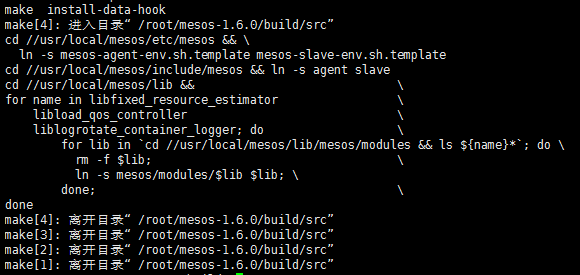
# ../configure --prefix="/usr/local/mesos"



# make check //时间很久，结果中的错误test可忽略



# make install



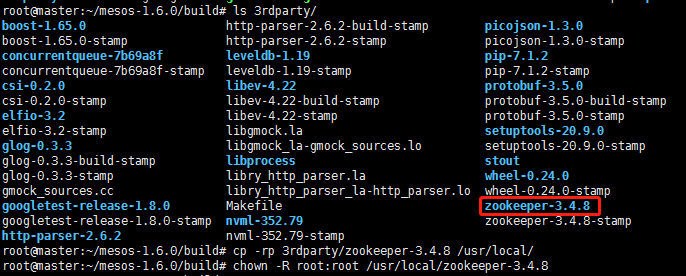
## 3.配置Zookeeper

（仅master节点）

# cd /root/mesos-1.6.0/build

# cp -rp 3rdparty/zookeeper-3.4.8 /usr/local/

# chown -R root:root /usr/local/zookeeper-3.4.8



# cd /usr/local/zookeeper-3.4.8/conf/

# cp zoo\_sample.cfg zoo.cfg

# vi zoo.cfg //修改以下参数

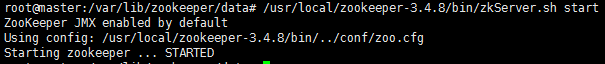
 tickTime=2000  
 initLimit=10  
 syncLimit=5  
 dataDir=/var/lib/zookeeper/data  
 dataLogDir=/var/log/zookeeper/logs  
 clientPort=2181  
 server.1=master:2888:3888

# mkdir -p /var/lib/zookeeper/data /var/log/zookeeper/logs

# cd /var/lib/zookeeper/data/

# echo '1' > myid

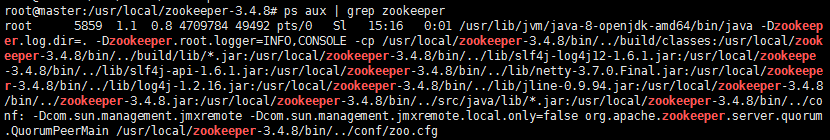
# /usr/local/zookeeper-3.4.8/bin/zkServer.sh start //启动zookeeper



# echo ruok | nc 10.10.9.17 2181 //检查zookeeper，回复“imok”则正常



# ps aux | grep zookpeer //查看zookeeper进程



## 4.配置master和slave节点

（方式相同）

# cd /usr/local/mesos/etc/mesos/

# cp mesos-master-env.sh.template mesos-master-env.sh

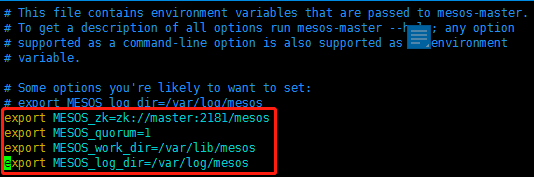
# vi mesos-master-env.sh //添加以下参数

export MESOS\_zk=zk://master:2181/mesos

export MESOS\_quorum=1

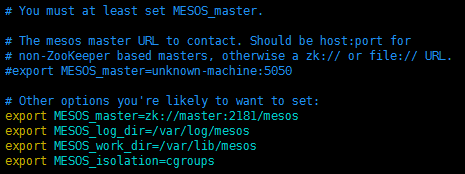
export MESOS\_work\_dir=/var/lib/mesos

export MESOS\_log\_dir=/var/log/mesos



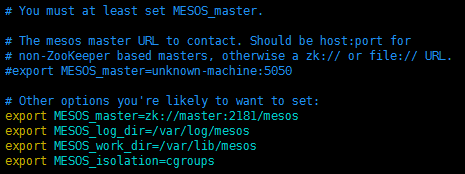
# cp mesos-slave-env.sh.template mesos-slave-env.sh

#vi mesos-slave-env.sh



# cp mesos-agent-env.sh.template mesos-agent-env.sh

#vi mesos-agent-env.sh //内容同mesos-slave-env.sh

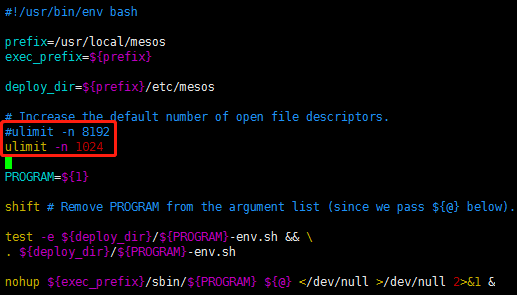


#echo “master” > masters

#vi slaves

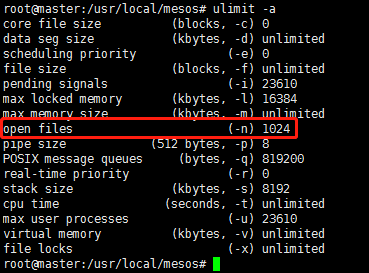


#vi /usr/local/mesos/sbin/mesos-daemon.sh

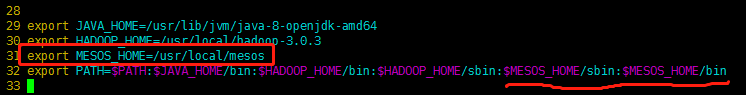


注：此步骤将ulimit -n 8192修改为ulimit -n 1024

这条指令是在向服务器索要资源，但是 通过ulimit -a查看可以看出-n为1024,系统要求这个数不能更大，因此把8192更改为1024即可



#vi /etc/profile //增加mesos环境变量



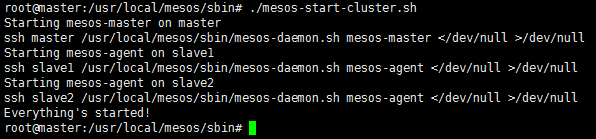
# source /etc/profile

## 5.启动mesos集群

（在master节点执行）

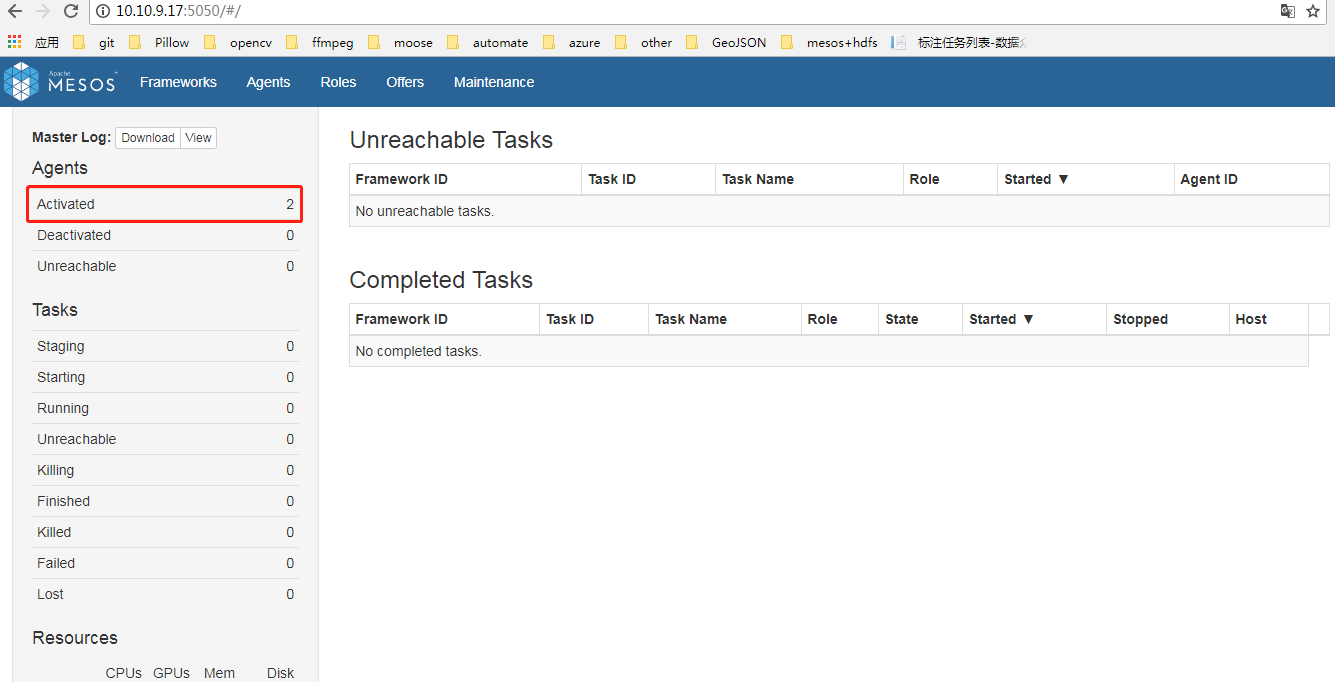
#cd /usr/local/mesos/sbin

# ./mesos-start-cluster.sh



## 6.客户端访问验证

在客户端访问<http://10.10.9.17:5050>



看到两个slave均为active，至此mesos部署完毕

# 五、部署chronos

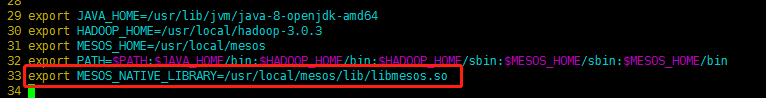
## 1.安装chronos

（仅在master节点）

# apt-get –y install git npm //安装依赖包

# export MESOS\_NATIVE\_LIBRARY=/usr/local/mesos/lib/libmesos.so //添加环境变量，临时生效，需要find查出libmesos.so的绝对路径

#vi /etc/profile //添加永久环境变量



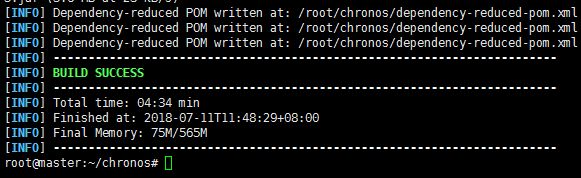
# cd /root

# git clone https://github.com/mesos/chronos.git

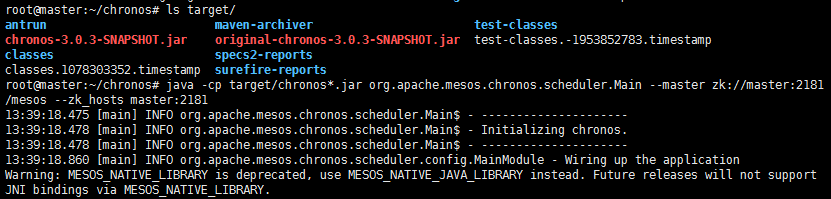
# cd chromos

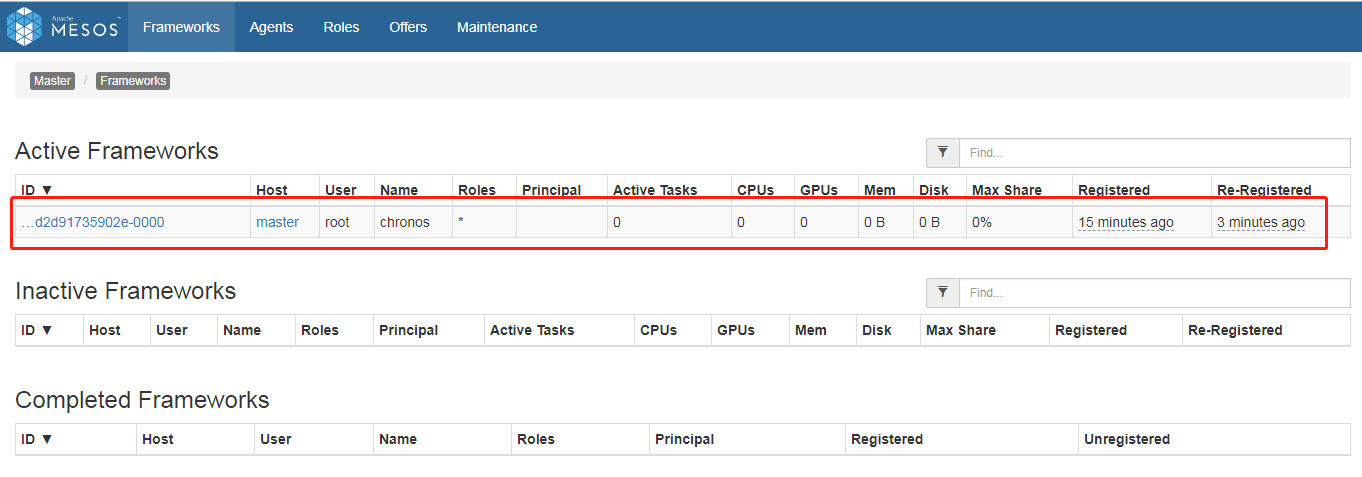
# mvn package -Dmaven.test.skip=true //打chronos jar包，跳过单元测试





#java -cp target/chronos\*.jar org.apache.mesos.chronos.scheduler.Main --master zk://master:2181/mesos --zk\_hosts master:2181 //检查服务，mesos页面frameworks上显示active后执行ctrl+c结束





# java -jar target/chronos\*.jar --master zk://master:2181/mesos --zk\_hosts master:2181 --http\_port 4400 &> /dev/null & //后台启动chronos服务

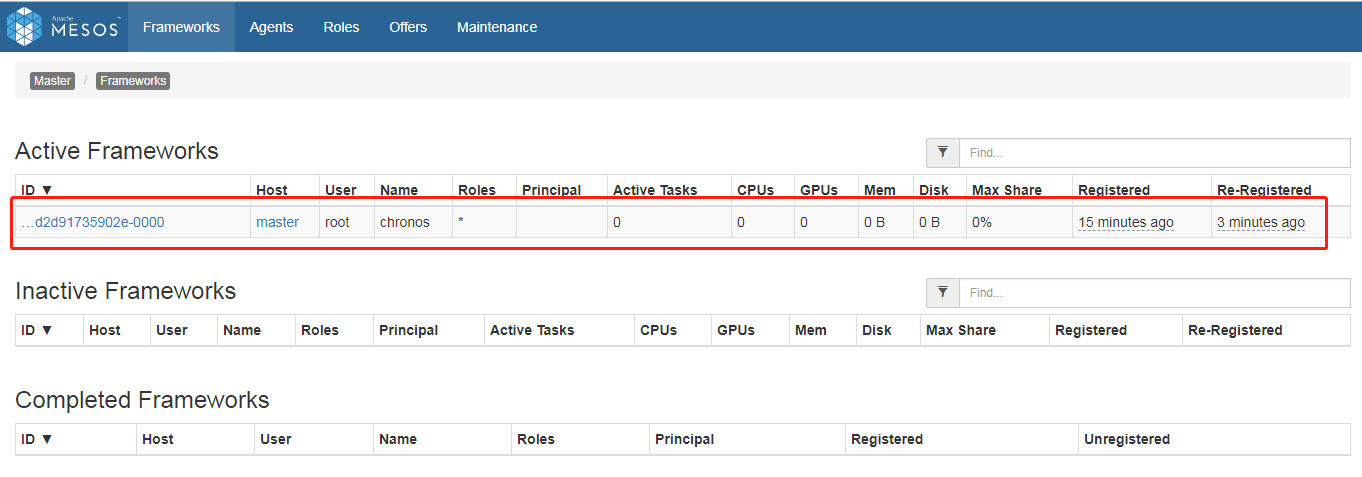


# ps aux | grep chronos //检查chronos服务

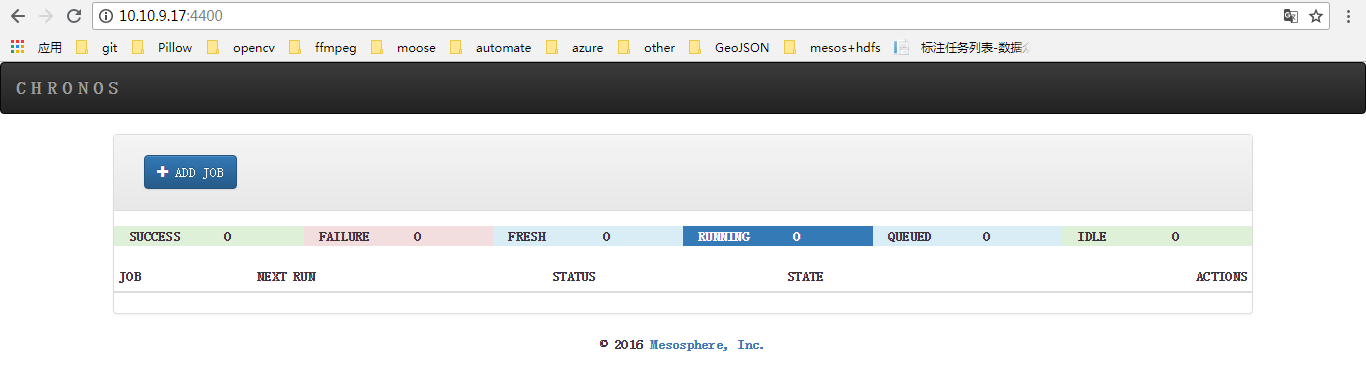


## 2.客户端访问验证

1.刷新mesos页面，frameworks上显示active



2.客户端访问http://10.10.9.17:4400



chromos页面显示正常。至此，chronos编译安装结束