# Hadoop 分布式集群配置

### 环境

一共六台服务器(包括三台腾讯云主机/三台阿里云主机).

服务器环境为 Ubuntu 14.04 64 位 跟 CentOS 7.2 64 位.

使用 Mobaxterm 作为远程连接工具.

使用 hadoop 版本: hadoop 2.7.

使用腾讯云主机 CentOS 7.2 64 位系统作为 Master机,其他作为 Slave 机.

### 准备工作

六台服务器都需要配置好Java环境以及安装ssh.

为了方便管理起见,给所有服务器增加一个 hadoop 账户,并添加到 sudoers 组.

#### 安装Java

sudo apt-get install openjdk-7-jre openjdk-7-jdk  
# Ubuntu  
或者  
sudo yum install java-1.7.0-openjdk java-1.7.0-openjdk-devel  
# CentOS

#### 配置 JAVA\_HOME 路径

dpkg -L openjdk-7-jdk | grep '/bin/javac'  
# Ubuntu  
或者  
rpm -ql java-1.7.0-openjdk-devel | grep '/bin/javac'  
# CentOS

将输出的 Java 安装路径添加到 ~/.bashrc,设置 linux 环境变量.

vim ~/.bashrc  
# 然后将 export JAVA\_HOME=安装路径 命令添加到 .bashrc 中.  
source ~/.bashrc  
# 使修改后的配置生效.



#### 配置hadoop用户

创建新用户 hadoop,并且使用 /bin/bash 作为 shell:

useradd -m hadoop -s /bin/bash

配置用户密码:

passwd hadoop  
# 配置的六台服务器密码一致,便于管理

为 hadoop 用户增加管理员权限:

sudo adduser hadoop sudo  
# Ubuntu

visudo  
# CentOS 在 root 账户下执行

在打开的文件中,找到 root ALL=(ALL) ALL 一行,在其下一行添加: hadoop ALL=(ALL) ALL.



#### 安装 ssh

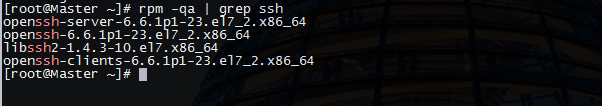
CentOS 默认安装了 SSH client、SSH server, 而Ubuntu 仅仅默认安装了 SSH client,还需要另外安装 SSH server .

sudo apt-get install openssh-server  
# Ubuntu

对于 CentOS ,测试 SSH 安装成功:

rpm -qa | grep ssh

如果没有出现下图所示,则需要重新安装:



sudo yum install openssh-clients openssh-server

测试:执行 ssh localhost,如果可以正常访问本机,说明 SSH 安装正常.

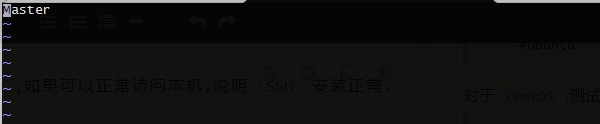
#### 网络配置

##### 修改主机名

为了直观管理各个主机,可以将每台主机根据其地位修改命名为: Master,Slave1,Slave2,Slave3,Slave4,Slave5.

执行命令:

sudo vim /etc/hostname



修改后,重启服务器,使修改生效.

##### 配置 ip 映射

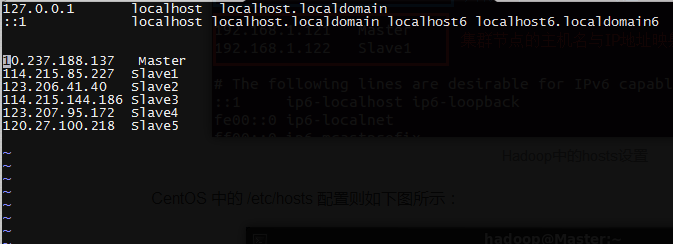
将已知的服务器 ip 及其对应的主机名写入到每台服务器的 hosts 文件:

sudo vim /etc/hosts

**注意:**

此时,在 Master 机上配置时,要使用内网 ip 对应 Master 主机名,而其他 Slave 机用公网 ip 对应 Master 主机名(否则之后无法启动 Master 机上的 namenode 等进程).

* Master 机 hosts 配置:

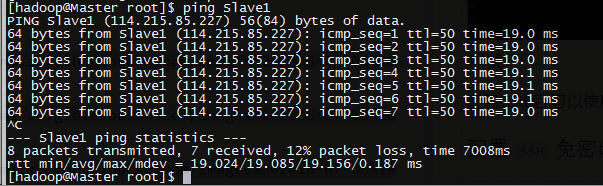


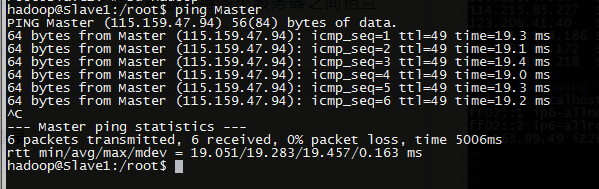
* Slave 机 hosts 配置:



##### 测试:

配置完成后,可以使用 ping 命令测试是否能够使得各个服务器之间相互 ping 通.





#### 配置 SSH 免密码登录

首先确定自己使用的用户名是 hadoop,然后执行一下命令:

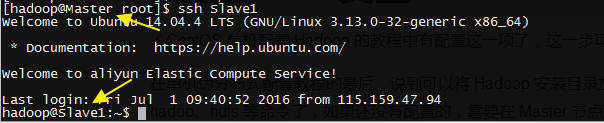
cd ~/.ssh   
# 如果没有该目录，先执行一次ssh localhost  
rm ./id\_rsa\*   
# 删除之前生成的公钥(因为修改过主机名)  
ssh-keygen -t rsa   
# 一直按回车就可以  
cat ./id\_rsa.pub >> ./authorized\_keys  
# 公匙加入授权,可以使用 ssh Master 命令测试一下  
scp ~/.ssh/id\_rsa.pub hadoop@Slave1:/home/hadoop/  
scp ~/.ssh/id\_rsa.pub hadoop@Slave2:/home/hadoop/  
scp ~/.ssh/id\_rsa.pub hadoop@Slave3:/home/hadoop/  
scp ~/.ssh/id\_rsa.pub hadoop@Slave4:/home/hadoop/  
scp ~/.ssh/id\_rsa.pub hadoop@Slave5:/home/hadoop/  
# 将公钥使用 scp 命令依次远程拷贝到其他五台 Slave 机

之后在每台 Slave 机上把公匙加入授权:

mkdir ~/.ssh   
# 如果不存在该文件夹需先创建，若已存在则忽略  
cat ~/id\_rsa.pub >> ~/.ssh/authorized\_keys

测试,能否成功在 Master 机上面无密码登录其他 Slave 机:

ssh Slave1



### 安装 hadoop

#### 下载安装

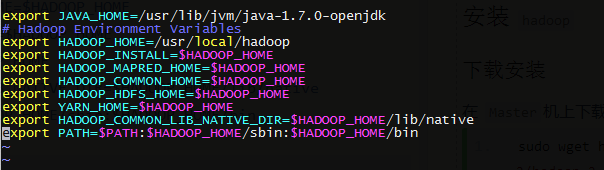
在 Master机上下载 hadoop 安装包:

sudo wget http://mirrors.cnnic.cn/apache/hadoop/common/hadoop-2.7.2/hadoop-2.7.2-src.tar.gz  
sudo tar -zxf hadoop-2.7.2-src.tar.gz -C /usr/local   
# 解压到/usr/local中  
cd /usr/local/  
sudo mv ./hadoop-2.6.0/ ./hadoop   
# 将文件夹名改为hadoop  
sudo chown -R hadoop ./hadoop   
# 修改文件权限

#### 环境变量

执行 vim ~/.bashrc,加入:

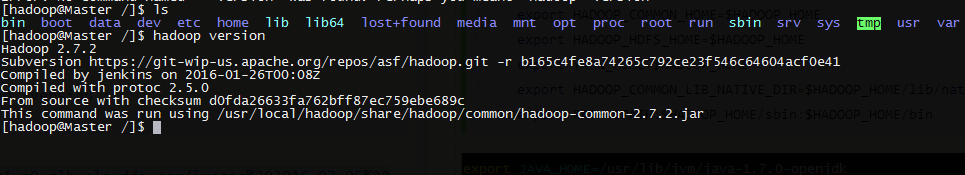
export HADOOP\_HOME=/usr/local/hadoop  
export HADOOP\_INSTALL=$HADOOP\_HOME  
export HADOOP\_MAPRED\_HOME=$HADOOP\_HOME  
export HADOOP\_COMMON\_HOME=$HADOOP\_HOME  
export HADOOP\_HDFS\_HOME=$HADOOP\_HOME  
export YARN\_HOME=$HADOOP\_HOME  
export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$HADOOP\_HOME/lib/native  
export PATH=$PATH:$HADOOP\_HOME/sbin:$HADOOP\_HOME/bin



执行 source ~/.bashrc命令使修改生效.

#### 测试

在任意路径下,执行 hadoop -version:



### 配置集群

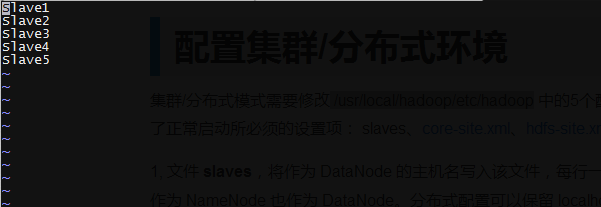
#### 配置 Master

集群/分布式模式需要修改 /usr/local/hadoop/etc/hadoop 中的5个配置文件： slaves、core-site.xml、hdfs-site.xml、mapred-site.xml、yarn-site.xml.

在 Master 机上修改一下文件:

* 文件 slaves,将作为 DataNode 的主机名写入该文件,每行一个,删除掉默认的 localhost ,加入五个 Slave 机的主机名:

vim /usr/local/hadoop/etc/hadoop/slaves



* 文件 core-site.xml 改为下面的配置:

<configuration>  
 <property>  
 <name>fs.defaultFS</name>  
 <value>hdfs://Master:9000</value>  
 </property>  
 <property>  
 <name>hadoop.tmp.dir</name>  
 <value>file:/usr/local/hadoop/tmp</value>  
 <description>Abase for other temporary directories.</description>  
 </property>  
</configuration>

* 文件 hdfs-site.xml,dfs.replication 一般设为3,但我们有五个 Slave 节点,所以 dfs.replication 的值还是设为5:

<configuration>  
 <property>  
 <name>dfs.namenode.secondary.http-address</name>  
 <value>Master:50090</value>  
 </property>  
 <property>  
 <name>dfs.replication</name>  
 <value>5</value>  
 </property>  
 <property>  
 <name>dfs.namenode.name.dir</name>  
 <value>file:/usr/local/hadoop/tmp/dfs/name</value>  
 </property>  
 <property>  
 <name>dfs.datanode.data.dir</name>  
 <value>file:/usr/local/hadoop/tmp/dfs/data</value>  
 </property>  
</configuration>

* 文件 mapred-site.xml (需要先重命名，默认文件名为 mapred-site.xml.template,执行命令 mv mapred-site.xml.template mapred-site.xml),然后配置修改如下:

<configuration>  
 <property>  
 <name>mapreduce.framework.name</name>  
 <value>yarn</value>  
 </property>  
 <property>  
 <name>mapreduce.jobhistory.address</name>  
 <value>Master:10020</value>  
 </property>  
 <property>  
 <name>mapreduce.jobhistory.webapp.address</name>  
 <value>Master:19888</value>  
 </property>  
</configuration>

* 文件 yarn-site.xml:

<configuration>  
 <property>  
 <name>yarn.resourcemanager.hostname</name>  
 <value>Master</value>  
 </property>  
 <property>  
 <name>yarn.nodemanager.aux-services</name>  
 <value>mapreduce\_shuffle</value>  
 </property>  
</configuration>

#### 配置 Slave

配置好后，将 Master 上的 /usr/local/Hadoop 文件夹复制到各个节点上:

cd /usr/local  
tar -zcf ~/hadoop.master.tar.gz ./hadoop   
# 先压缩再复制  
cd ~  
scp ./hadoop.master.tar.gz Slave1:/home/hadoop  
scp ./hadoop.master.tar.gz Slave2:/home/hadoop  
scp ./hadoop.master.tar.gz Slave3:/home/hadoop  
scp ./hadoop.master.tar.gz Slave4:/home/hadoop  
scp ./hadoop.master.tar.gz Slave5:/home/hadoop

然后在 Slave 机上面解压文件:

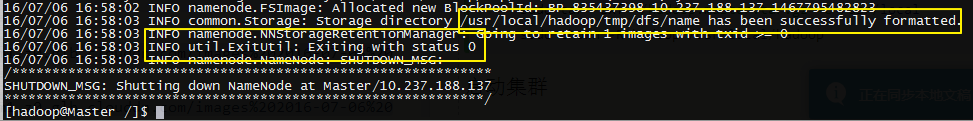
sudo rm -r /usr/local/hadoop   
# 删掉旧的（如果存在）  
sudo tar -zxf ~/hadoop.master.tar.gz -C /usr/local  
sudo chown -R hadoop /usr/local/hadoop

#### 启动集群

首次启动需要先在 Master 机执行 NameNode 的格式化:

hdfs namenode -format

成功的话,会看到 successfully formatted 和 Exitting with status 0 的提示,若为 Exitting with status 1 则是出错.



**注意1:**

CentOS 系统默认开启了防火墙,在开启 Hadoop 集群之前，需要关闭集群中每个节点的防火墙。

systemctl stop firewalld.service   
# 关闭firewall  
systemctl disable firewalld.service   
# 禁止firewall开机启动

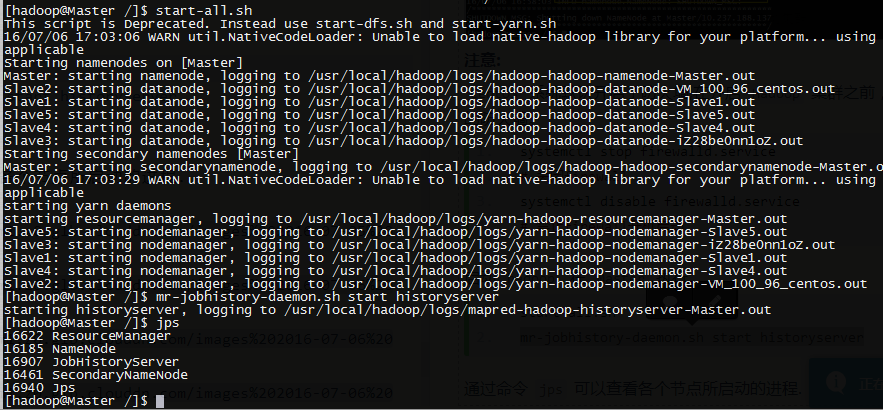
**注意2:**

如果发生过集群启动失败,尤其 DataNode 无法启动,执行 rm -rf /usr/local/hadoop/tmp/,将所有 Salve 节点上的 tmp删除,然后重新执行 NameNode 的格式化.

启动 hadoop ,在 Master 执行:

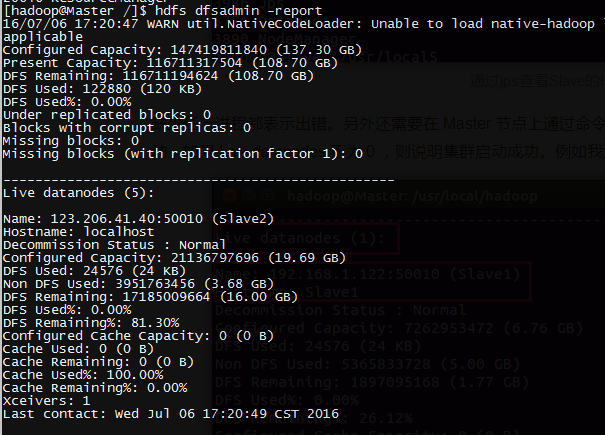
start-all.sh  
mr-jobhistory-daemon.sh start historyserver

通过命令 jps 可以查看各个节点所启动的进程.

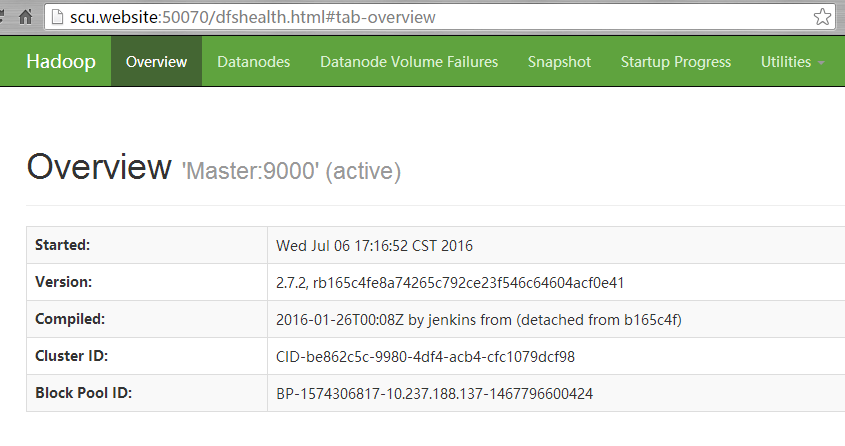




缺少任一进程都表示出错.另外还需要在 Master 机上通过命令 hdfs dfsadmin -report 查看 DataNode 是否正常启动，如果 Live datanodes 不为 0 ,则说明集群启动成功:



之后也可以通过 Web 页面 http://master ip:50070/看到查看 DataNode 和 NameNode 的状态：



至此, hadoop 集群启动成功.

在 Master 机上执行以下命令关闭 hadoop 集群:

stop-all.sh  
mr-jobhistory-daemon.sh stop historyserver