Hadoop 2.2.0 详细配置（作者：梁世威）

建议配置1台，克隆两台，修改/etc/hostname即可

## 创建hadoop用户组和用户

创建用户组

sudo addgroup hadoop

创建用户

sudo adduser -ingroup hadoop hadoop

给hadoop用户添加权限，打开/etc/sudoers文件

sudo gedit /etc/sudoers

在root ALL=(ALL)ALL行下添加ALL=(ALL:ALL) ALL。如果不添加这行，hadoop将不能执行sudo操作。

C:\Users\Administrator\AppData\Roaming\Tencent\Users\228930204\QQ\WinTemp\RichOle\8K~{RDLRQV}[KR]D{6L@]K6.jpg

1. 配置jdk

进入在存放jdk文件夹的当前目录，将解压好的jdk1.7.0\_10文件夹用最高权限复移动到/usr/lib/jvm目录里，此时即使没有jvm目录也可以执行如下命令，jvm文件夹将得到创建。

sudo mv jdk1.7.0\_45 /usr/lib/jvm/

接下来需要设置环境变量，进入当前用户名的主文件夹，修改.profile文件。注意，该文件是当前用户的配置文件，如果要对使用计算机的所有用户生效，那么需要设置的文件应该是 /etc/profile。

解压

tar zxvf jdk1.7.0\_45

配置hadoop用户环境变量

sudo gedit ~/.bashrc

在末尾加上:

# set java environment

export JAVA\_HOME=/usr/lib/jvm/jdk1.7.0\_45

export CLASSPATH=.:$CLASSPATH:$JAVA\_HOME/lib:$JAVA\_HOME/jre/lib

export PATH=$PATH:$JAVA\_HOME/bin:$JAVA\_HOME/jre/bin

# Hadoop

export HADOOP\_PREFIX="/usr/local/hadoop"

export PATH=$PATH:$HADOOP\_PREFIX/bin:$HADOOP\_PREFIX/sbin

export HADOOP\_COMMON\_HOME=${HADOOP\_PREFIX}

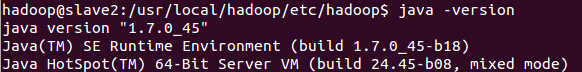
export HADOOP\_HDFS\_HOME=${HADOOP\_PREFIX}

export HADOOP\_MAPRED\_HOME=${HADOOP\_PREFIX}

export HADOOP\_YARN\_HOME=${HADOOP\_PREFIX}

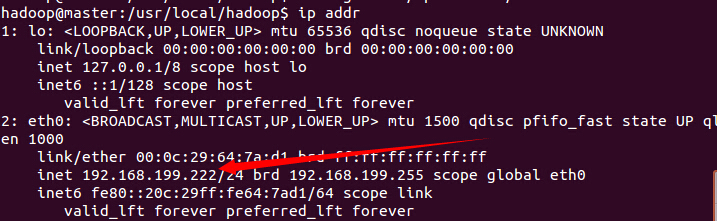
然后保存关闭，此时在命令行输入java -version将会仍然显示java没有安装。此时，需要使用source更新下.bashrc文件：

source  ~/.bashrc

再次使用java -version命令可以看到如下输出：  


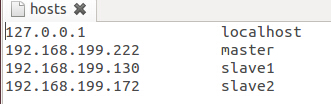
到此，已经将jdk配置完成。如果之前系统里面已经安装了openjdk，可以使用如下方法将默认jdk更改过来：

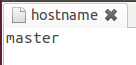
3.采用桥接模式获得静态虚拟机ip

ip addr 获得该节点ip地址

在三台主机上分别设置：/etc/hosts 和/etc/hostname

hosts这个文件用于定义主机名和IP地址之间的映射关系。



(修改hostname后重启虚拟机生效)

4.关闭防火墙

ufw disable (关闭防火墙重启虚拟机生效)

5.换源

若出现ssh等软件无法安装，进行换源操作

1、首先备份Ubuntu12.04源列表

sudo cp /etc/apt/sources.list /etc/apt/sources.list.backup （备份下当前的源列表）

2、修改更新源

sudo gedit /etc/apt/sources.list （打开Ubuntu 12.04源列表文件）

在后面添加:

deb http://ubuntu.cn99.com/ubuntu/ precise main restricted universe multiverse

deb http://ubuntu.cn99.com/ubuntu/ precise-updates main restricted universe multiverse

deb http://ubuntu.cn99.com/ubuntu/ precise-security main restricted universe multiverse

deb http://ubuntu.cn99.com/ubuntu/ precise-backports main restricted universe multiverse

deb http://ubuntu.cn99.com/ubuntu-cn/ precise main restricted universe multiverse

3、sudo apt-get update 更新

6.安装SSH服务（建立无密码登陆本机）

这里的ssh和三大框架:spring,struts,hibernate没有什么关系，ssh可以实现远程登录和管理

安装openssh-server;

sudo apt-get install ssh openssh-server

首先要转换成hadoop用户，执行以下命令：

su  hadoop

1.创建ssh-key，，这里我们采用rsa方式;

ssh-keygen -t rsa

（注：回车后会在~/.ssh/下生成两个文件：id\_rsa和id\_rsa.pub这两个文件是成对出现的）

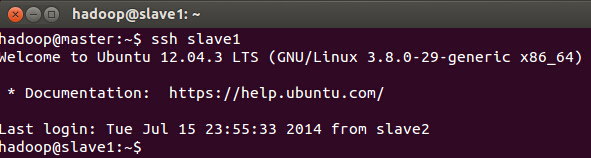
2.进入~/.ssh/目录下，将id\_rsa.pub追加到authorized\_keys授权文件中，开始是没有authorized\_keys文件的;

cd ~/.ssh

cat id\_rsa.pub >> authorized\_keys

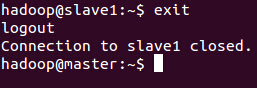
3. 登录localhost（slave1）;

ssh (localhost)slave1



( 注：当ssh远程登录到其它机器后，现在你控制的是远程的机器，需要执行退出命令才能重新控制本地主机。)

4. 执行退出命令;

Exit 

将生成的authorized\_keys文件拷贝到两台slave主机相同的文件夹下，命令如下：

scp authorized\_keys slave1:~/.ssh/

scp authorized\_keys slave2:~/.ssh/

ssh 必须安装并且保证 sshd一直运行，以便用Hadoop 脚本管理远端Hadoop守护进程。

附ssh部分命令：

重启SSH

/etc/init.d/ssh restart

确认sshserver是否启动

ps -e | grep ssh C:\Users\Administrator\AppData\Roaming\Tencent\Users\228930204\QQ\WinTemp\RichOle\47I`1FVGEK73X~6GXRRHP1M.jpg

如果只有ssh-agent那ssh-server还没有启动，需要/etc/init.d/ssh start，如果看到sshd那说明ssh-server已经启动了。

## 7.Hadoop配置

切换为hadoop用户

su hadoop

解压hadoop到/usr/local/

tar zxvf hadoop-2.2.0.tar.gz /usr/local/

将hadoop 2.2.0 改名为hadoop

mv hadoop 2.2.0 hadoop

进入etc下修改配置文件

cd /usr/local/hadoop/etc/hadoop

编辑hadoop-env.sh(修改JAVA\_HOME的配置)

sudo gedit hadoop-env.sh

export JAVA\_HOME=/usr/lib/jvm/jdk1.7.0\_45

编辑yarn-env.sh，在底部增加

export JAVA\_HOME=/usr/lib/jvm/jdk1.7.0\_45

**编辑core-site.xml**

<configuration>

<property>

<name>fs.defaultFS</name>

<value>hdfs://master:9000</value>

</property>

<property>

<name>hadoop.tmp.dir</name>

<value>/usr/local/hadoop/temp</value>

</property>

</configuration>

**编辑hdfs-site.xml**

<configuration>

<property>

<name>dfs.replication</name>

<!—该数值为slave节点数 -->

<value>2</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<!-- 注意创建相关的目录结构 -->

<value>file:/usr/local/hadoop/dfs/name</value>

<final>true</final>

</property>

<property>

<name>dfs.datanode.data.dir</name>

<!-- 注意创建相关的目录结构 -->

<value>file:/usr/local/hadoop/dfs/data</value>

</property>

</configuration>

**编辑yarn-site.xml**

<configuration>

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>

<value>org.apache.hadoop.mapred.ShuffleHandler</value>

</property>

<property>

<name>yarn.resourcemanager.hostname</name>

<!-- resourcemanager hostname或ip地址-->

<value>master</value>

</property>

</configuration>

**编辑mapred-site.xml**

默认没有该文件

<configuration>

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

<final>true</final>

</property>

</configuration>

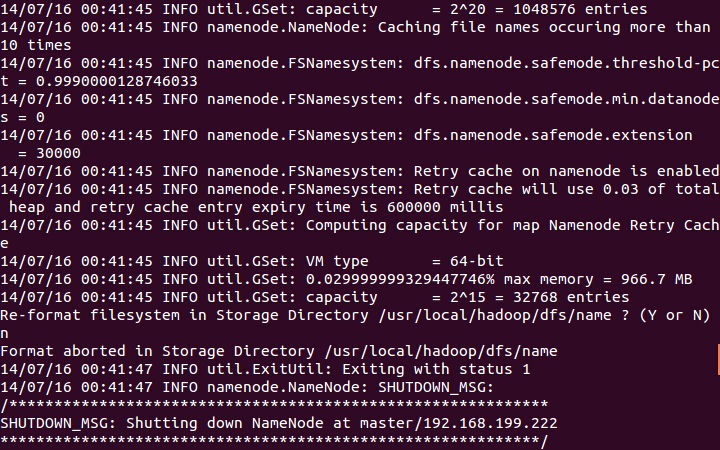
把所有Datanode节点设置可写权限（防止启动服务的时候出错）

sudo chown -R hadoop:hadoop /usr/local/hadoop/

## 启动hadoop

## 1.第一次启动需要在master 执行format

## bin/hdfs namenode -format



## 2.在master启动服务

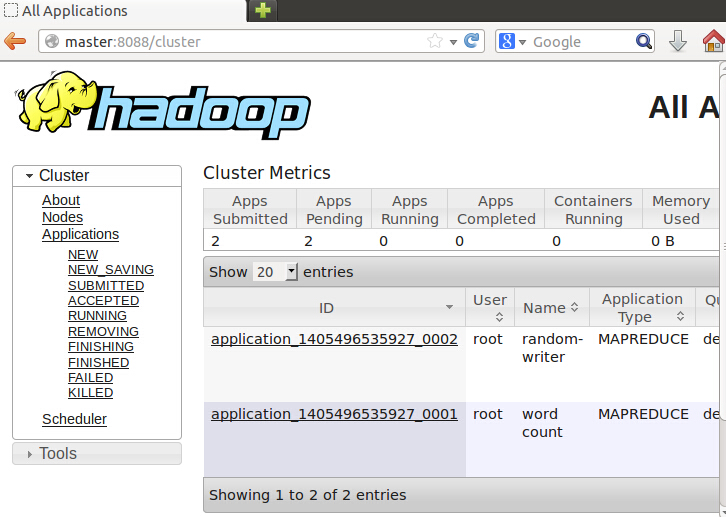
## sbin/start-all.shC:\Users\Administrator\AppData\Roaming\Tencent\Users\228930204\QQ\WinTemp\RichOle\[BDR1M8WP$0JTL[O80GT{BX.jpg

## C:\Users\Administrator\Desktop\GJ3P1HF4HM0%F}TQ82DF~WY.jpg3.在master 验证启动进程：

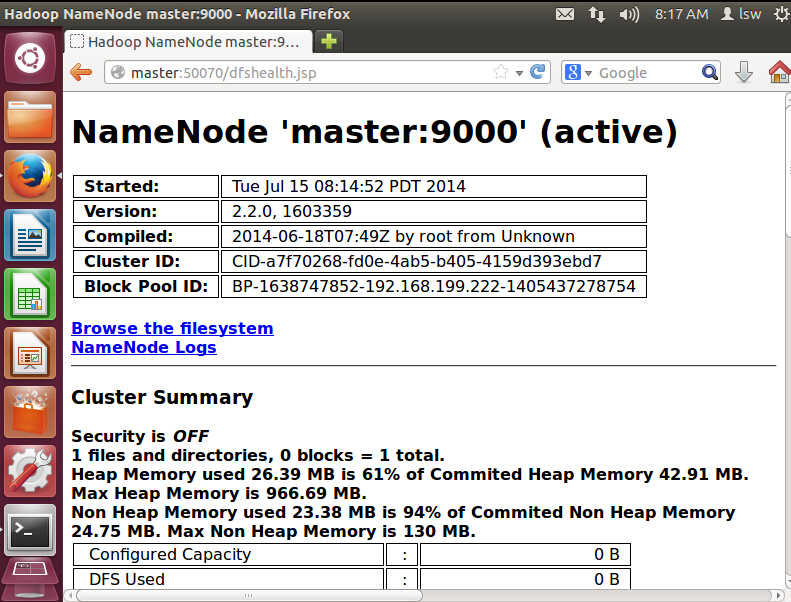
4

## 4.在slave 验证启动进程：

## 5.打开浏览器输入：http:// master:8088 可查看相关的应用运行情况。



6. 打开浏览器输入：http:// master:50070 可查看hdfs目录



## 7.测试

bin/hdfs dfs -ls /

bin/hdfs dfs -mkdir /user

bin/hdfs dfs -mkdir -p /user/micmiu/wordcount/in

创建三个文件（内容相同）

micmiu-01.txt：

micmiu-02.txt：

micmiu-03.txt：

Hi Michael welcome to Hadoop

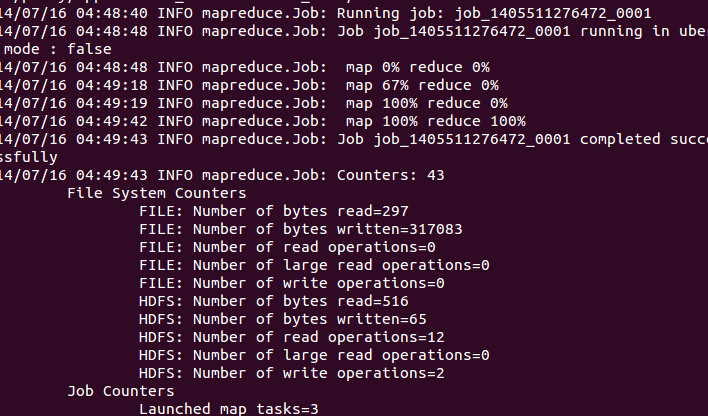
more see micmiu.com

上传三个文件

bin/hdfs dfs -put micmiu\*.txt /user/micmiu/wordcount/in

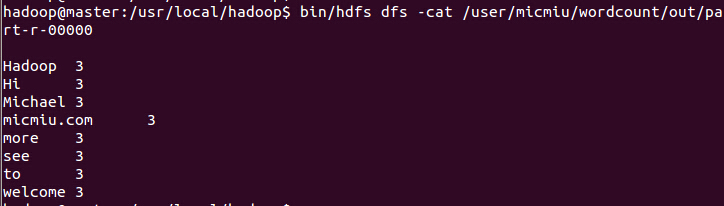
启动wordcount

bin/ hadoop jar share/hadoop/mapreduce/hadoop-mapreduce-examples-2.2.0.jar wordcount /user/micmiu/wordcount/in /user/micmiu/wordcount/out



查看结果输出out文件

bin/ hdfs dfs -cat /user/micmiu/wordcount/out/part-r-00000



## 附录：遇到的部分问题总结

1. slave节点的DataNode自动关闭问题

原因: Namenode上namespaceID与datanode上namespaceID不一致。

每次namenode format会重新创建一个namenodeId,而tmp/dfs/data下包含了上次format下的id,namenode format清空了namenode下的数据,但是没有清空datanode下的数据,所以造成namenode节点上的namespaceID与datanode节点上的namespaceID不一致。启动失败。

解决方案：

(1)停掉集群服务

　　(2)在出问题的datanode节点上删除data目录，data目录即是在hdfs-site.xml文件中配置的dfs.data.dir目录，本机器上那个是/var/lib/hadoop-0.20/cache/hdfs/dfs/data/

　　(3)格式化namenode.

(4)重新启动集群。

1. [NameNode结点不能启动](http://blog.csdn.net/xiaolang85/article/details/8517324)

jps查看结点启动情况没有NameNode结点

一种原因是在core-site.xml中指定了如下配置：

  <property>  
    <name>hadoop.tmp.dir</name>  
    <value>/tmp/hadoop/hadoop-${user.name}</value>  
  </property>

但是tmp会在启动的时候清空，而dfs的数据就会变得不一致，导致启动失败，可以重新更改位置：

  <property>  
    <name>hadoop.tmp.dir</name>  
    <value>/var/hadoop/hadoop-${user.name}</value>  
  </property>

然后执行: hadoop namenode -format  命令（由于namenode数据存储目录变更，namenode需要重新格式化）

重启后即可。

1. 运行wordcount时权限不足

解决方案：

在配置hdfs-site.xml中增加配置代码：

<property>

<name>dfs.permissions.enabled</name>

<value>false</value>

</property>

1. 解除安全模式

解决方案：

hadoop dfsadmin -safemode leave