Ubuntun 12.04.5 搭建 hadoop 单机版环境

一、使用虚拟机安装 ubuntu

如何永久的修改主机名？



二、在 ubuntu 下创建 hadoop 用户组和用户

1. 创建 hadoop 用户组

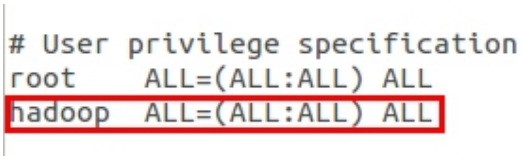
```
sudo addgroup hadoop
```

2. 创建 hadoop 用户

```
sudo adduser -ingroup hadoop hadoop
```

3. 给 hadoop 用户添加权限，打开/etc/sudoers 文件

```
sudo gedit /etc/sudoers
```



三、在 ubuntu 下安装 JDK

1. 下载 jdk-6u37-linux-i586.bin，下载后原目录为：/home/（用户名）/Downloads/
2. 在/usr/下新建 java 目录：sudo mkdir　/usr/java
3. 拷贝文件至该新建目录：
   sudo cp /home/baron/Downloads/jdk-6u37-linux-i586.bin /usr/java
4. 到/usr/java 目录下更改文件权限，使之可以运行：
   sudo chmod u+x jdk-6u37-linux-i586.bin
5. 运行该文件：sudo ./jdk-6u37-linux-i586.bin 。至此，usr/java/目录下面有一个 bin 文件包 jdk1.6.0_37，以及解压后的同名文件夹。
6. 在 profile 中配置 jdk 环境变量：sudo gedit /etc/profile，并在文档的最后加上以下几行，然后，保存：

```
export JAVA_HOME=/usr/java/jdk1.6.0_45
export PATH=$JAVA_HOME/bin:$PATH
export
CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
```

保存更改：source /etc/profile

```
注意：使用 echo $JAVA_HOME　　查看环境变量是否配置成功
      echo $PATH　　查看环境变量是否配置成功
      java -version　　查看 java 的版本
```

四、安装 ssh 服务

ssh 可以实现远程登录和管理，具体可以参考其他相关资料。

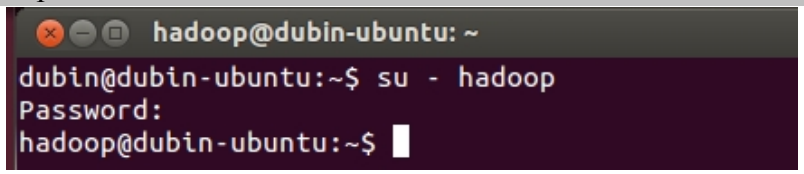
安装 openssh-server：

```
sudo apt-get install ssh openssh-server
```

五、建立 ssh 无密码登录本机

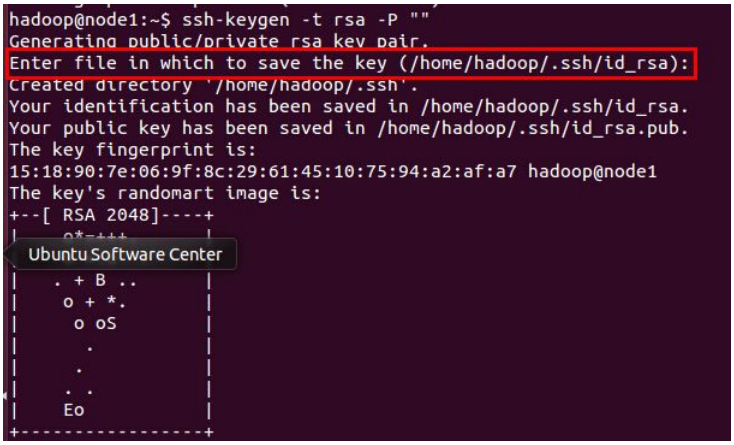
1. 若现在的用户不是 hadoop，那么执行下面的命令转换为 hadoop 用户：

```
su -hadoop
```



2. 创建 ssh-key，这里我采用的是 rsa 方式：

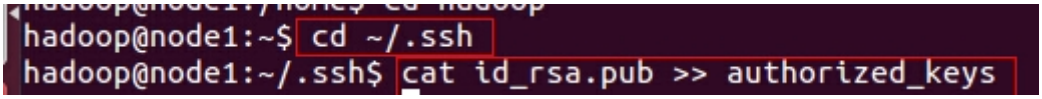
```
ssh-keygen -t rsa -P ""
```



注：在红色框的那行直接回车就行，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。

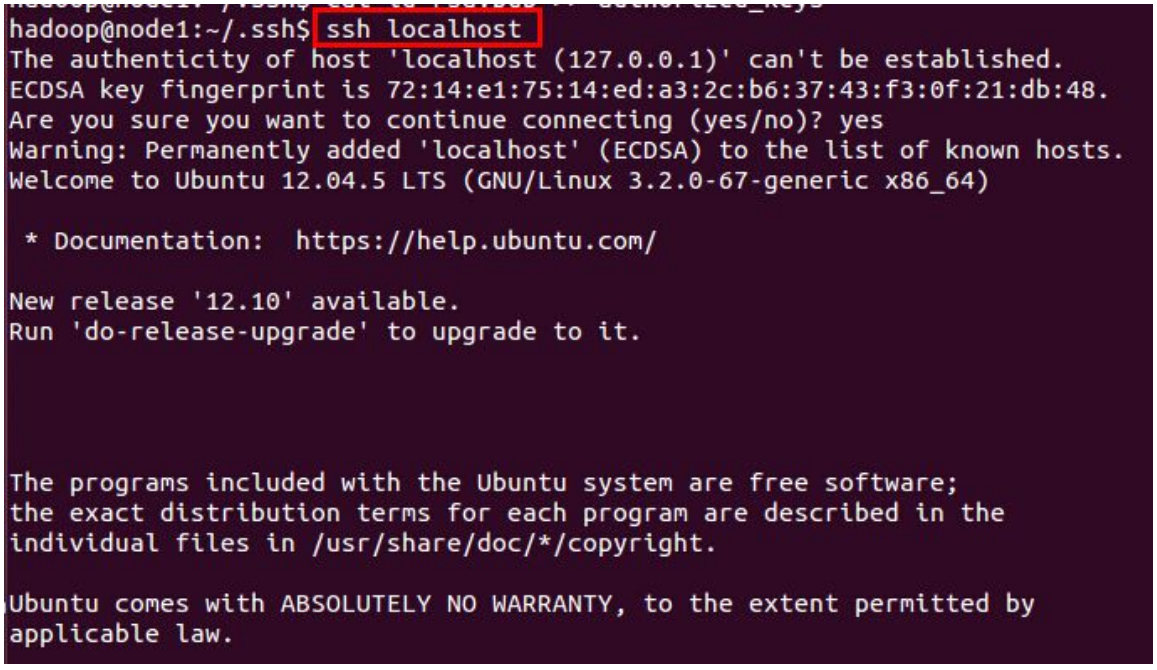
3. 进入~/.ssh/目录下，将 id_rsa.pub 追加到 authorized_keys 授权文件中，开始是没有 authorized_keys 文件的：

```
cd ~/.ssh
cat id_rsa.pub >> authorized_keys
```



（完成之后就可以无密码登录本机了）

4. 登录 localhost



（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。）

5. 执行退出命令

```
exit
```

六、安装 hadoop

1．假设现在 hadoop 在 Downloads，将它复制到安装目录/usr/local/下：

```
sudo cp /home/hadoop/Downloads/hadoop-1.0.4-bin.tar.gz /usr/local
```

2．解压 hadoop-1.0.4-bin.tar.gz：

```
hadoop@node1:/usr/local$ sudo tar -zxf hadoop-1.0.4-bin.tar.gz
```

3．将解压出的文件夹改名为 hadoop：

```
sudo mv hadoop-1.0.4 hadoop
```

4．将该 hadoop 文件夹的属主用户设为 hadoop：

```
sudo chown -R hadoop:hadoop hadoop
```

5．打开 hadoop/conf/hadoop-env.sh 文件：

```
sudo gedit hadoop/conf/hadoop-env.sh
```

6．配置 conf/hadoop-env.sh（找到#export JAVA_HOME=...,去掉#，然后加上本机 jdk 的路径）：

```
export JAVA_HOME=/usr/java/jdk1.6.0_45
```

```
# The java implementation to use.  Required.
export JAVA_HOME=/usr/java/jdk1.6.0_45
```

7．打开 conf/core-site.xml 文件：

```
sudo gedit hadoop/conf/core-site.xml
```

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:9000</value>
</property>
</configuration>
```

8．打开 conf/mapred-site.xml 文件：

```
sudo gedit hadoop/conf/mapred-site.xml
```

*mapred-site.xml

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
      <name>mapred.job.tracker</name>
      <value>localhost:9001</value>
</property>
</configuration>
```

9．打开 conf/hdfs-site.xml 文件：

```
sudo gedit hadoop/conf/hdfs-site.xml
```

*hdfs-site.xml

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

<!-- Put site-specific property overrides in this file. -->

<configuration>
<property>
<name>dfs.name.dir</name>
<value>/usr/local/hadoop/datalog1,/usr/local/hadoop/datalog2</value>
</property>
<property>
<name>dfs.data.dir</name>
<value>/usr/local/hadoop/data1,/usr/local/hadoop/data2</value>
</property>
<property>
<name>dfs.replication</name>
<value>2</value>
</property>
</configuration>
```

10．打开 conf/masters 文件，添加作为 secondarynamenode 的主机名，作为单机版环境，这里只需填写 localhost 就 Ok 了。

```
sudo gedit hadoop/conf/masters
```

11．打开 conf/slaves 文件，添加作为 slave 的主机名，一行一个。作为单机版，这里也只需填写 localhost 就 Ok 了

```
sudo gedit hadoop/conf/slaves
```

七、在单机上运行 hadoop

1. 进入 hadoop 目录下，格式化 hdfs 文件系统，初次运行 hadoop 时，一定要有该操作

```
cd /usr/local/hadoop
bin/hadoop namenode -fromat
```

2. 当你看到下图时，说明你的 hdfs 文件系统格式化成功了。

```
hadoop@node1:/usr/local/hadoop$ bin/hadoop namenode -format
14/08/09 06:10:25 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   host = node1/127.0.1.1
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 1.0.4
STARTUP_MSG:   build = https://svn.apache.org/repos/asf/hadoop/common/branches/b
ranch-1.0 -r 1393290; compiled by 'hortonfo' on Wed Oct  3 05:13:58 UTC 2012
************************************************************/
14/08/09 06:10:25 INFO util.GSet: VM type       = 64-bit
14/08/09 06:10:25 INFO util.GSet: 2% max memory = 19.33375 MB
14/08/09 06:10:25 INFO util.GSet: capacity      = 2^21 = 2097152 entries
14/08/09 06:10:25 INFO util.GSet: recommended=2097152, actual=2097152
14/08/09 06:10:25 INFO namenode.FSNamesystem: fsOwner=hadoop
14/08/09 06:10:26 INFO namenode.FSNamesystem: supergroup=supergroup
14/08/09 06:10:26 INFO namenode.FSNamesystem: isPermissionEnabled=true
14/08/09 06:10:26 INFO namenode.FSNamesystem: dfs.block.invalidate.limit=100
14/08/09 06:10:26 INFO namenode.FSNamesystem: isAccessTokenEnabled=false accessK
eyUpdateInterval=0 min(s), accessTokenLifetime=0 min(s)
14/08/09 06:10:26 INFO namenode.NameNode: Caching file names occuring more than
10 times
14/08/09 06:10:26 INFO common.Storage: Image file of size 112 saved in 0 seconds
.
14/08/09 06:10:26 INFO common.Storage: Storage directory /usr/local/hadoop/datal
og1 has been successfully formatted.
14/08/09 06:10:26 INFO common.Storage: Image file of size 112 saved in 0 seconds
.
14/08/09 06:10:26 INFO common.Storage: Storage directory /usr/local/hadoop/datal
og2 has been successfully formatted.
14/08/09 06:10:26 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at node1/127.0.1.1
************************************************************/
```

3. 启动 bin/start-all.sh:

```
bin/start-all.sh
```

```
hadoop@node1:/usr/local/hadoop$ bin/start-all.sh
starting namenode, logging to /usr/local/hadoop/libexec/../logs/hadoop-hadoop-na
menode-node1.out
localhost: starting datanode, logging to /usr/local/hadoop/libexec/../logs/hadoo
p-hadoop-datanode-node1.out
localhost: starting secondarynamenode, logging to /usr/local/hadoop/libexec/../l
ogs/hadoop-hadoop-secondarynamenode-node1.out
starting jobtracker, logging to /usr/local/hadoop/libexec/../logs/hadoop-hadoop-
jobtracker-node1.out
localhost: starting tasktracker, logging to /usr/local/hadoop/libexec/../logs/ha
doop-hadoop-tasktracker-node1.out
```

4. 检测 hadoop 是否启动成功：

```
jps
```

如果有 Namenode，SecondaryNameNode，TashTacker，DataNode，JobTracker 五个进程就说明 hadoop 单机版环境配置好了。

```
hadoop@node1:/usr/local/hadoop$ jps
8041 SecondaryNameNode
8361 TaskTracker
7801 DataNode
8410 Jps
7557 NameNode
8124 JobTracker
```

参考：http://os.51cto.com/art/201211/364167.htm
http://www.cnblogs.com/iuranus/archive/2012/11/10/2764319.html