

# 超详细解说 Hadoop 伪分布式搭建

时间 2013-09-24 19:02:04 [ITeye-博客](#) 原

文 <http://wojiaolongyinong.iteye.com/blog/1946817>

单节点伪分布式 Hadoop 配置

( 声明：文档里面需要用户输入的均已斜体表示 )

## 第一步： 安装 JDK

因为 Hadoop 运行必须安装 JDK 环境，因此在安装好 Linux 后进入系统的第一步便是安装 JDK ，安装过程和在 Windows 环境中的安装步骤很类似，首先去 Oracle 官网 去下载安装包，然后直接进行解压。我自己解压在路径 /usr/jvm 下面，假如你的安装包现在已经下载在 jvm 文件夹下面，然后 按 Ctrl+Alt +t 进去命令行，然后输入

```
cd /usr/jvm
```

进入到 jvm 文件夹下面，然后再输入如下命令进行解压：

```
sudo tar -zxvf jdk-7u40-linux-i586.tar.gz
```

## 第二步： 配置环境变量

解压结束以后，像在 Windows 系统中一样，需要配置环境变量，在 Ubuntu 中设置环境变量的过程为打开文件 /etc/profile ，因为权限的问题，因此在命令行需要输入 的是

```
sudo gedit /etc/profile
```

然后在根据提示输入用户密码即可，然后在文件最下面添加如下：

```
export JAVA_HOME=/usr/jvm/jdk1.7.0_40
```

```
export CLASSPATH=".:$JAVA_HOME/lib:  
$JAVA_HOME/jre/lib$CLASSPATH"
```

```
export PATH="$JAVA_HOME/bin:$JAVA_HOME/jre/bin:/usr/hadoop/hadoop-  
1.2.1/bin:$PATH"
```

上面这三个以单词 export 开始的三个语句就类似于我们在 Windows 中的环境变量中设置一样，而且在这个里面和 Windows 中不同的是，在 Windows 中使用 “;” 号来表示分隔，但是在 Ubuntu 中是以 “:” 号来表示分隔。还需要注意的是，上面的路径都是我自己配置的时候的路径，因为我的 JDK 解压在 /usr/jvm 中，所以我的 JAVA\_HOME 设置的是那个路径，而且如果安装的 JDK 版本不同那么后面的也不一样。同理在 CLASSPATH 路径中也是因为我自己的安装路径进行设置的，因此在配置过程中需要读者注意。在 PATH 路径中最后面还将 Hadoop 的路径也添加了进去，因此在读者安装了 Hadoop 后也将这个路径添加进环境变量 PATH 中去。

在配置完环境变量后，我们来将我们安装的 JDK 设置为 Ubuntu 系统默认的 JDK，因为之前系统里面自带 openjdk，在命令行里面输入如下：

```
sudo update-alternatives --install /usr/bin/java java /usr/jvm/jdk1.7.0_40/bin/  
java 300
```

```
sudo update-alternatives --install /usr/bin/javac javac /usr/jvm/jdk1.7.0_40/b  
in/javac 300
```

```
sudo update-alternatives --config java
```

然后我们就可以在命令行输入 java -version 来进行察看 JDK 是否已经配置好了。

### 第三步：安装 Hadoop

我们可以去 Hadoop 官网上去下载安装包，我自己下载使用的是 `hadoop-1.2.1.tar.gz`，然后当安装包下载结束后，将安装包解压到指定位置，我将安装包解压到了 `/usr/hadoop` 目录下。

具体步骤是，像解压 JDK 一样，首先加入下载的 Hadoop 安装包在 `/usr/hadoop` 文件夹下面。然后在命令行下进入 `/usr/hadoop` 文件，类似于上面的。然后输入解压命令如下

```
sudo tar -zxvf hadoop-1.2.1.tar.gz
```

后面的 Hadoop 安装包名称具体看你下载的版本，我下载的是 1.2.1 版本的。为了以后操作 `/usr/hadoop` 文件夹里面的文件方便，我们设置一下文件夹的权限，在命令行输入如下

```
sudo chown -hR long /usr/hadoop
```

注意：在上面的命令中，`long` 是我自己此时登陆的用户名，因此你需要将那个改成你自己的用户名。

### 第四步：配置 Hadoop 环境变量

在上面解压完 Hadoop 以后，现在我们来设置环境变量，其实在上面刚才我们配置 JDK 环境变量的时候，已经在 `PATH` 路径后面添加了 Hadoop 安装目录的 `bin` 目录的路径，所以那个就代表环境变量已经设置好了，但是读者一定要注意，不要 Copy，要明确自己的 Hadoop 安装路径来进行配置。

### 第五步：设置 SSH（安全外壳协议）

推荐安装 OpenSSH，Hadoop 需要通过 SSH 来启动 Slave 列表中各台主机的守护进程，因此 SSH 是必需安装的。虽然我们现在搭建的是一个伪分布式的平台，但是 Hadoop 没有区分开集群式和伪分布式，对于伪分布式，Hadoop 会采用与集群相同的处理方式，即按次序启动文件 `conf/slaves` 中记载的主机进程，只不过在伪分布式中 Slave 为 `localhost` 而已，所以对于伪分布式，SSH 是必须的。

配置过程（首先确保连接上网络）：

① 安装 SSH，在命令行输入如下

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

## ② 配置可以免密码登陆本机

在命令行输入（注意其中的 ssh 前面还有一个 “.” 不要遗漏）

```
ssh-keygen -t dsa -P "" -f ~/.ssh/id_dsa
```

（解释一下上面这条命令，ssh-keygen 代表生成密钥；-t 表示指定生成的密钥类型；dsa 是 dsa 密钥认证的意思；-P 用于提供密语（接着后面是两个单引号，不要打错）；-f 表示指定生成密钥文件）

这条命令完成后，会在当前文件夹下面的 .ssh 文件夹下创建 id\_dsa 和 id\_dsa.pub 两个文件，这是 SSH 的一对私钥和公钥，把 id\_dsa.pub（公钥）追加到授权的 key 中去，输入如下命令：

```
cat ~/.ssh/id_dsa.pub >> ~/.ssh/authorized_keys
```

至此，免密码登陆本机已经配置完毕。

说明：一般来说，安装 SSH 时会自动在当前用户下创建.ssh 这个隐藏文件夹，一般不会直接看到，除非安装好了以后，在命令行使用命令 ls 才会看到。

## ③ 输入 ssh localhost，显示登陆成功信息。

## 第六步：配置 Hadoop 伪分布式模式

现在进入到安装 Hadoop 的文件夹，找到里面的 conf 文件夹，点击进去。

- 配置 hadoop 环境文件 hadoop-env.sh

打开文件，找到某行有 “# export JAVA\_HOME = ...” 字样的地方，去掉 “#”，然后在等号后面填写你自己的 JDK 路径，比如像我自己的 JDK 路径，那就改为了 如下所示

```
“export JAVA_HOME=/usr/jvm/jdk1.7.0_40”
```

- 配置 Hadoop 的核心文件 core-site.xml

打开文件，会发现标签 <configuration></configuration> 中是空的，在空的地方添加如下配置

```
<property>
```

```
    <name>fs.default.name</name>
```

```
    <value>hdfs://localhost:9000</value>
```

```
</property>
```

```
<property>
```

```
  <name>dfs.replication</name>
```

```
  <value>1</value>
```

```
</property>
```

```
<property>
```

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

```
<value>/home/long/tmp</value>
```

```
</property>
```

（注意：在最后一个 value 值中，上面是 long，是因为那是我的用户名，所以你需要将那个修改为你自己的用户名）

- 配置 Hadoop 中 MapReduce 的配置文件 mapred-site.xml

打开文件，会发现标签 <configuration></configuration> 中是空的，在空的地方添加如下配置

```
<property>
```

```
<name>mapred.job.tracker</name>
```

```
<value>localhost:9001</value>
```

```
</property>
```

第七步：格式化 Hadoop 文件系统 HDFS 并启动 Hadoop

首次运行 hadoop 必须进行格式化 Hadoop 文件系统，以后运行即可跳过。  
打开命令行，进入安装了 Hadoop 的文件路径下，然后在命令行输入

```
bin/hadoop namenode -format
```

格式化文件系统，然后启动 Hadoop，在命令行里面输入

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

验证是否正常启动，在命令行里面输入 jps，然后回车，如果在命令行里面出现如下类似画面（因为前面的数字可以不同）

```
3235 NameNode
```

```
4113 Jps
```

```
3819 JobTracker
```

```
4059 TaskTracker
```

```
3721 SecondaryNameNode
```

```
3487 DataNode
```



则说明已经正常启动。如果以后需要关闭 Hadoop 的话，在 Hadoop 安装的文件夹路径下面在命令行输入

```
bin/stop-all.sh
```

来关闭 Hadoop 。

第八步：跑一个 Hadoop 中自带的 WordCount 程序，来体验一把

步骤如下（我在自己平台上的，读者可仿照实验）：

1) 准备一个文本文件

首先我在桌面，新建了一个空白文档 test，在里面输入一段话，或是几一些什么单词什么的，保存。

2) 将文本文件上传到 dfs 文件系统下的 input 目录下，打开命令行，进入到安装 hadoop 的文件夹下，然后输入如下

```
bin/hadoop dfs -copyFromLocal /home/long/桌面/test input
```

（注：如果 dfs 中不包含 input 目录的话就会自动创建一个）

3) 然后在命令行中输入如下命令，执行 WordCount 程序

```
bin/hadoop jar hadoop-examples-1.2.1.jar wordcount input output
```

（注：因为这个程序是 Hadoop 安装包里面自带的，就在 hadoop-examples-1.2.1.jar 中，后面的数字因为版本号的不同而不同，后面的 input 代表输入文件夹，output 代表输出文件夹，系统输出时会自动创建）

读者如果这个执行成功了，就会发现有很多输出信息，从屏幕上显示，当程序运行结束后。

#### 4) 察看结果 在命令行里面输入

```
bin/hadoop dfs -cat output/part-r-00000
```

现在你就可以看见自己刚才输入文本里面的单词计数了。

至此，伪分布式搭建结束！