

Hadoop 集群（第 5 期副刊）

——JDK 和 SSH 无密码配置

1、Linux配置java环境变量

1.1 原文出处

地址：<http://blog.csdn.net/jiedushi/article/details/6672894>

1.2 解压安装jdk

在 shell 终端下进入 jdk-6u14-linux-i586.bin 文件所在目录，
执行命令 `./jdk-6u14-linux-i586.bin` 这时会出现一段协议，连续敲回车，当询问是否同意的时候，输入 `yes`，回车。之后会在当前目录下生成一个 `jdk1.6.0_14` 目录，你可以将它复制到 任何一个目录下。

1.3 需要配置的环境变量

1) PATH 环境变量

作用是指定命令搜索路径，在 shell 下面执行命令时，它会到 PATH 变量所指定的路径中查找看是否能找到相应的命令程序。我们需要把 jdk 安装目录下的 bin 目录增加到现有的 PATH 变量中，bin 目录中包含经常要用到的可执行文件如 `javac/java/javadoc` 等，设置好 PATH 变量后，就可以在任何目录下执行 `javac/java` 等工具了。

2) CLASSPATH 环境变量

作用是指定类搜索路径，要使用已经编写好的类，前提当然是能够找到它们了，JVM 就是通过 CLASSPATH 来寻找类的。我们 需要把 jdk 安装目录下的 lib 子目录中的 `dt.jar` 和 `tools.jar` 设置到 CLASSPATH 中，当然，当前目录 “.” 也必须加入到该变量中。

3) JAVA_HOME 环境变量

它指向 jdk 的安装目录，Eclipse/NetBeans/Tomcat 等软件就是通过搜索 JAVA_HOME 变量来找到并使用安装好的 jdk。

1.4 三种配置环境变量的方法

1) 修改/etc/profile 文件

如果你的计算机仅仅作为开发使用时推荐使用这种方法，因为所有用户的 shell 都有权使用这些环境变量，可能会给系统带来安全性问题。

- 用文本编辑器打开 `/etc/profile`
- 在 `profile` 文件末尾加入：

```
export JAVA_HOME=/usr/share/jdk1.6.0_14
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
```

- 重新登录
- 注解

- a. 你要将 /usr/share/jdk1.6.0_14 改为你的 jdk 安装目录
- b. linux 下用冒号 “:” 来分隔路径
- c. \$PATH / \$CLASSPATH / \$JAVA_HOME 是用来引用原来的环境变量的值
在设置环境变量时特别要注意不能把原来的值给覆盖掉了，这是一种常见的错误。
- d. CLASSPATH 中当前目录 “.” 不能丢,把当前目录丢掉也是常见的错误。
- e. export 是把这三个变量导出为全局变量。
- f. 大小写必须严格区分。

2) 修改.bash_profile 文件

这种方法更为安全，它可以把使用这些环境变量的权限控制到用户级别，如果你需要给某个用户权限使用这些环境变量，你只需要修改其个人用户主目录下的.bash_profile 文件就可以了。

- 用文本编辑器打开用户目录下的.bash_profile 文件
- 在.bash_profile 文件末尾加入：

```
export JAVA_HOME=/usr/share/jdk1.6.0_14
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
```

- 重新登录

3) 直接在 shell 下设置变量

不赞成使用这种方法，因为换个 shell，你的设置就无效了，因此这种方法仅仅是临时使用，以后要使用的时候又要重新设置，比较麻烦。

只需在 shell 终端执行下列命令：

```
export JAVA_HOME=/usr/share/jdk1.6.0_14
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
```

1.5 测试jdk

- 1) 用文本编辑器新建一个 Test.java 文件，在其中输入以下代码并保存：

```
public class test {  
    public static void main(String args[]) {  
        System.out.println("A new jdk test !");  
    }  
}
```

2) 编译:

在 shell 终端执行命令 `javac Test.java`

3) 运行:

在 shell 终端执行命令 `java Test`

当 shell 下出现 “A new jdk test !” 字样则 jdk 运行正常。

1.6 卸载jdk

- 找到 jdk 安装目录的_uninst 子目录
- 在 shell 终端执行命令 `./uninstall.sh` 即可卸载 jdk。

2、配置OpenSSH无密码登陆

2.1 原文出处

地址: <http://www.iteye.com/topic/421608>

2.2 文章序言

最近在搭建 Hadoop 环境需要设置无密码登陆,所谓无密码登陆其实是指通过证书认证的方式登陆,使用一种被称为"公私钥"认证的方式来进行 ssh 登录。

在 linux 系统中,ssh 是远程登录的默认工具,因为该工具的协议使用了 RSA/DSA 的加密算法.该工具做 linux 系统的远程管理是非常安全的。telnet,因为其不安全性,在 linux 系统中被搁置使用了。

"公私钥"认证方式简单的解释:首先在客户端上创建一对公私钥 (公钥文件: `~/.ssh/id_rsa.pub`; 私钥文件: `~/.ssh/id_rsa`)。然后把公钥放到服务器上 (`~/.ssh/authorized_keys`),自己保留好私钥.在使用 ssh 登录时,ssh 程序会发送私钥去和服务器的公钥做匹配.如果匹配成功就可以登录了。

在 Ubuntu 和 Cygwin 配置都很顺利,而在 Centos 系统中配置时遇到了很多问题。故此文以 Centos (Centos5) 为例详细讲解如何配置证书验证登陆,具体操作步骤如下:

2.3 确认系统已经安装好OpenSSH的server 和client

安装步骤这里不再讲述，不是本文的重点。

2.4 确认本机sshd的配置文件（root）

```
$ vi /etc/ssh/sshd_config
```

找到以下内容，并去掉注释符”#“

```
RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile      .ssh/authorized_keys
```

2.5 如果修改了配置文件需要重启sshd服务（root）

```
$ vi /sbin/service sshd restart
```

2.6 ssh登陆系统 后执行测试命令

```
$ ssh localhost
```

回车会提示你输入密码，因为此时我们还没有生成证书。

2.7 生成证书公私钥的步骤

```
$ ssh-keygen -t dsa -P "" -f ~/.ssh/id_dsa
$ cat ~/.ssh/id_dsa.pub >> ~/.ssh/authorized_keys
```

2.8 测试登陆 ssh localhost

```
$ ssh localhost
```

正常情况下会登陆成功，显示一些成功登陆信息，如果失败请看下面的“一般调试步骤”。

2.9 一般调试步骤

本人在配置时就失败了，按照以上步骤依旧提示要输入密码。于是用 `ssh -v` 显示详细的登陆信息查找原因：

```
$ ssh -v localhost
```

回车显示了详细的登陆信息如下：

```
..... 省略
debug1: Authentications that can continue: publickey,gssapi-with-mic,password
debug1: Next authentication method: gssapi-with-mic
debug1: Unspecified GSS failure. Minor code may provide more information
Unknown code krb5 195
debug1: Unspecified GSS failure. Minor code may provide more information
Unknown code krb5 195
debug1: Unspecified GSS failure. Minor code may provide more information
Unknown code krb5 195
debug1: Next authentication method: publickey
debug1: Trying private key: /home/huaxia/.ssh/identity
debug1: Trying private key: /home/huaxia/.ssh/id_rsa
debug1: Offering public key: /home/huaxia/.ssh/id_dsa
debug1: Authentications that can continue: publickey,gssapi-with-mic,password
debug1: Next authentication method: password
huaxia@localhost's password:
```

同时用 root 用户登陆查看系统的日志文件：

```
$tail /var/log/secure -n 20
```

```
..... 省略
Jul 13 11:21:05 shnap sshd[3955]: Accepted password for huaxia from 192.168.8.253 port 51837
ssh2
Jul 13 11:21:05 shnap sshd[3955]: pam_unix(sshd:session): session opened for user huaxia by
(uid=0)
Jul 13 11:21:47 shnap sshd[4024]: Connection closed by 127.0.0.1
Jul 13 11:25:28 shnap sshd[4150]: Authentication refused: bad ownership or modes for file
/home/huaxia/.ssh/authorized_keys
Jul 13 11:25:28 shnap sshd[4150]: Authentication refused: bad ownership or modes for file
/home/huaxia/.ssh/authorized_keys
```

```
Jul 13 11:26:30 shnap sshd[4151]: Connection closed by 127.0.0.1
..... 省略
```

从上面的日志信息中可知文件 `/home/huaxia/.ssh/authorized_keys` 的权限有问题。
查看 `/home/huaxia/.ssh/` 下文件的详细信息如下：

```
$ ls -lh ~/.ssh/
总计 16K
-rw-rw-r-- 1 huaxia huaxia 602 07-13 11:22 authorized_keys
-rw----- 1 huaxia huaxia 672 07-13 11:22 id_dsa
-rw-r--r-- 1 huaxia huaxia 602 07-13 11:22 id_dsa.pub
-rw-r--r-- 1 huaxia huaxia 391 07-13 11:21 known_hosts
```

修改文件 `authorized_keys` 的权限（权限的设置非常重要,因为不安全的设置安全设置,会让你不能使用 **RSA 功能**）：

```
$ chmod 600 ~/.ssh/authorized_keys
```

再次测试登陆如下：

```
$ ssh localhost
Last login: Wed Jul 13 14:04:06 2011 from 192.168.8.253
```

看到这样的信息表示已经成功实现了本机的无密码登陆。

2.10 认证登陆远程服务器

备注：远程服务器 OpenSSH 的服务当然要启动。
拷贝本地生产的 key 到远程服务器端（两种方法）

1) 方法一：

```
$ cat ~/.ssh/id_rsa.pub | ssh 远程用户名@远程服务器 ip 'cat - >> ~/.ssh/authorized_keys'
```

2) 方法二：

在本机上执行：

```
$ scp ~/.ssh/id_dsa.pub michael@192.168.8.148:/home/michael/
```

登陆远程服务器 `michael@192.168.8.148` 后执行：

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

本机远程登陆 192.168.8.148 的测试：

```
$ssh michael@192.168.8.148
Linux michael-VirtualBox 2.6.35-22-generic #33-Ubuntu SMP Sun Sep 19 20:34:50 UTC 2010
i686 GNU/Linux
Ubuntu 10.10

Welcome to Ubuntu!
* Documentation: https://help.ubuntu.com/

216 packages can be updated.
71 updates are security updates.

New release 'natty' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Jul 13 14:46:37 2011 from michael-virtualbox
michael@michael-VirtualBox:~$
```

可见已经成功登陆。

如果登陆测试不成功，需要修改远程服务器 192.168.8.148 上的文件 `authorized_keys` 的权限（**权限的设置非常重要，因为不安全的设置安全设置,会让你不能使用 RSA 功能**）

```
chmod 600 ~/.ssh/authorized_keys
```