

# Hadoop 集群（第 4 期）

## ——SecureCRT 使用

### 1、SecureCRT简介

SecureCRT 是一款支持 SSH（SSH1 和 SSH2）的终端仿真程序，同时支持 Telnet 和 rlogin 协议。SecureCRT 是一款用于连接运行包括 Windows、UNIX 和 VMS 的远程系统的理想工具。通过使用内含的 VCP 命令程序可以进行加密文件的传输。有流行 CRTTelnet 客户机的所有特点，包括：自动注册、对不同主机保持不同的特性、打印功能、颜色设置、可变屏幕尺寸、用户定义的键位图和优良的 VT100, VT102, VT220 和 ANSI 竞争。能从命令行中运行或从浏览器中运行。其它特点包括文本手稿、易于使用的工具条、用户的键位图编辑器、可定制的 ANSI 颜色等。SecureCRT 的 SSH 协议支持 DES, 3DES 和 RC4 密码和密码与 RSA 鉴别。

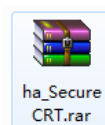


### 2、SecureCRT安装

#### 2.1 下载软件

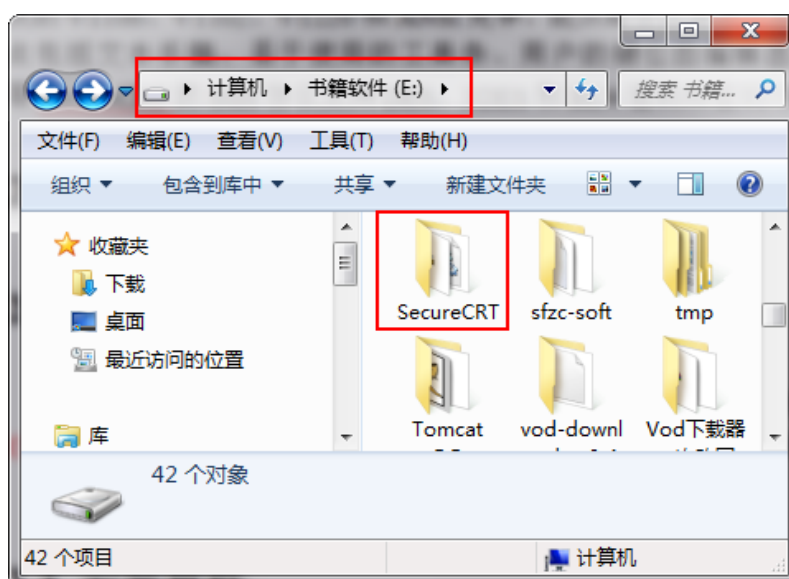
为了使实验室**后续**人员还能按照该文档设置，故把本次用的软件上传到 **CSDN** 上，即使你实验室找不到该软件的情况下还可以从下面地址中下载。

地址：<http://download.csdn.net/detail/xia520pi/4088823>

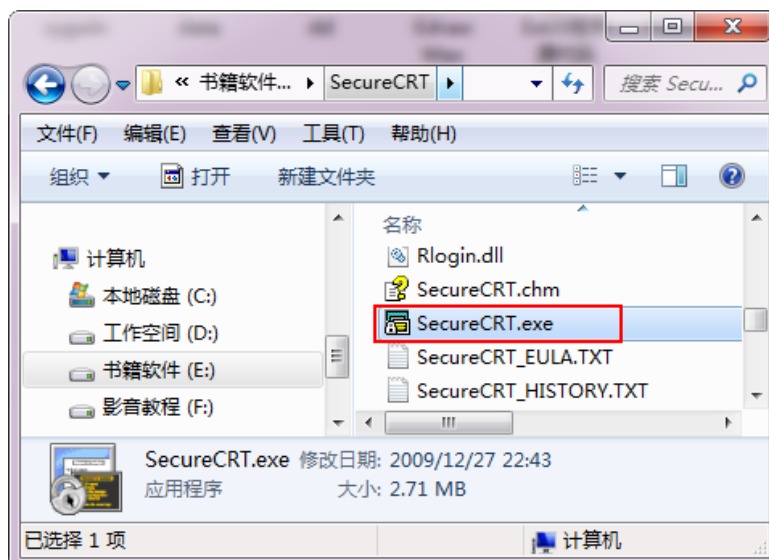


#### 2.2 安装软件

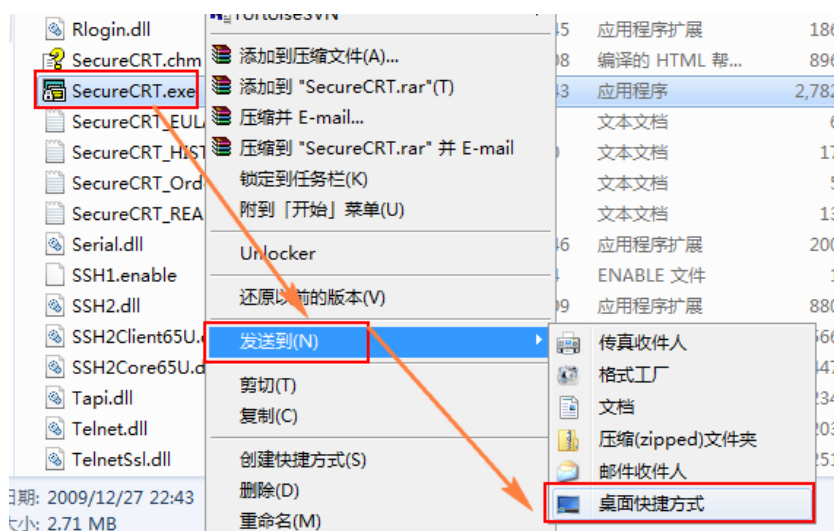
因为这包软件是**绿色安装**，所以只需要把**压缩包**解压到你想要的地方即可。例如：



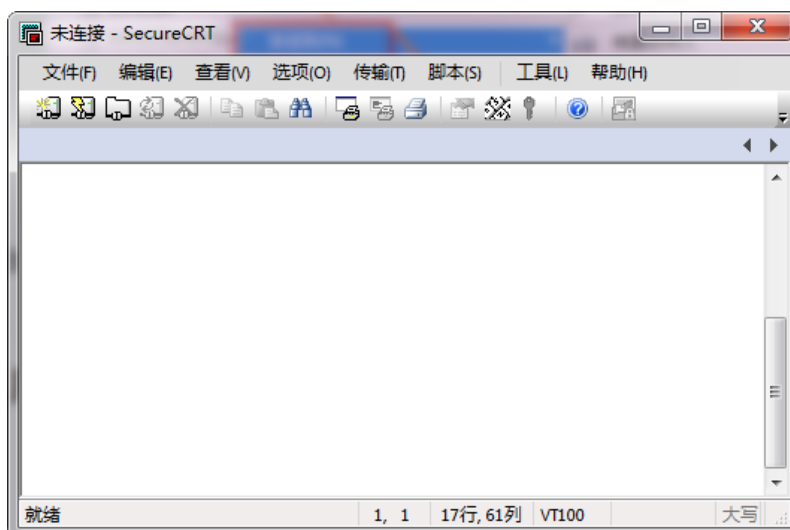
上图中是本人的解压之后的地方,我放在了 **E 盘**下,打开文件夹,找到 **SecureCRT.exe**。



接着**右击**“**SecureCRT.exe**”,建立桌面快捷方式。



当**首次**运行该软件时会弹出一个选项框,别管它,**关掉**即可,这是会出现如下界面。



上图就是这个软件的启动界面,下面我们就将如何实现该软件实现远程登录我们的

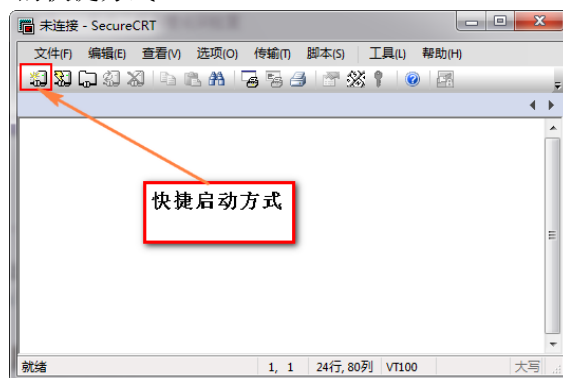
Linux 系统。

### 3、用密码登录Linux

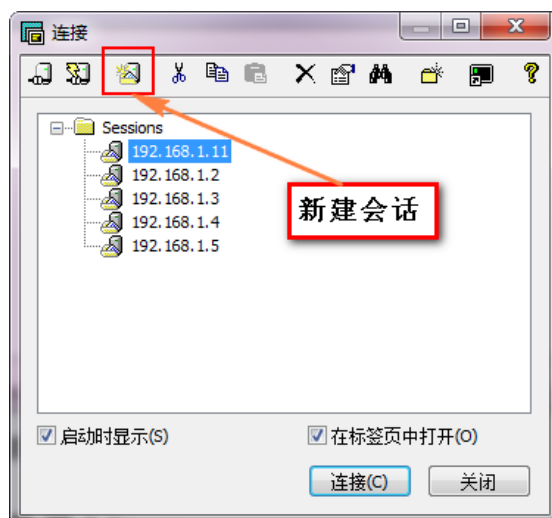
这小节我们将采用“密码”的方式登录 Linux，首先我们做的第一件事儿就是建立“远程会话”。有两种途径建立。

第一种：“文件”→“连接”

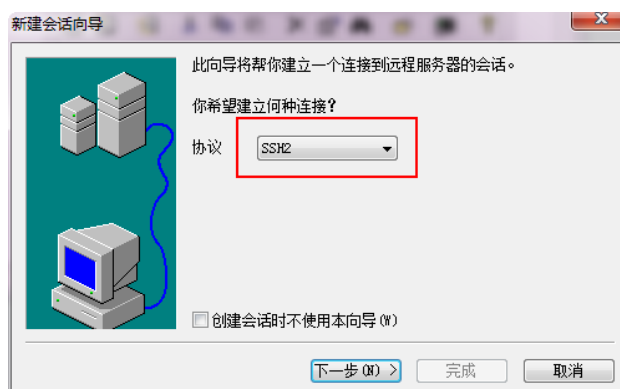
第二种：用菜单栏的快捷方式



按照两种方式都可以，然后弹出下面选项框，照着图那样点击所示按钮。



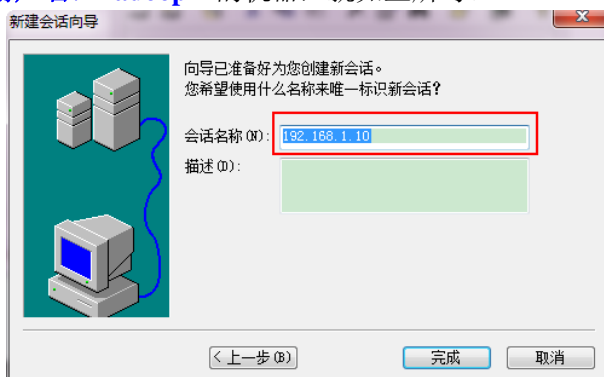
然后弹出如下界面。



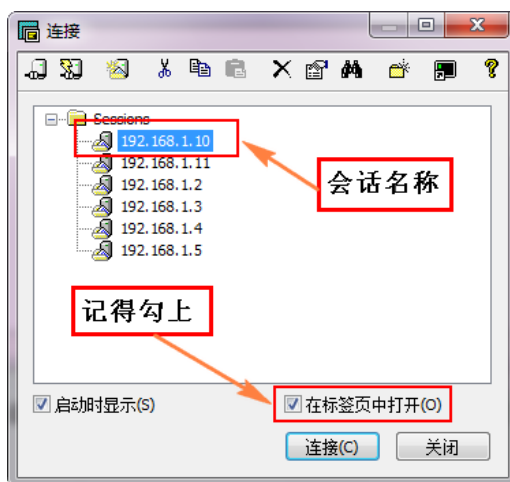
选择“SSH2”，接着点击“下一步”。



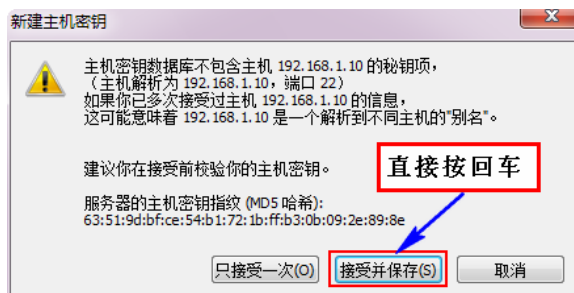
按照上图所示, 填写要远程连接的“主机名”和“用户名”, 比如我要连接“主机名: 192.168.1.10”且“用户名: hadoop”的机器, 就如上所写。



上面这个“会话名称”可以修改, 也可以保持不变, 点击“完成”, 就设置成功了。



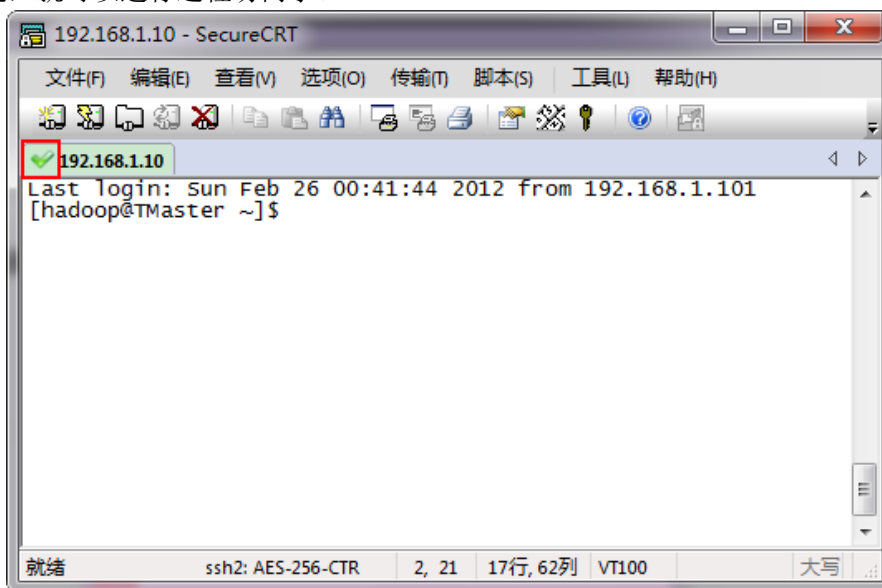
如果刚才改“会话名称”, 上图的就不是一个 IP 地址了, 而是你自己设置的名字。还有记得勾上“在标签页中打开”, 如果你想在界面中打开多个终端, 就有这个必要。这时选择你想远程连接, 选择一个“会话名称”, 比如“192.168.1.10”, 点击“连接”。



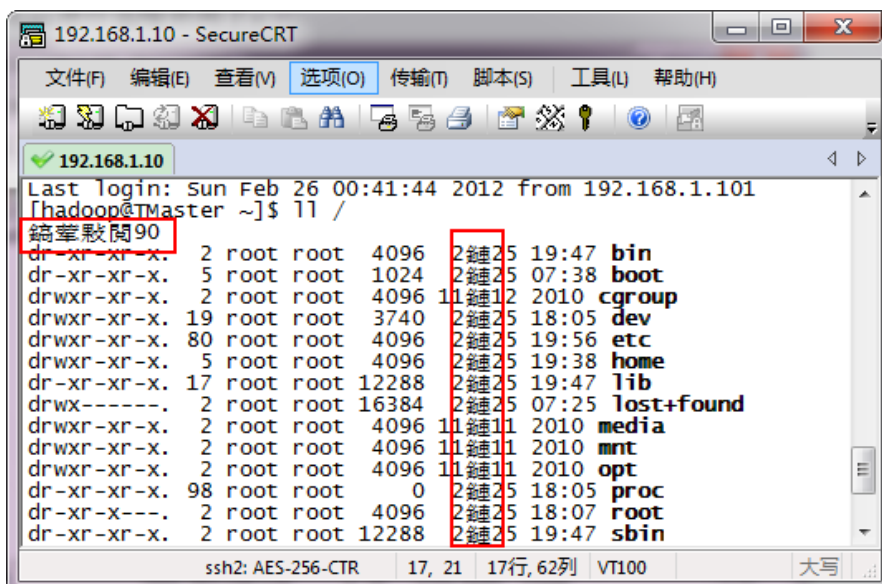
出现上面画面，默认“接受并保存”，直接按回车，就可以了。



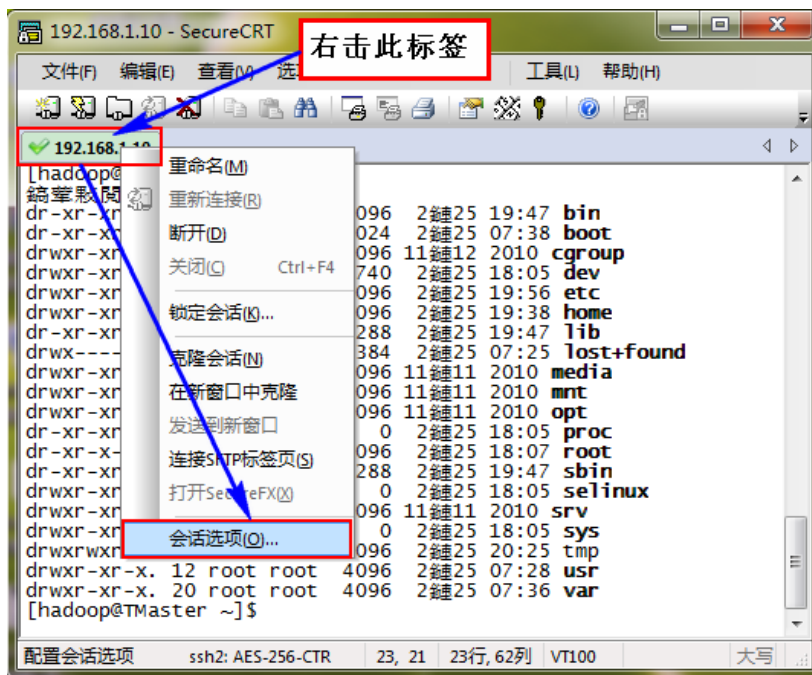
然后就会让你输入密码，写上密码，在“保存密码”前打上对勾，接着“确认”。如果网络畅通，就可以进行远程访问了。



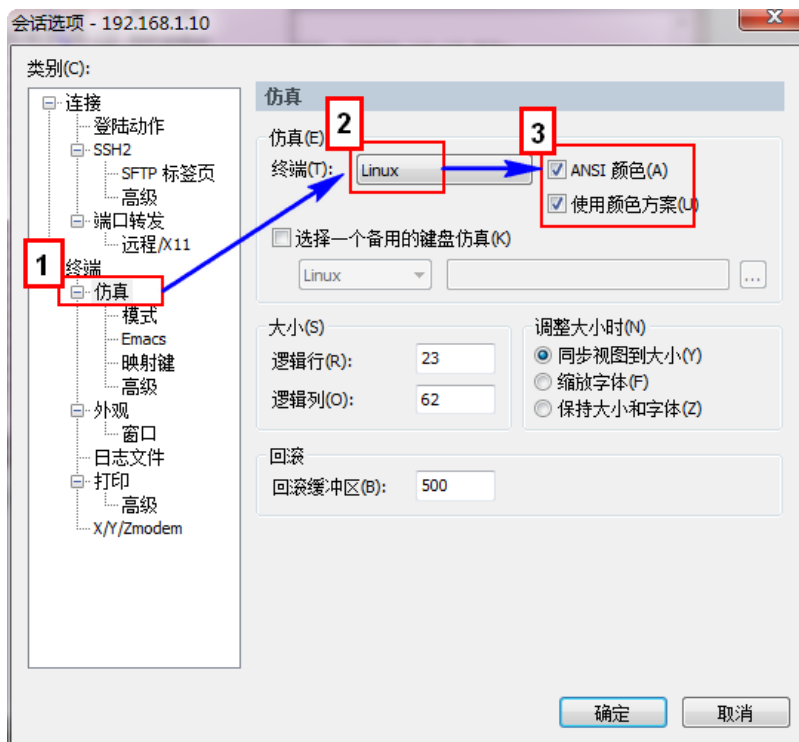
出现上面的“绿色”对勾，就表示你已经连接上了，我们现在试一个命令吧。



我们发现了两个不如意的地方，第一个：出现了红色标记的乱码；第二个：黑白显示不是很舒服。当然这个我们是可以进行调节的。那我们还等什么呢？那就赶紧行动吧，使我们的界面变得更加人性化。



按照上图中，右击“192.168.1.10”标签，会弹出一个菜单。点击“会话选项”。

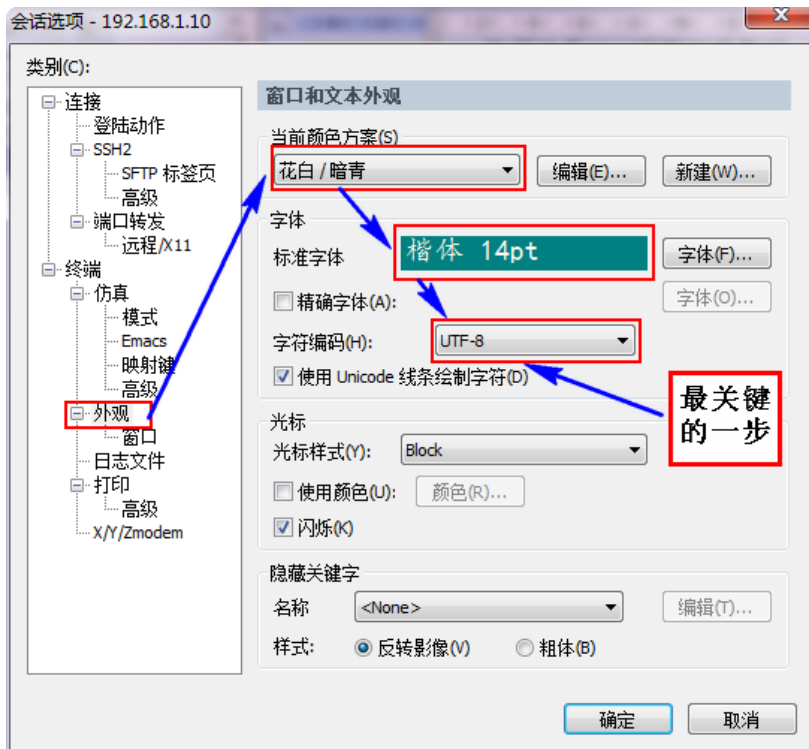


按照上图中三步进行设置，点击“仿真”，在终端选择“Linux”，接着，把“使用颜色方案”和“ANSI 颜色”打上对勾。

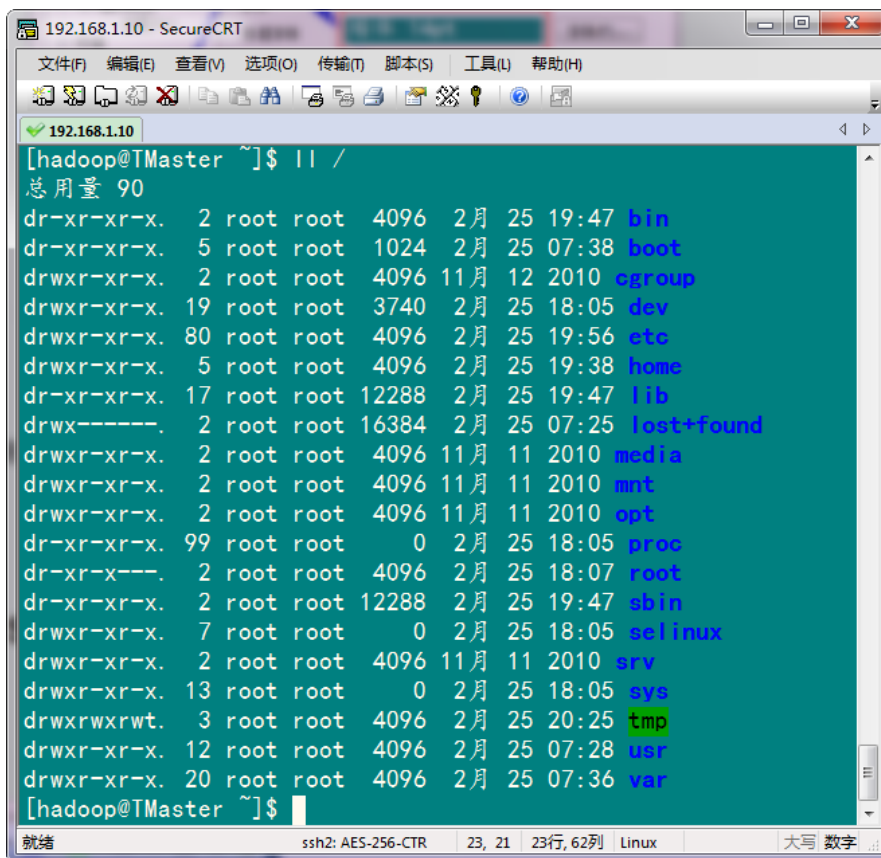
然后还有一个问题没有解决，就是“中文乱码”问题，好，我们选择“外观”，可以更改“当前颜色方案”，默认的是“Monochrome”，我比较喜欢“花白/暗青”，接着设置字体，如果你感觉默认字体比较舒服，就按原来不动，如果感觉自己比较小，可以把字体设置大，我比较喜欢“楷体 14 号”，这样看显示结果比较舒服一点，你现在该问了，我们不是解决“中文乱码”问题吗？怎么现在竟弄了一下无关紧要的事儿，哈哈，下面就该设置编码了，细心的你会发现有一个“字符编码”的选项，默认是“Default”，现在我们选择“UTF-8”。



这样设置完, 重新连接, 就换成我们想要的界面了。下面是我们刚才设置之后的界面截图。



下图是我们重新连接之后的界面, 发现界面颜色变了, 而且也**没有乱码**了。



好了, 到目前为止, 我们用**密码连接 Linux** 已经结束了, 现在就可以远程对 Linux 进行操作了。下一节我们将进行 SSH 密钥连接 Linux, 这样每次连接就不需要进行输入用户名和密码就可以直接进行连接, 而且安全性是最高的。

## 4、用SSH登录Linux

### 4.1 SSH基础知识

我想肯定有不少朋友在使用 SecureCRT 做为 SSH 的客户端软件，但都很少使用他的 RSA/DSA 加密功能吧，怎么你还不知道 RSA 是什么？

SSH，特别是 **OpenSSH**，是一个类似于 telnet 或 rsh，ssh 客户程序也可以用于登录到远程机器。我们中有许多人把优秀的 OpenSSH 用作古老的 telnet 和 rsh 命令的替代品，**OpenSSH** 不仅是**安全**的而且是**加密**的。OpenSSH 更加吸引人的特性之一是它能够使用基于一对**互补**的**数字式密钥**的 **RSA** 和 **DSA** 认证协议来认证用户。RSA 和 DSA 认证承诺不必提供密码就能够同远程系统建立连接，这是它的主要魅力之一。虽然这非常吸引人，但是 OpenSSH 的新用户们常常以一种快速却不完善的方式配置 RSA/DSA，结果虽然实现了**无密码登录**，却也在此过程中开了一个很大的安全漏洞。

#### 什么是 RSA/DSA 认证？

SSH，特别是 OpenSSH (**完全免费**的 SSH 的实现)，是一个不可思议的工具。类似于 telnet 或 rsh，ssh 客户程序也可以用于登录到远程机器。所要求的只是该远程机器正在运行 sshd，即 ssh 服务器进程。但是，与 telnet 不同的是，ssh 协议非常安全。加密数据流，确保数据流的完整性，甚至安全可靠的进行认证它都使用了专门的算法。

然而，虽然 ssh 的确很棒，但还是有一个 ssh 功能组件常常被忽略、被危险的误用或者简直就是被误解。这个组件就是 OpenSSH 的 RSA/DSA 密钥认证系统，它可以代替 OpenSSH 缺省使用的标准安全密码认证系统。

OpenSSH 的 RSA 和 DSA 认证协议的基础是一对专门生成的密钥，分别叫做**专用密钥**和**公用密钥**。使用这些基于密钥的认证系统的优势在于：在许多情况下，有可能不必手工输入密码就能建立起安全的连接。

#### 两项注意事项

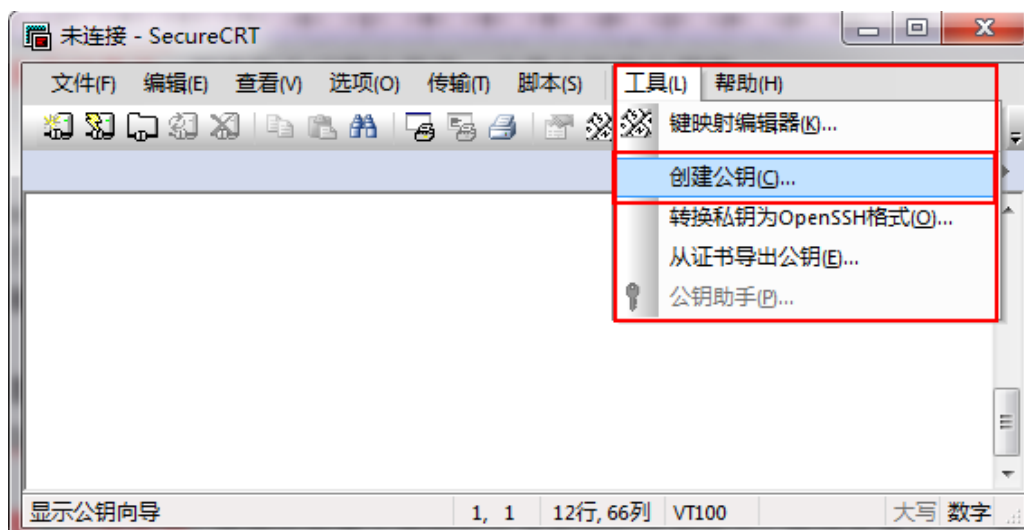
关于 RSA 和 DSA 认证有两项重要的注意事项。**第一项**是我们的确只需要生成一对密钥。然后我们可以把我们的**公用密钥**拷贝到想要**访问**的那些远程机器上，它们都会根据我们的那把**专用密钥**进行恰当的认证。换句话说，我们并不需要为想要访问的 每个系统都准备一对密钥。**只要**一对就足够了。

**另一项注意**事项是**专用密钥**不应落入其它人手中。正是**专用密钥**授权我们访问远程系统，任何拥有我们的专用密钥的人都会被授予和我们完全相同的特权。如同我们不想让陌生人有我们的住处的钥匙一样，我们应该保护我们的**专用密钥**以防未经授权的使用。在比特和字节的世界里，这意味着没有人是本来就应该能读取或是拷贝我们的专用密钥的。

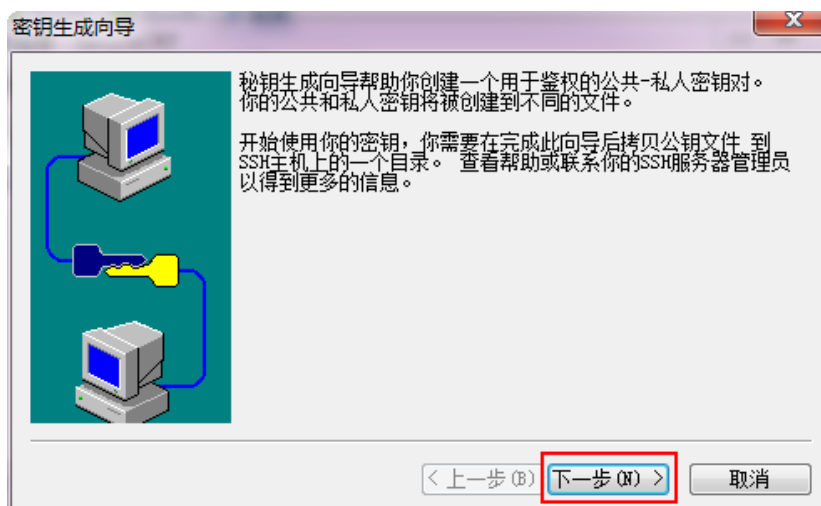
### 4.2 创建公钥

了解了 SSH 的基本知识后，我们开始进行本小节的 SSH 登录 Linux，首先在 SecureCRT 软件的菜单栏中：**工具→创建公钥**。

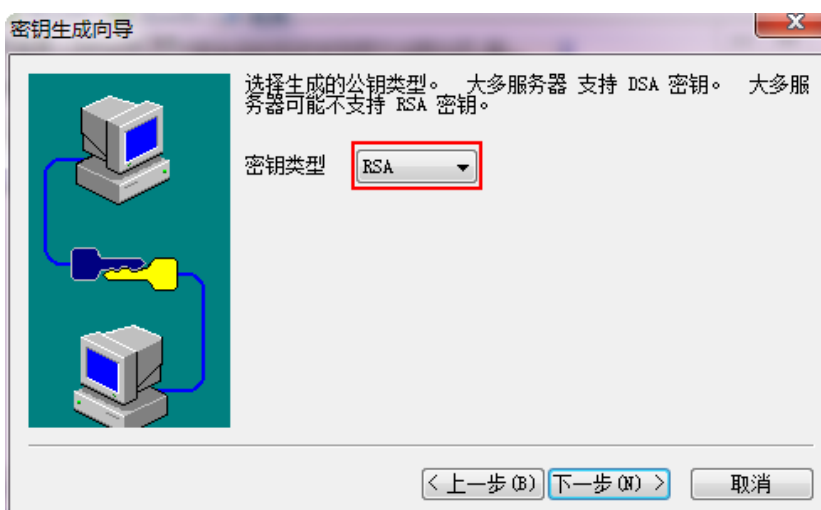




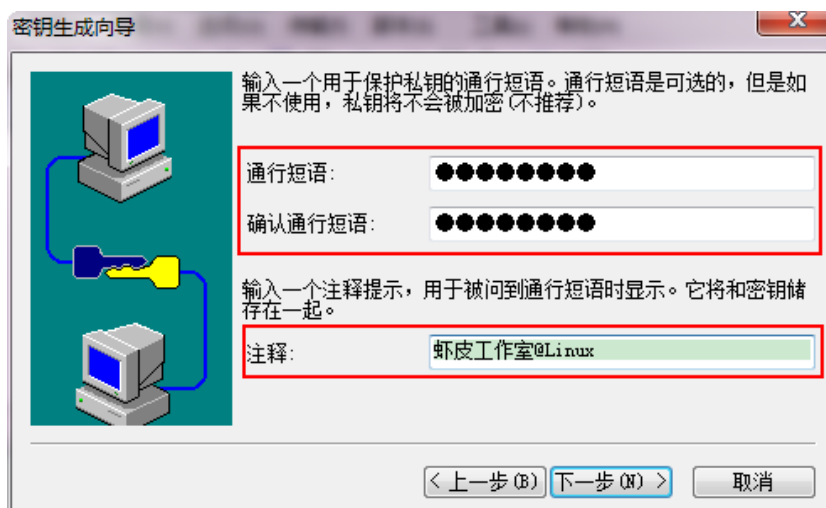
点击“创建公钥”之后出现下图，并借着点击“下一步”即可。



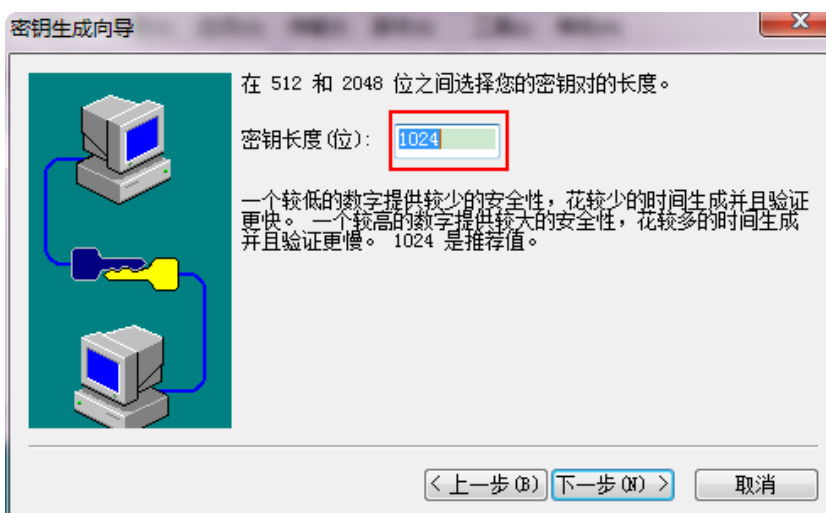
选择非对称加密方式 **RSA**，然后点击“下一步”。



通行短语是可选的，如果设置了，在连接上服务器的时候需要输入。例如：我这里设置了通行短语“**myhadoop**”，并且下面的注释写为“**虾皮工作室@Linux**”。



下一步，密钥长度，默认即可，2048 也行，越长越安全。我们在这里就保持默认即可，如果选择的越大，生成密钥生成的时间也就越长。

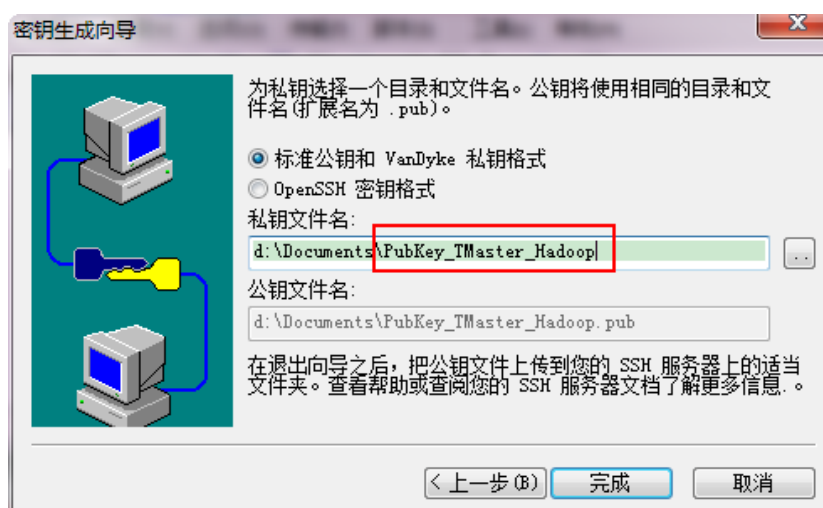


下一步，生成密钥，请耐心等待。



完成后下一步，这里保存了私钥地址和文件名。建议将这个名称改一改，因为生成多个用户的密钥的时候方便识别，而且放在 U 盘里随身携带的时候也好认识，例如把原来的

“Identity” 改写成 “PubKey\_TMaster\_Hadoop”。



到此为止，使用客户端 SecureCRT 生成密钥已经完成。接下来，就是要将密钥文件上传至服务器端，并在服务器端导入密钥。

### 4.3 SSH服务器设置

**第一步：** 在需要配置的用户根目录下创建.ssh 目录并设置一定的权限。本次在一般用户(hadoop) 下面进行。

```
[hadoop@TMaster ~]$ ll -a
总用量 32
drwx-----. 3 hadoop hadoop 4096 2月 26 06:50 .
drwxr-xr-x. 5 root root 4096 2月 25 19:38 ..
-rw-----. 1 hadoop hadoop 117 2月 26 05:45 .bash_history
-rw-r--r--. 1 hadoop hadoop 18 5月 31 2011 .bash_logout
-rw-r--r--. 1 hadoop hadoop 176 5月 31 2011 .bash_profile
-rw-r--r--. 1 hadoop hadoop 124 5月 31 2011 .bashrc
drwxr-xr-x. 2 hadoop hadoop 4096 11月 12 2010 .gnome2
-rw-----. 1 hadoop hadoop 608 2月 26 05:40 .viminfo

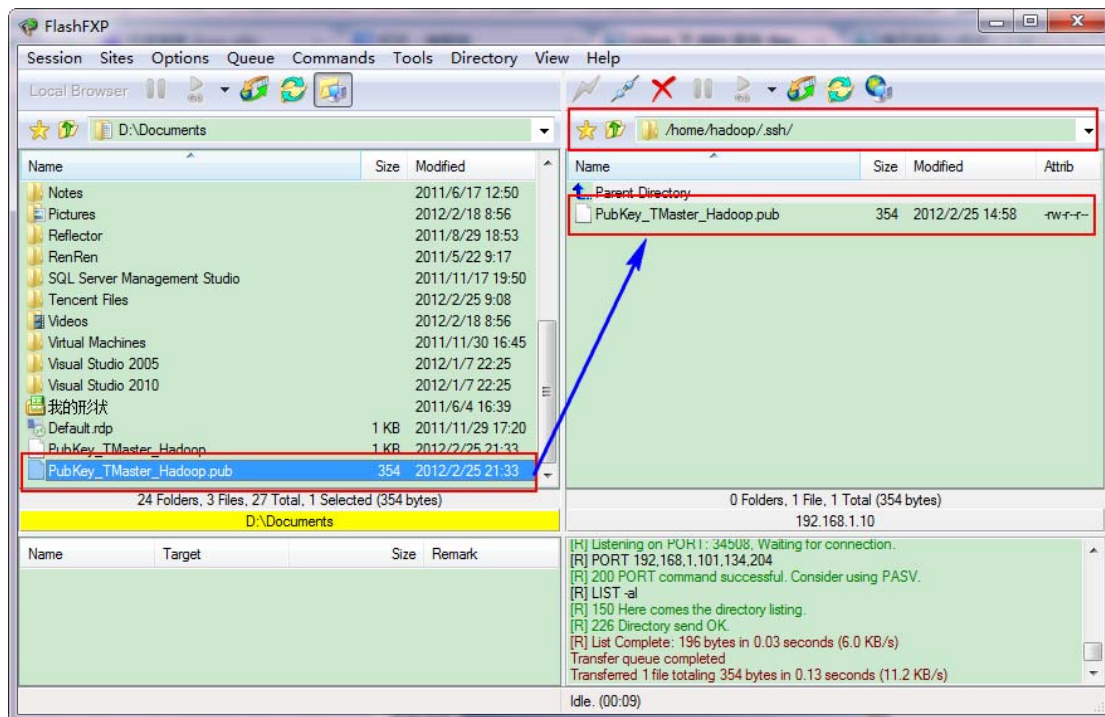
[hadoop@TMaster ~]$ pwd
/home/hadoop

[hadoop@TMaster ~]$ mkdir .ssh
[hadoop@TMaster ~]$ chmod 700 .ssh
[hadoop@TMaster ~]$ ll -a
总用量 36
drwx-----. 4 hadoop hadoop 4096 2月 26 06:51 .
drwxr-xr-x. 5 root root 4096 2月 25 19:38 ..
-rw-----. 1 hadoop hadoop 117 2月 26 05:45 .bash_history
-rw-r--r--. 1 hadoop hadoop 18 5月 31 2011 .bash_logout
-rw-r--r--. 1 hadoop hadoop 176 5月 31 2011 .bash_profile
-rw-r--r--. 1 hadoop hadoop 124 5月 31 2011 .bashrc
drwxr-xr-x. 2 hadoop hadoop 4096 11月 12 2010 .gnome2
drwx-----. 2 hadoop hadoop 4096 2月 26 06:51 .ssh
-rw-----. 1 hadoop hadoop 608 2月 26 05:40 .viminfo
[hadoop@TMaster ~]$
```

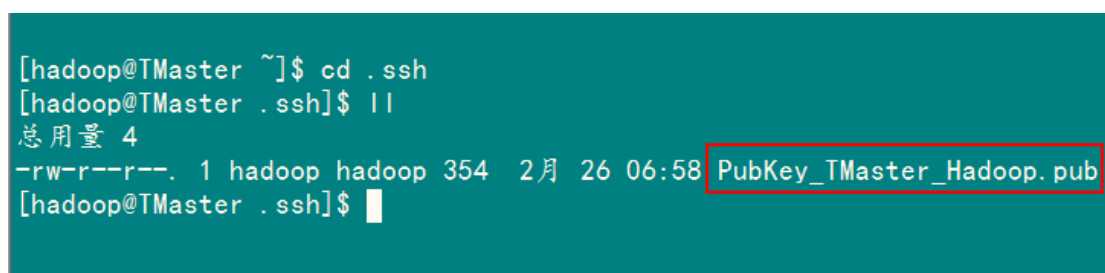
如上图所示通过下面命令建立了 .ssh 目录，并且修改了其权限。

```
# mkdir /root/.ssh
# chmod 700 /root/.ssh
```

**第二步：**采用 Hadoop 集群建立 VSFTP 服务器，通过 FTP 把我们的公钥上传上去，当然你可以通过 U 盘把公钥拷贝到服务器上。此次我们采用 FTP 的方式。

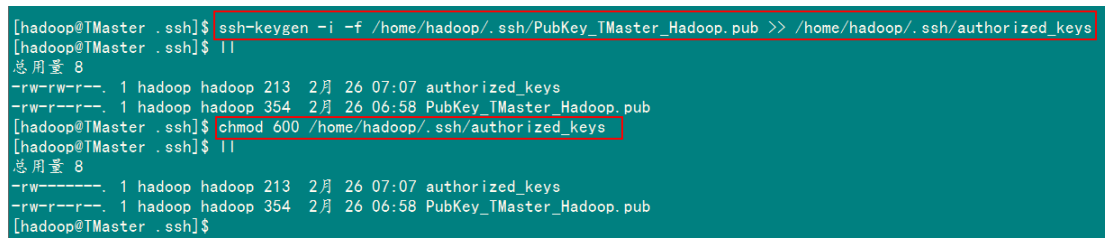


然后通过 SecureCRT 查看文件是否已经上传。



从上图我们知道已经上传了。

**第三步：**用上传的公钥生成 authorized\_keys 文件并设置文件的权限，将 SSH2 兼容格式的公钥转换成为 Openssh 兼容格式。



上图中遇到的命令如下所示：

```
# ssh-keygen -i -f /home/hadoop/.ssh/PubKey_TMaster_Hadoop.pub >>/home/hadoop/.ssh/authorized_keys
# chmod 600 /home/hadoop/.ssh/authorized_keys
```

上面命令红色标注的部分是根据实际情况变化的，比如这是从网上找的一个例子，他是这样设置的。

```
# ssh-keygen -i -f Identity.pub >> /root/.ssh/authorized_keys
# chmod 600 /root/.ssh/authorized_keys
```

上面蓝色标注的部分是两个命令的共同点。

**第四步：**设置 **ssh 配置**文件，为了安全建议只使用密钥登录，去掉默认的密码登录。

一般用户无法修改“/etc/ssh/sshd\_config”文件，所以必须切换到 root 最高权限用户下进行修改。使用下面命令。

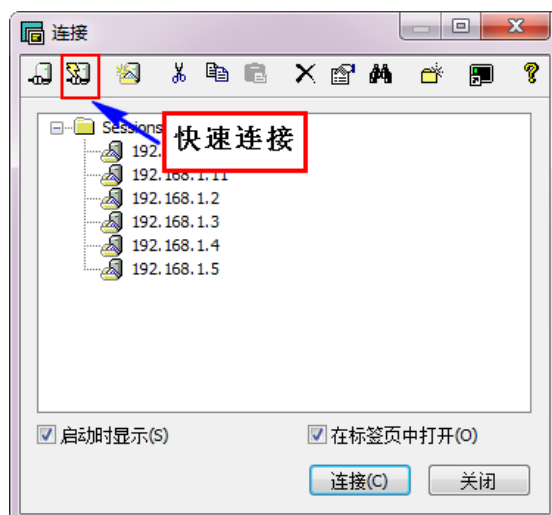
```
Protocol 2                # 使用 SSH2 协议
RSAAuthentication yes     # 启用 RSA 认证
PubkeyAuthentication yes  # 启用公钥私钥配对认证方式
AuthorizedKeysFile .ssh/authorized_keys # 公钥文件路径（和上面生成的文件同）
PasswordAuthentication no  # 禁止密码验证登陆（根据需要，不过一般启用了密钥，就不用密码了）
```

第五步：重启 Linux 服务器的 ssh 服务

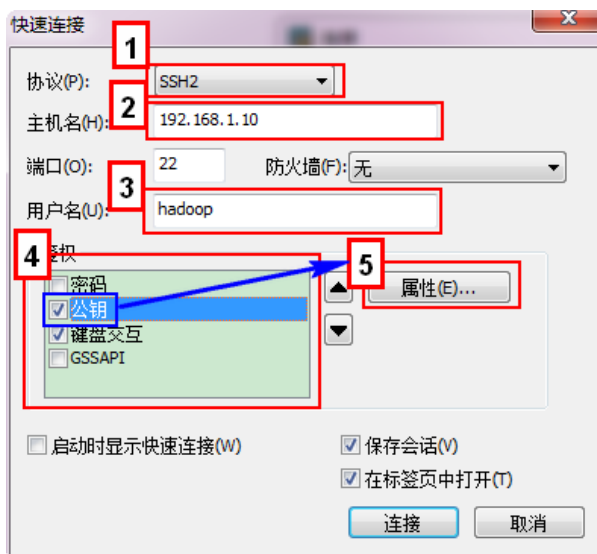
```
# service sshd restart
```

## 4.4 SSH连接服务器

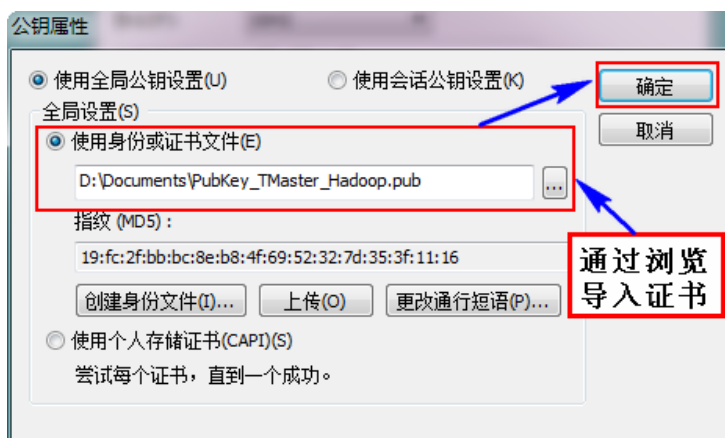
在“**用密码登陆 Linux**”小节中，我们讲到过用两种方式打开“**连接**”界面，然后选择“**快速连接**”。



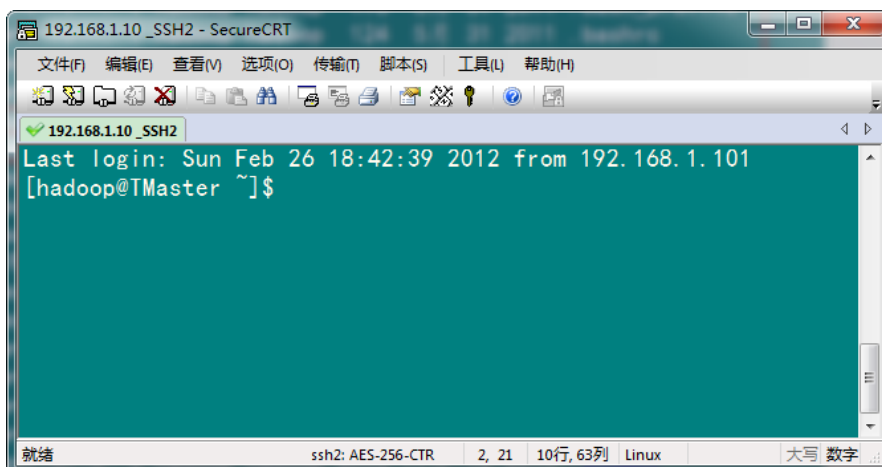
点击完“快速连接”会弹出下面界面:



按照上面的步骤进行, **步骤 1** 是默认就选择 **SSH2** 协议的, **步骤 2** 和 **步骤 3** 是根据实际情况填写的。比如我上面要连接远程 Linux 的 IP 为“192.168.1.10”且一般账户为“hadoop”的机器。**步骤 4** 只需要勾上“**公钥**”和“**键盘交互**”, 然后选中“**公钥**”, 点击其“**属性**”按钮, 及图中标注的**步骤 5**。



通过浏览选择我们刚才生成的证书, 点击确定, 回到“**快速连接**”界面点击“**连接**”按钮, 然后弹出一个窗口, 然填写我们的“**通行短语**”。把我们刚才设置的通行短语“**myhadoop**”输入进去, 就可以了。点击确定, 就可以实现 SSH 无密码连接了。





## 参考文献

---

感谢以下文章的编作者，没有你们的铺路，我或许会走得很艰难，参考不分先后，贡献同等珍贵。

【1】Linux 下 SSH 配合 SecureCRT 的密钥完美使用方法

地址: <http://linkboy.blog.51cto.com/821152/297512>

【2】SecureCRT 使用证书管理 linux

地址: <http://wenku.baidu.com/view/e116ba33a32d7375a4178039.html>

【3】SecureCRT 密钥远程登录 Linux

地址: <http://edges.blog.51cto.com/705035/581346>

【4】Linux 新建 SSH 登陆用户以及 Putty 和 SecureCRT 设置

地址: <http://hi.baidu.com/luosiyong/blog/item/e5e1980fb7ddfef3ab6457f7.html>