

LINUX 服务器环境搭建

李君

(QQ: 176125727)

Linux 是一个性能优异、安全的操作系统，许多大型的 Web 应用都是基于 Linux 作为服务器，所以系统的学习一下 Linux 操作系统是非常有必要的。本文不以 Linux 基础为例，主要以 Linux 服务器端开发环境搭建为主，适合有一定 Linux 基础的朋友。

本文采用的 Linux 环境为 RedHat Enterprise Linux Server4。本文主要讲解以下内容：

- ✚ Java 开发环境的安装和配置（JDK 安装、环境变量的配置）
- ✚ Tomcat 服务器安装和配置（Tomcat 安装、环境变量配置）
- ✚ 基于源码的 MySQL 数据库安装（此处之所以采用源码编译安装，为了后面的 PHP 环境做准备）
- ✚ Oracle 数据库的安装（包括服务器端和客户端的安装）
- ✚ Apache 服务器的安装和配置（主要用来运行 PHP 程序）
- ✚ PHP 环境安装和配置
- ✚ Flex 开发环境搭建

从上面可以看到，我们这篇教程要讲解的内容很多，所以我们需要准备如下的必备环境：

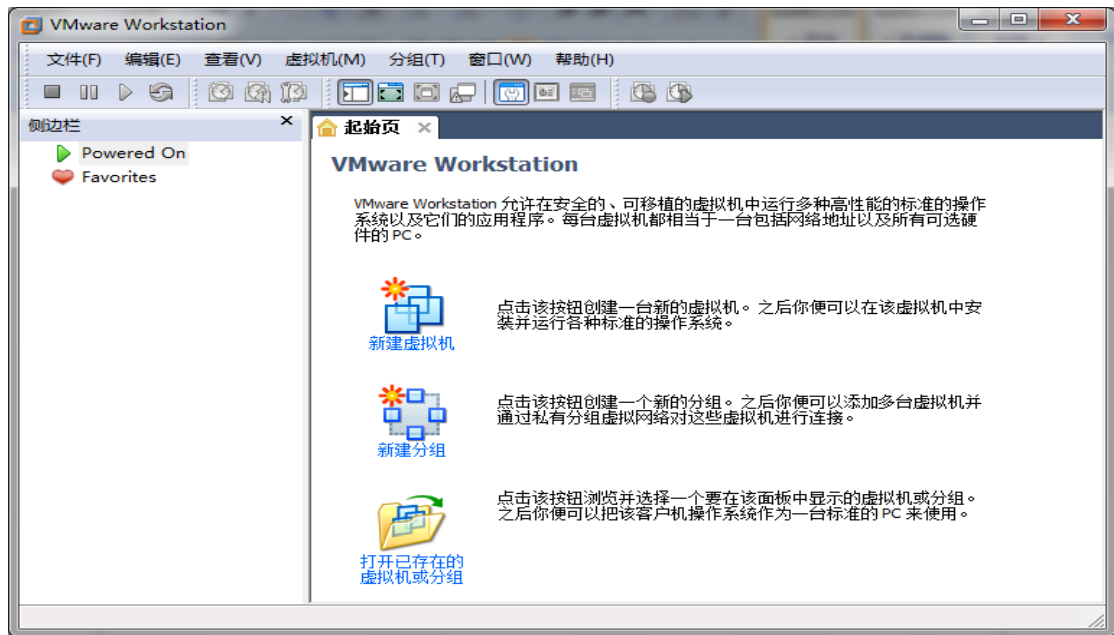
必备工具：

- ✚ VMware-workstation-6.5 或更高版本
- ✚ RedHat Enterprise Linux 4 的 iso 文件，共 4 个
- ✚ SSHSecureShellClient，用于远程连接 Linux
- ✚ jdk-6u18-linux-i586.bin，可以从 sun 网站下载
- ✚ mysql-5.1.41.tar.gz，MySQL 的 Linux 源码分发
- ✚ SQLyog Enterprise，MySQL 的可视化操作工具
- ✚ 10201_database_linux32.zip，Oracle 的 Linux 版本
- ✚ 10201_client.zip，Oracle 在 Win 下的客户端，注意根据操作系统选择
- ✚ PL/SQL Developer，操作 Oracle 的可视化工具
- ✚ PHP 相关组件
 - php-5.2.14.tar.gz
 - freetype-2.1.10.tar.gz
 - gd-2.0.35.tar.gz
 - jpegsrc.v6b.tar.gz
 - libpng-1.2.29.tar.gz
 - libxml2-2.6.2.tar.gz
 - zlib-1.2.3.tar.gz
 - curl-7.15.0.tar.gz
- ✚ httpd-2.2.16.tar.gz，Apache 服务器
- ✚ flex_sdk_3.5.0.12683.zip，Flex SDK

LINUX 环境搭建

由于本人采用的是 Windows7 环境，所以不能直接安装 Linux 操作系统，我将用 VMware 虚拟机来承载 Linux 操作系统，而 Windows 作为宿主机。下面我开始讲解具体步骤。

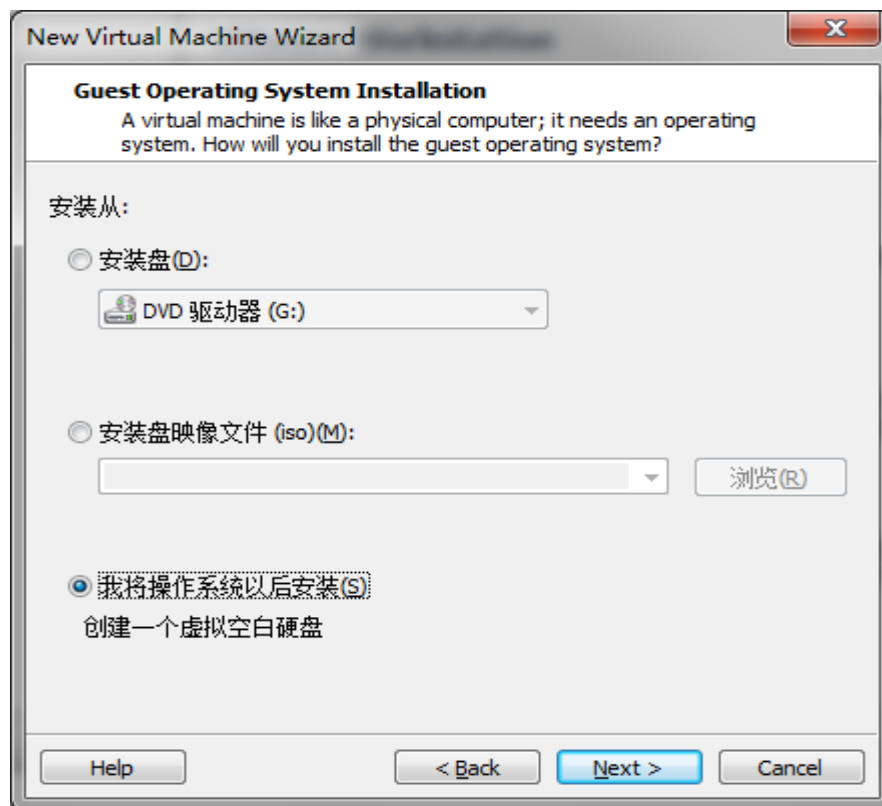
启动 VMWare，出现如下界面：



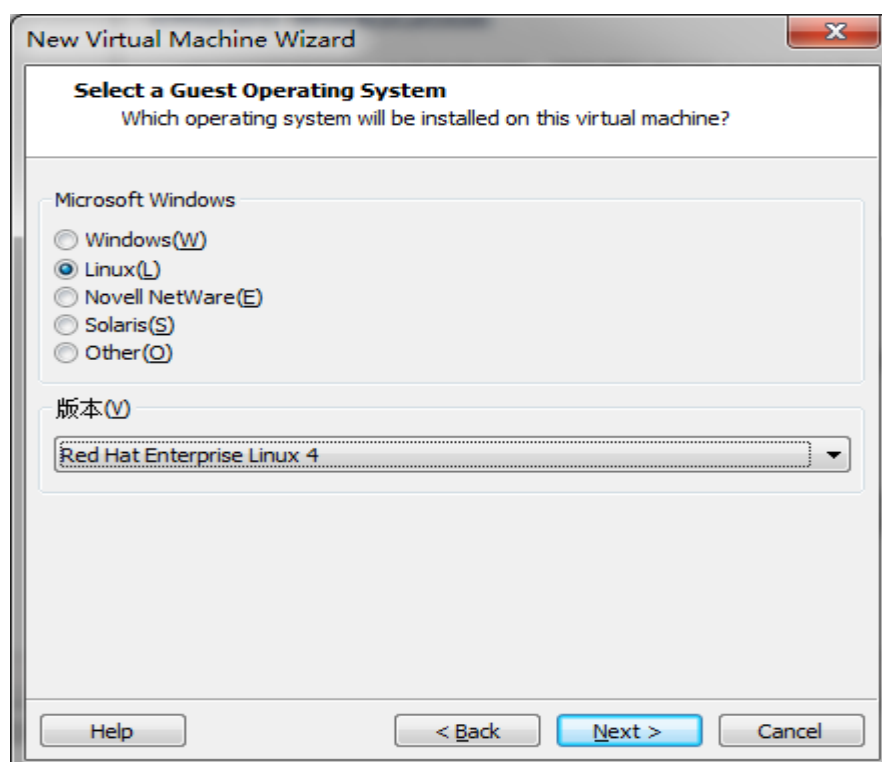
点击新建虚拟机按钮，出现如下界面：



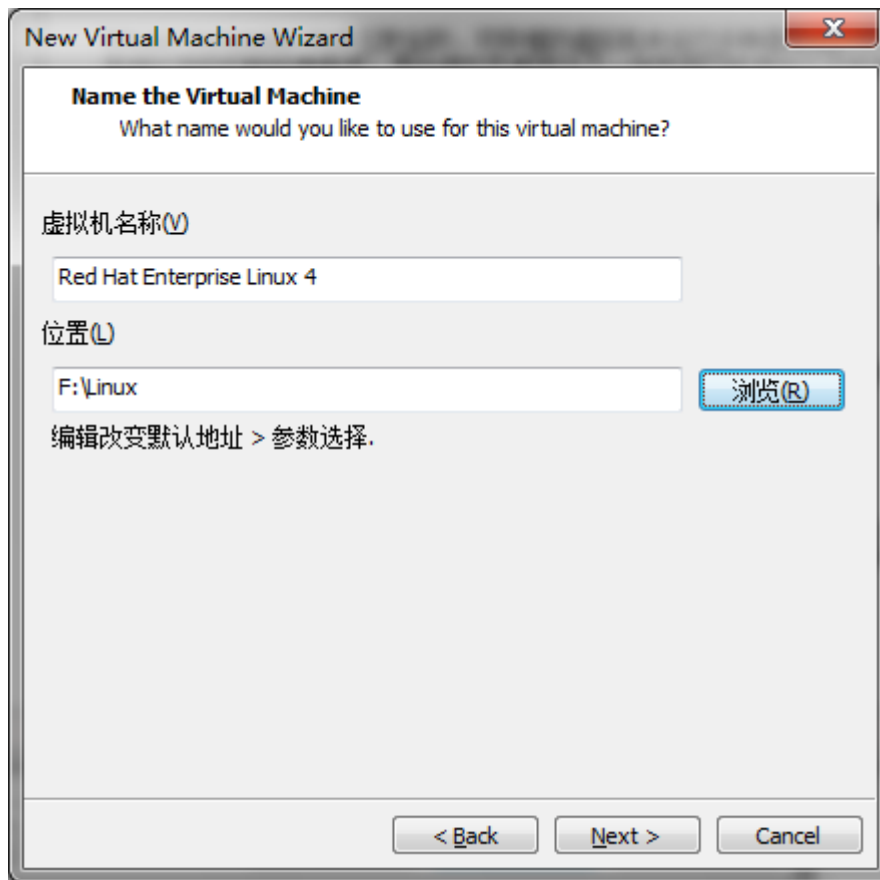
根据情况自行选择安装的类型，此处我采用默认安装类型。点击 **Next** 按钮出现如下界面：



选择“我将操作系统以后安装”，点击 **Next** 按钮出现如下界面：



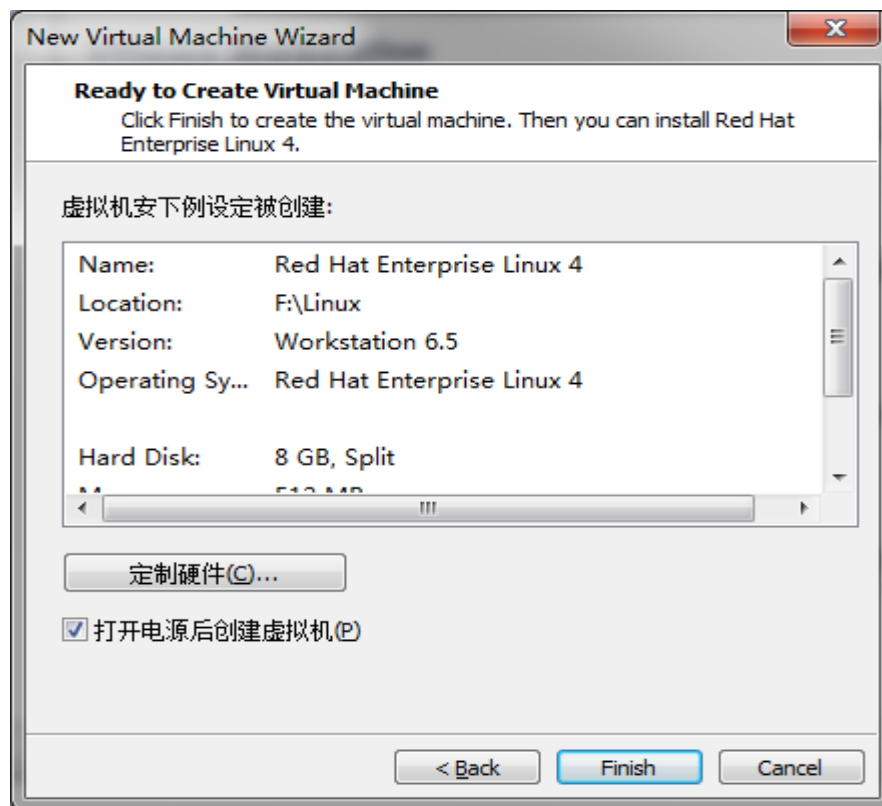
由于我们安装的是 Linux，根据需要选择 Linux 的版本，点击 Next 按钮，出现如下界面：



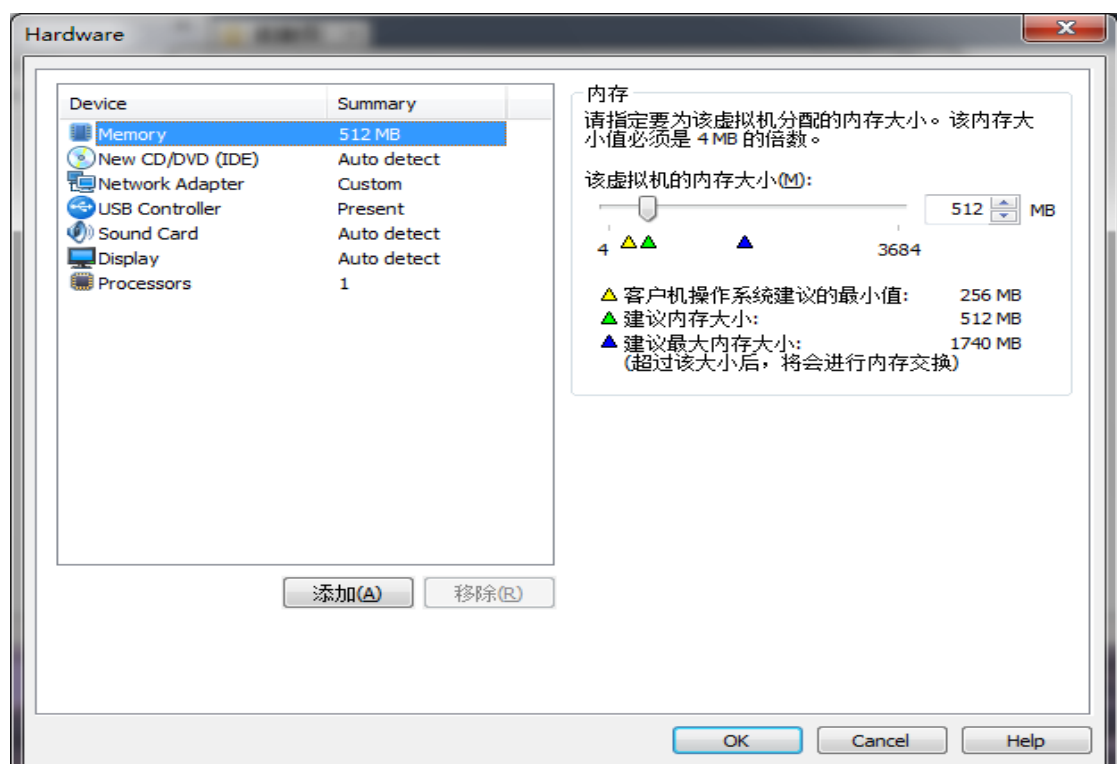
指定虚拟机名称和安装的位置，点击 Next 按钮出现如下界面：



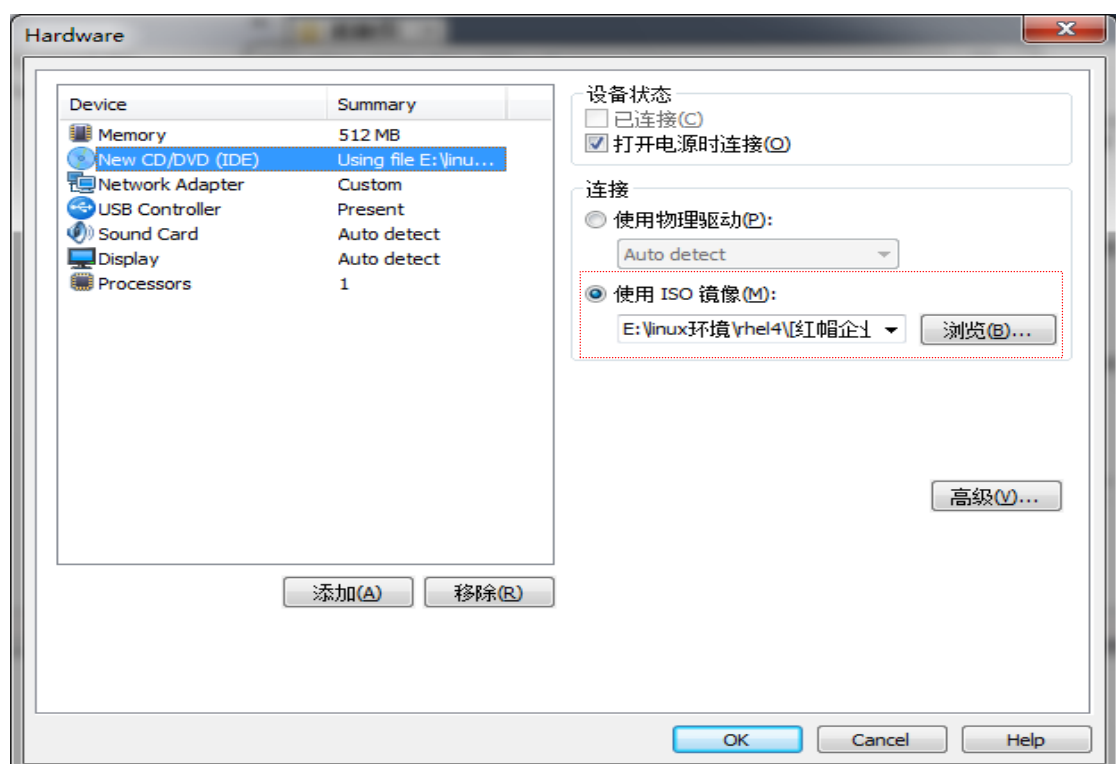
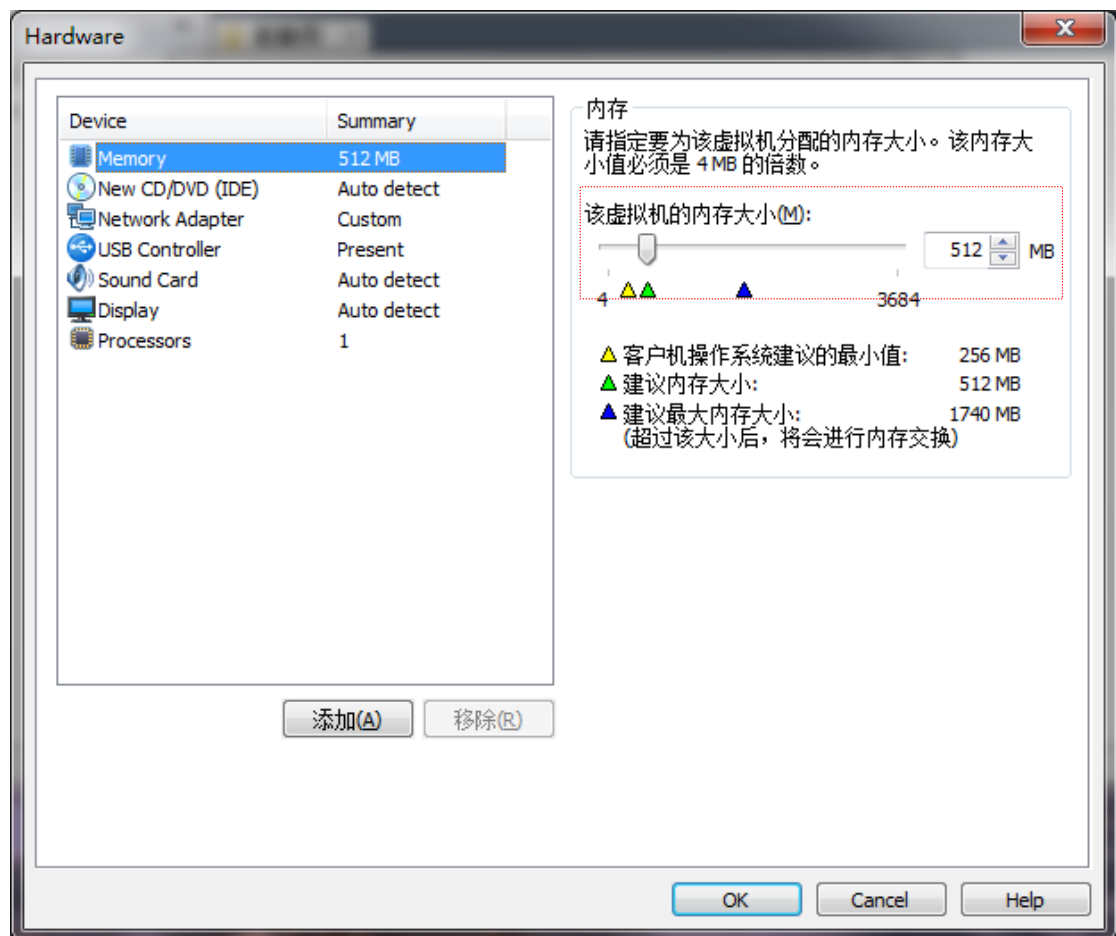
最大磁盘大小根据情况指定，选择“以每个文件为 2GB 存储虚拟磁盘”，点击 Next 出现如下界面：

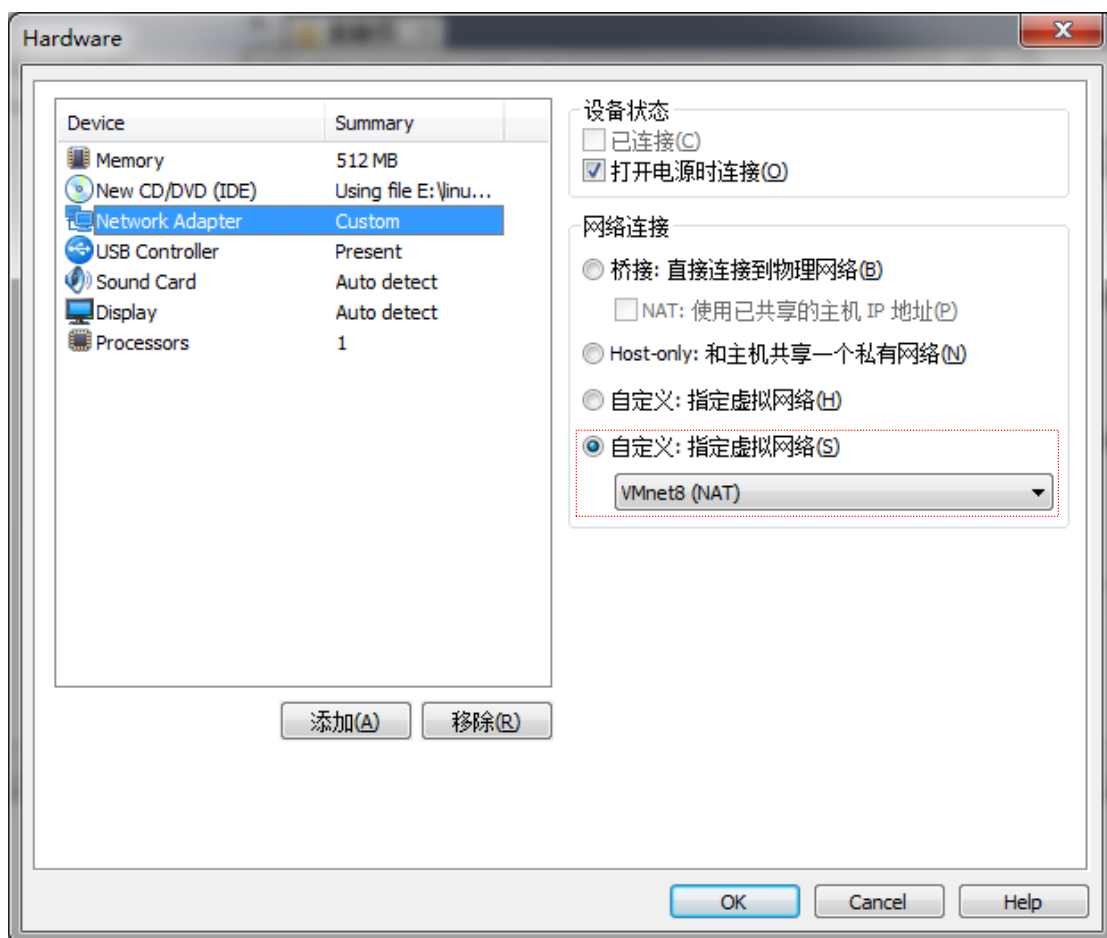


点击定制硬件按钮出现如下界面：

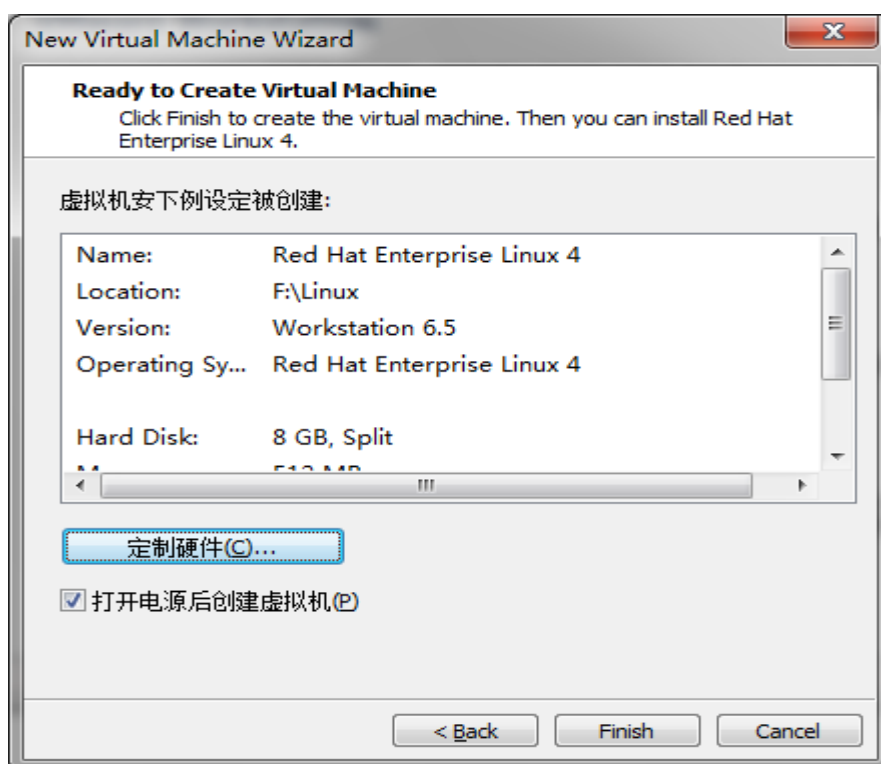


根据需要定制硬件，注意如下几处红色的选择：

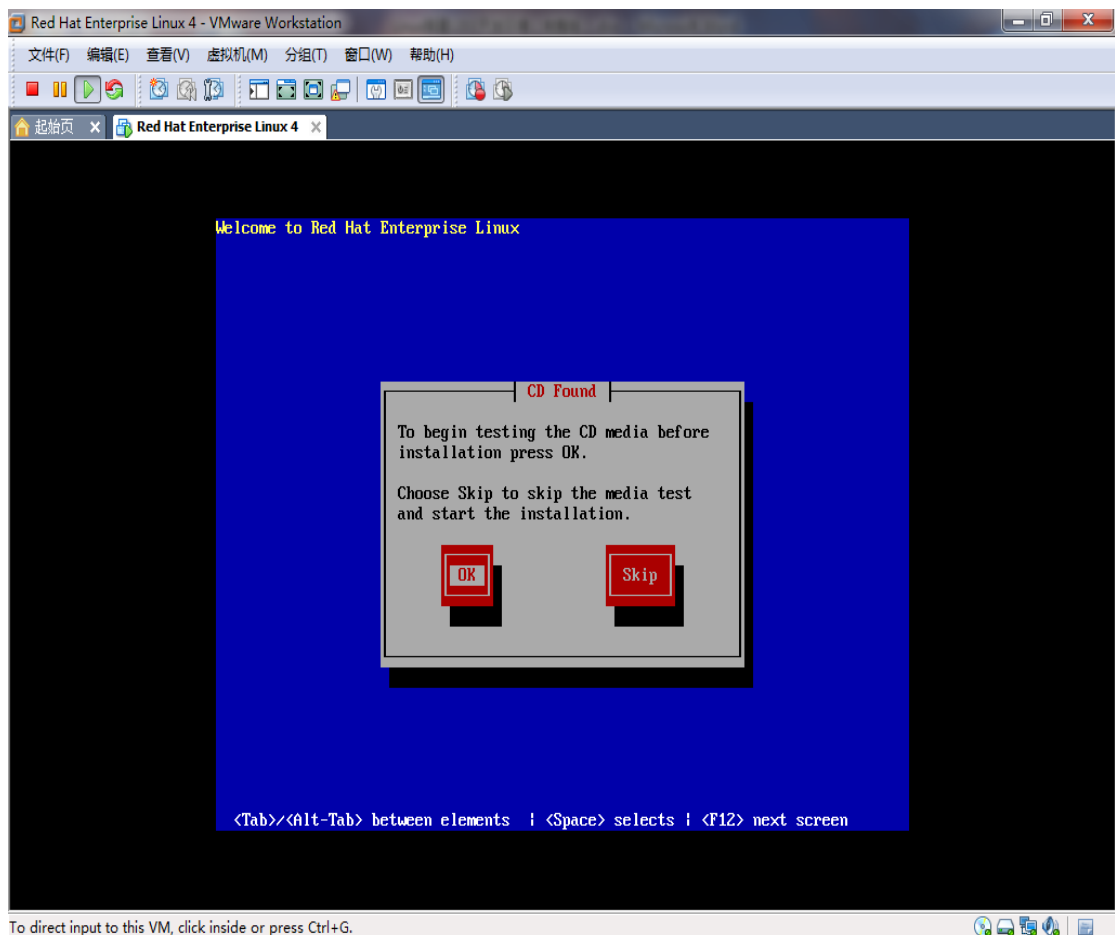
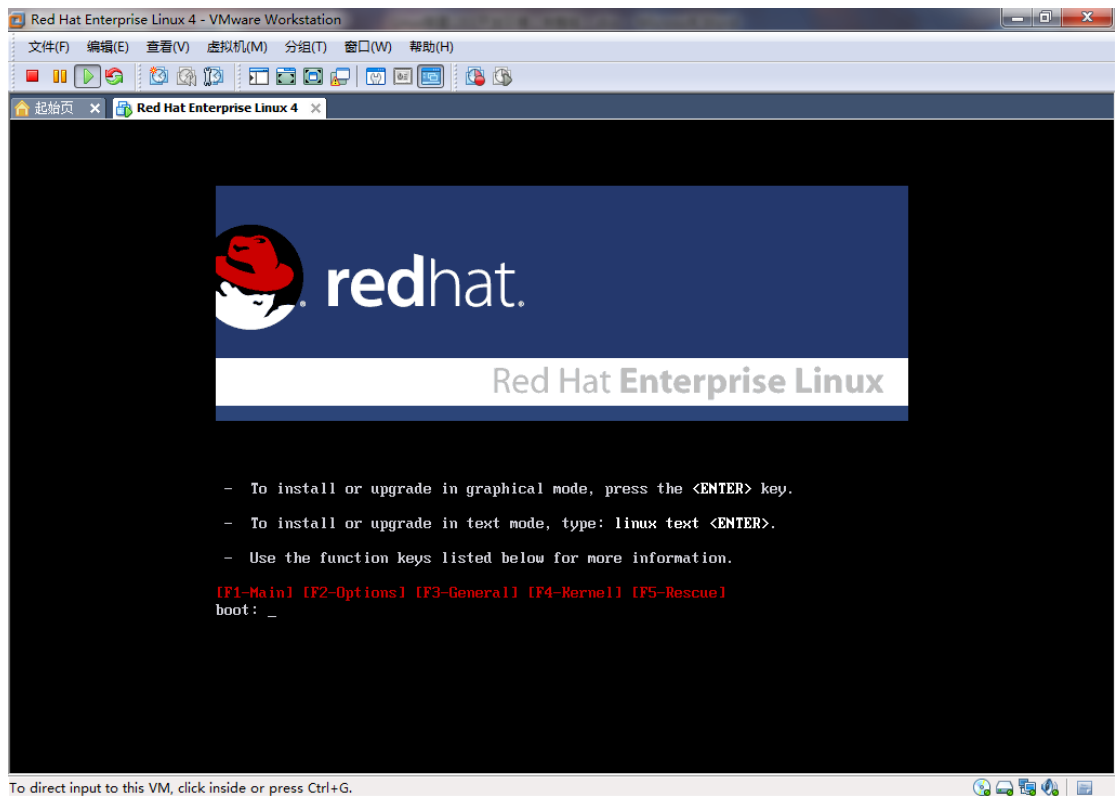




配置好这几项后点击“OK”按钮出现如下界面：



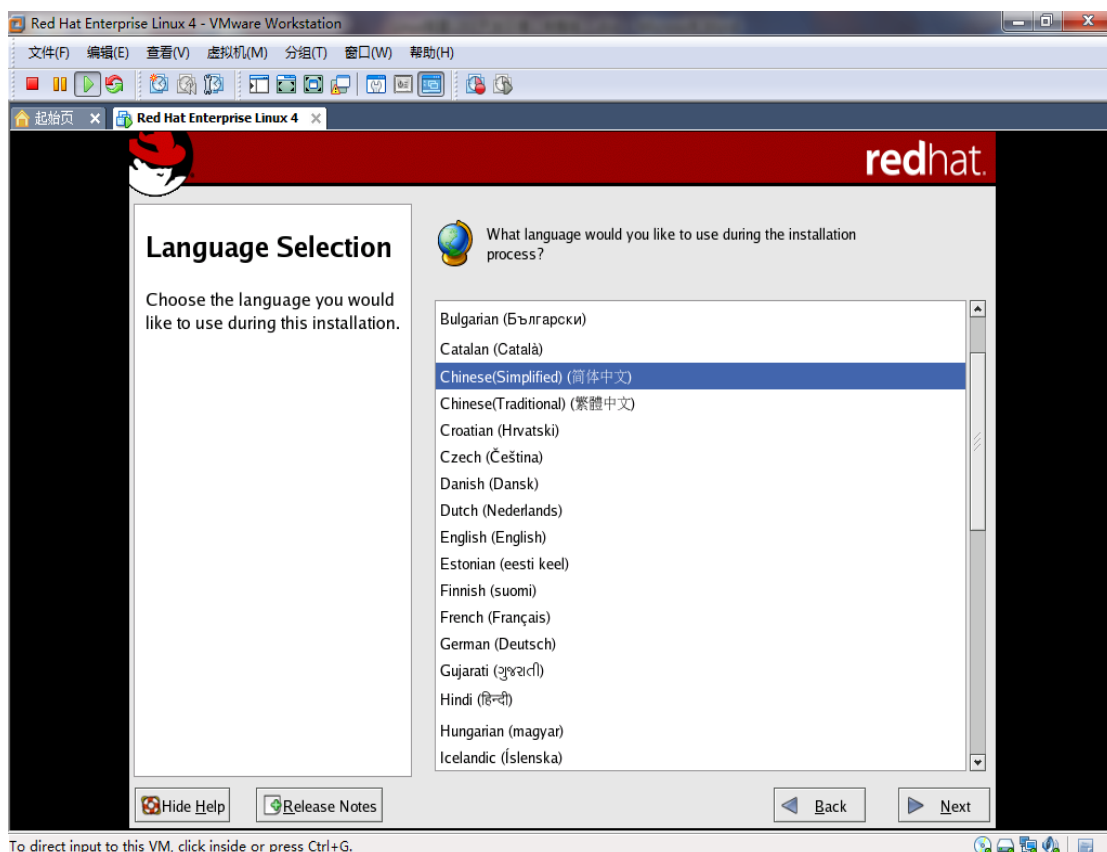
点击 Finish 按钮，出现如下界面，进入 Linux 安装。



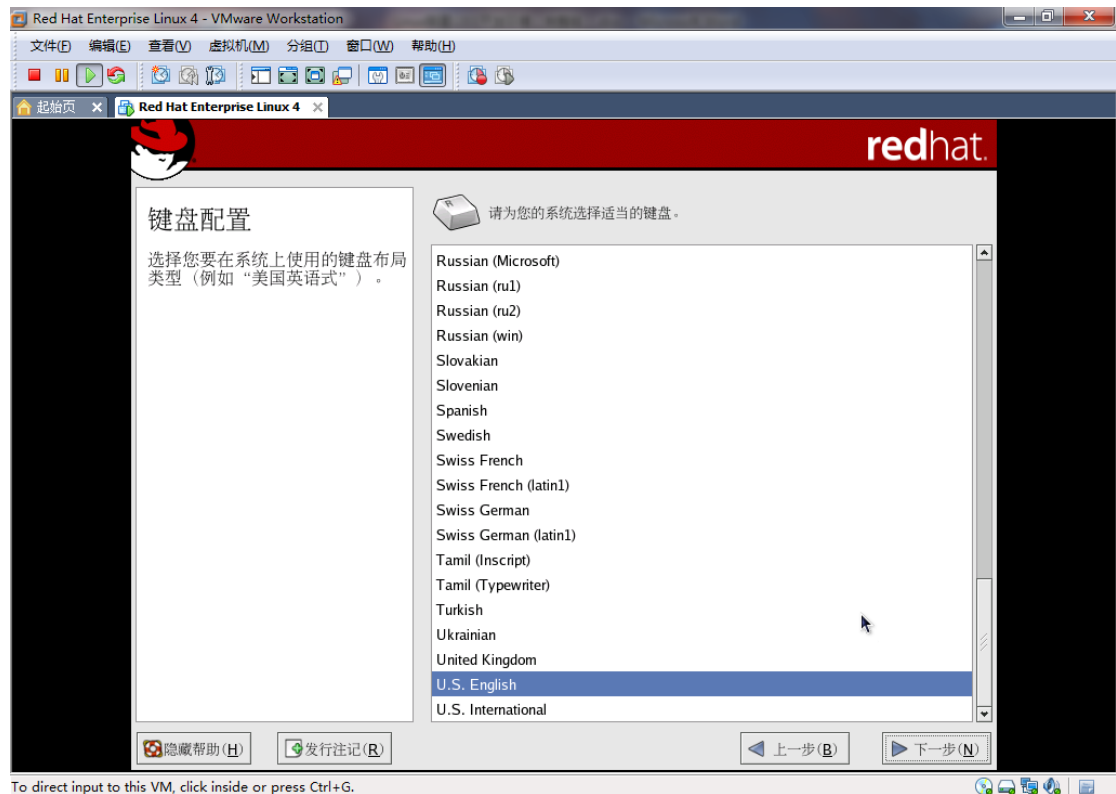
在检测文件完整性的时候选择“SKIP”即可，出现如下界面：



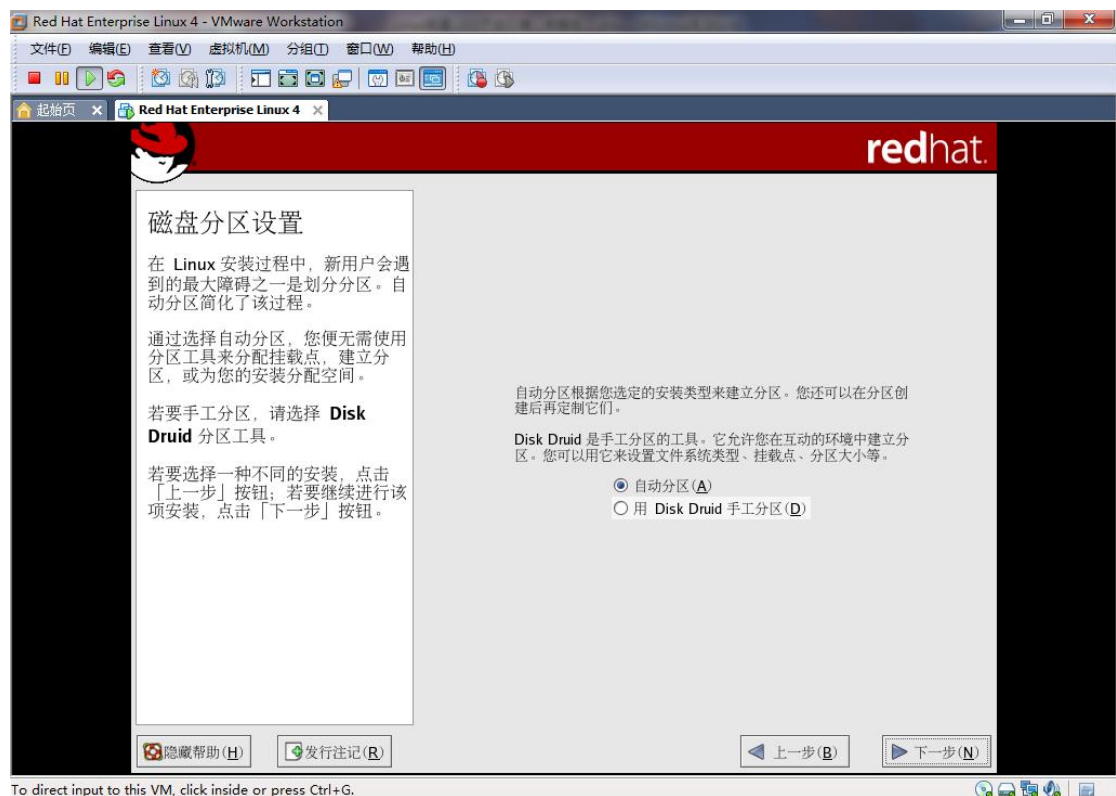
点击 Next 按钮，出现如下界面：



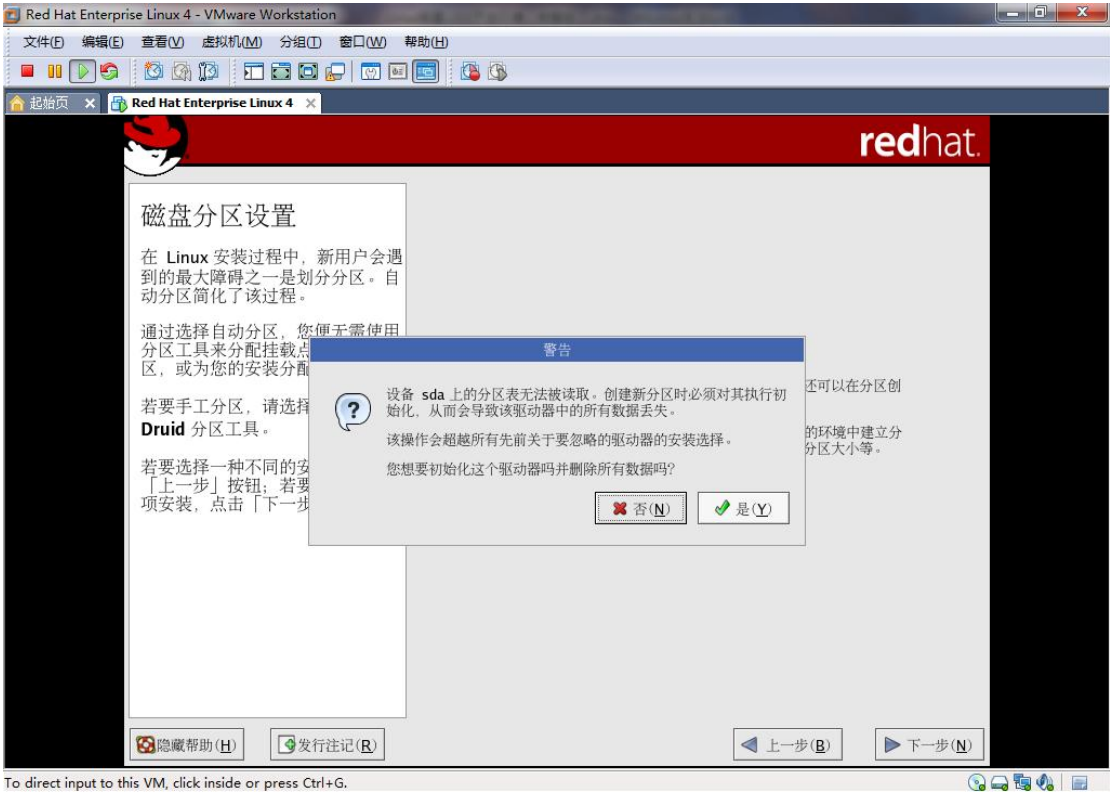
根据个人喜好选择 Linux 系统的语言，此处我选择“简体中文”，点击 Next 出现如下界面：



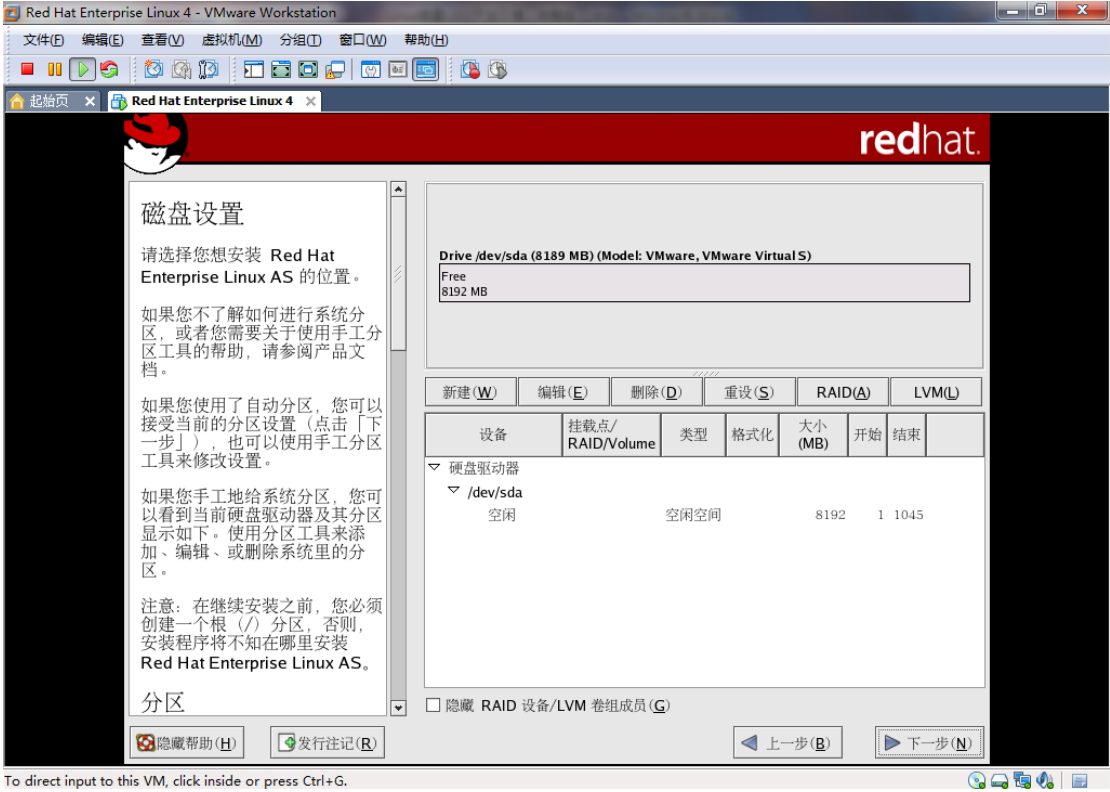
此处不需要修改，点击“下一步”按钮即可，出现如下界面：



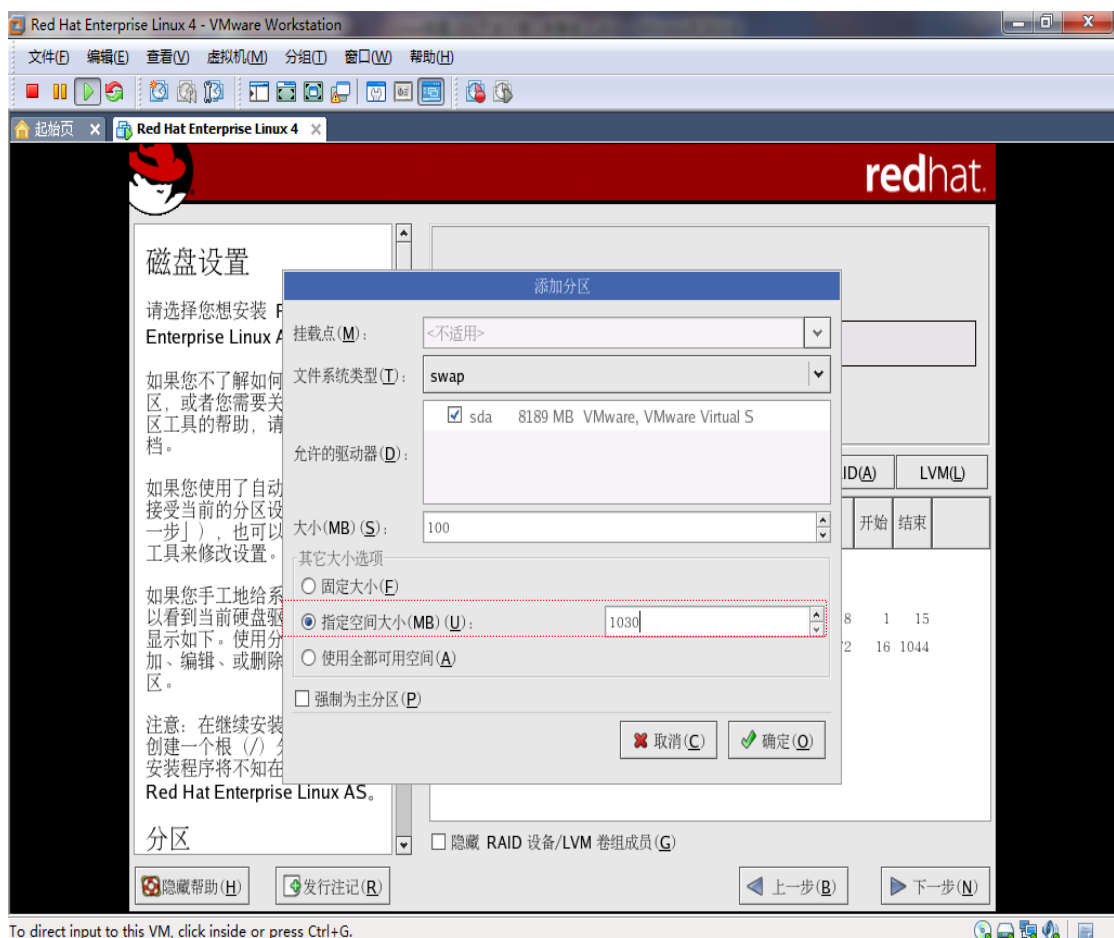
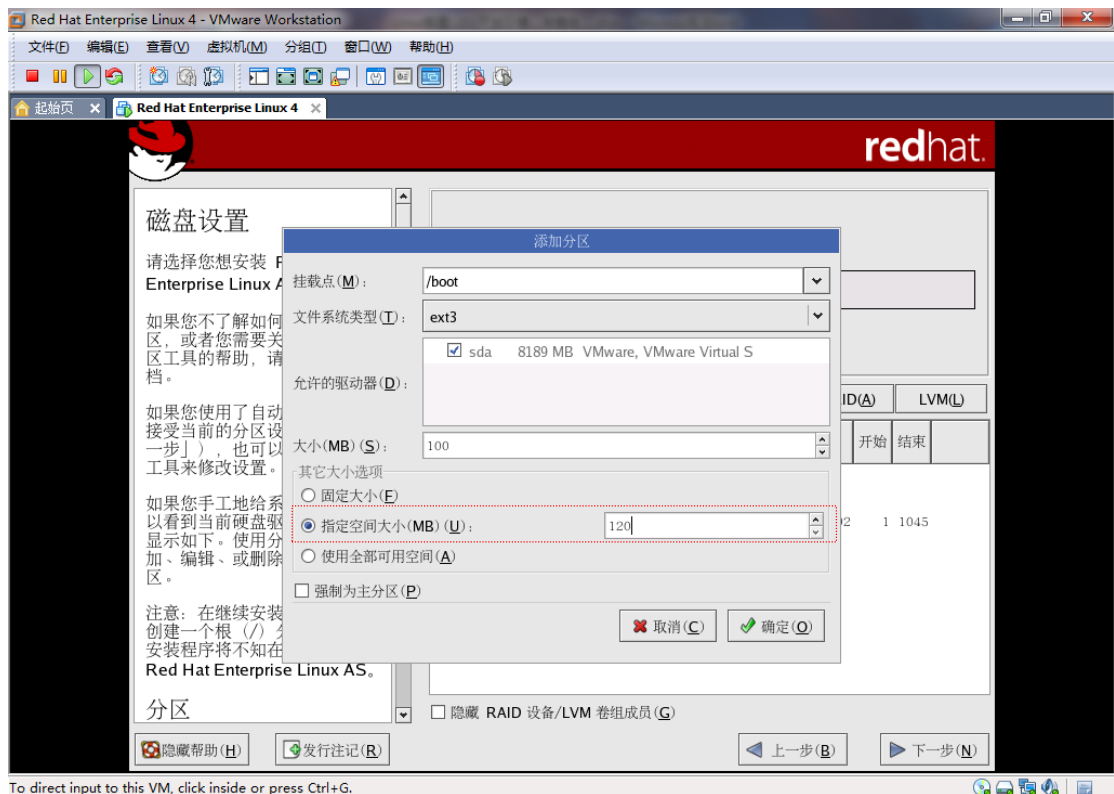
分区，此处不要选择“自动分区”，选择“用 Disk Druid 手工分区”，点击“下一步”出现如下界面：

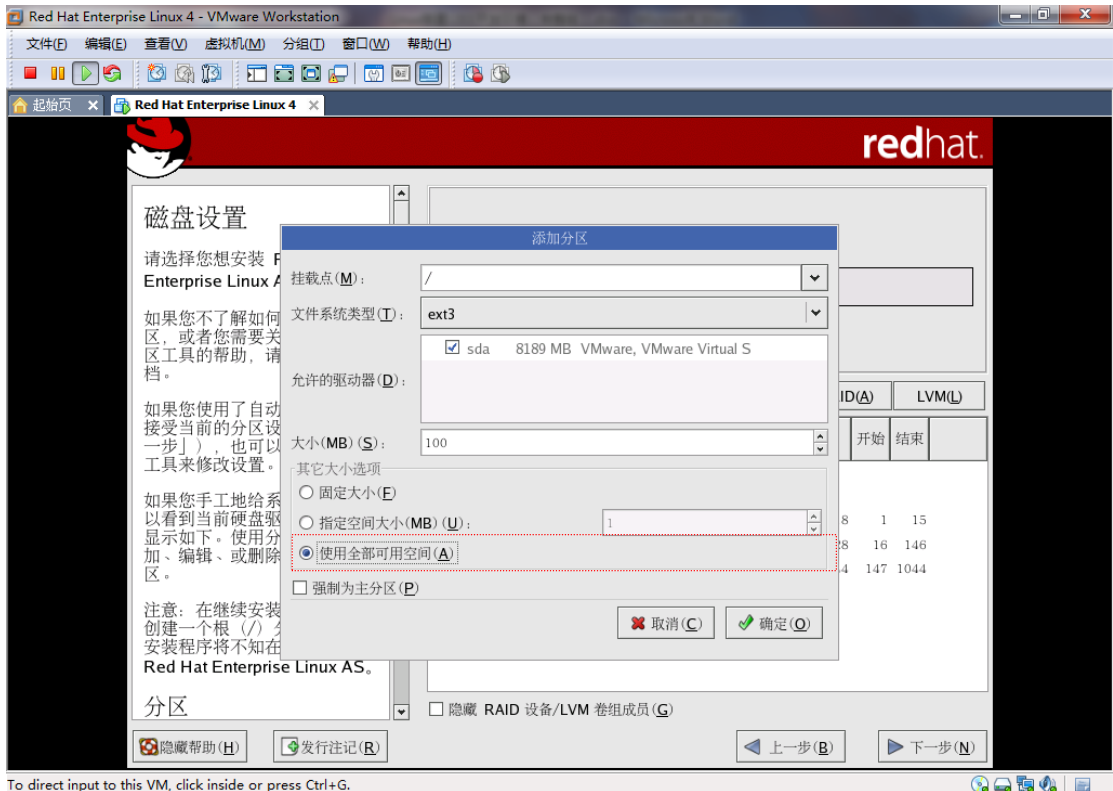


弹出警告，点击“是”按钮，出现分区界面如下：

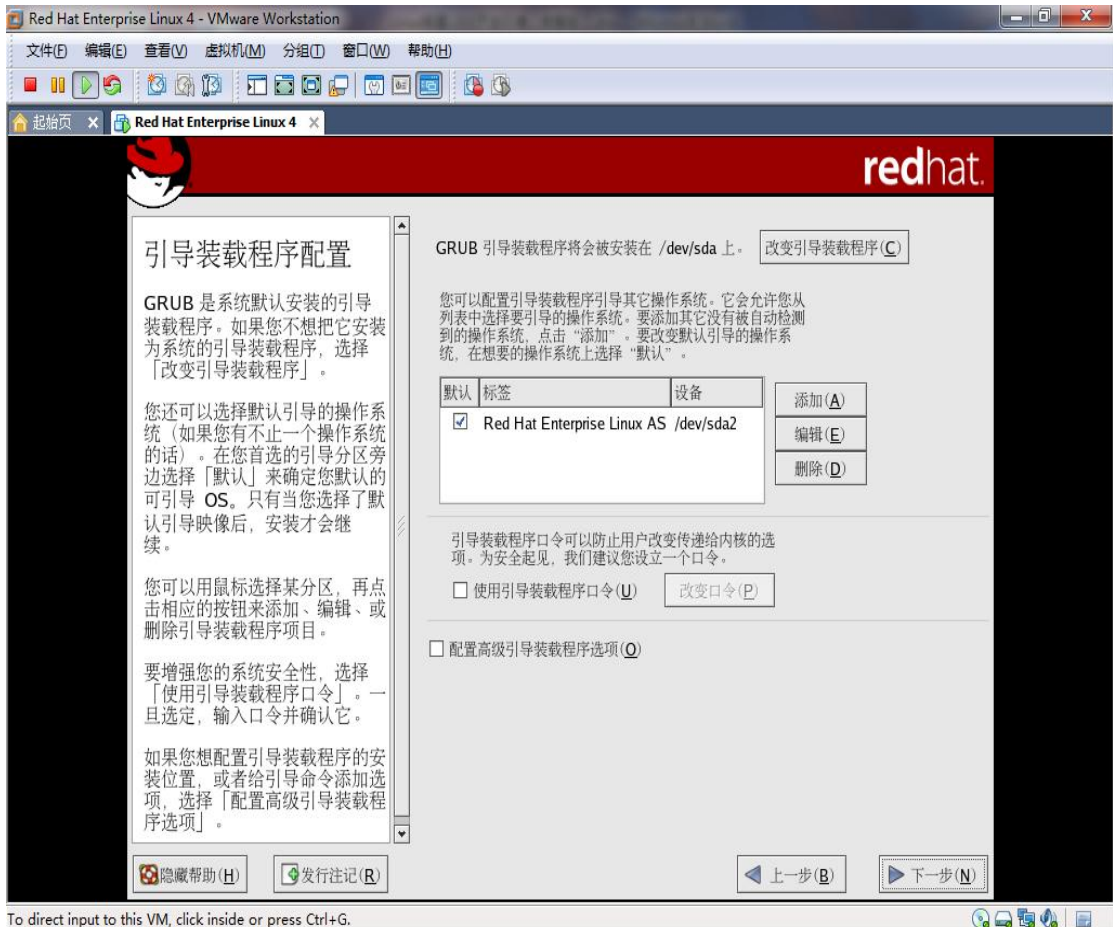


分区主要分为 3 个区：/boot 分区、Swap 分区、/分区，三个分区的值详见如下界面：

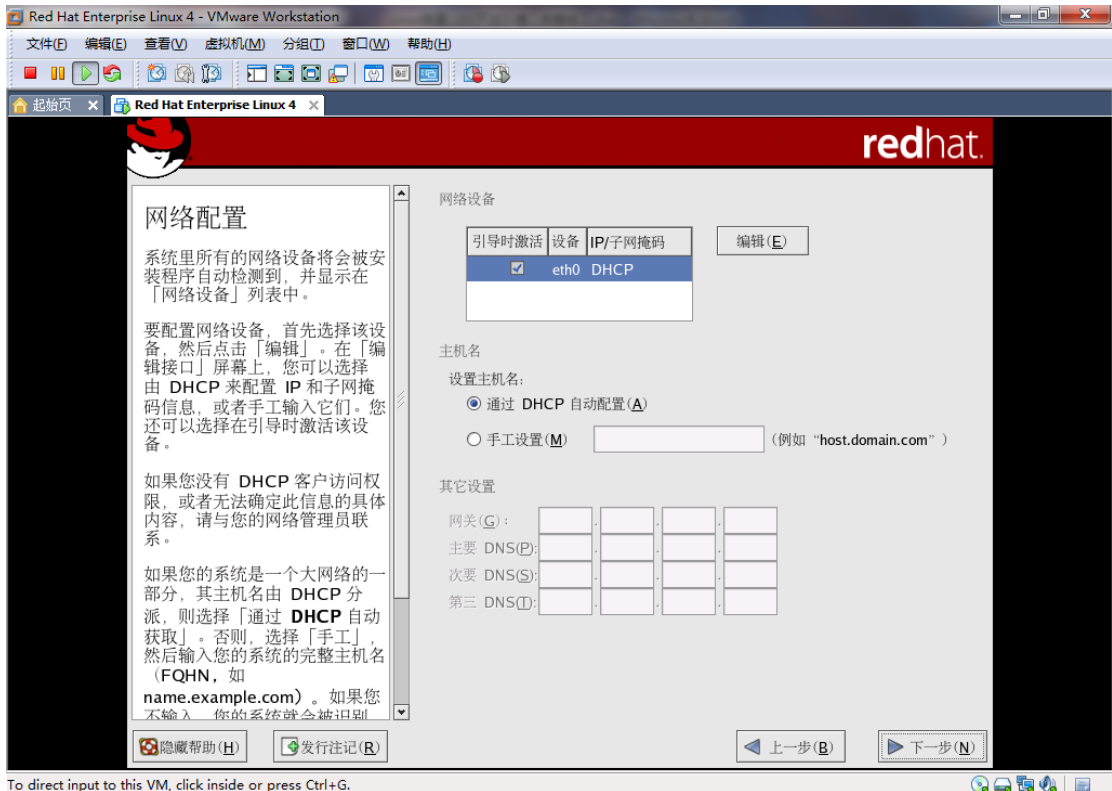




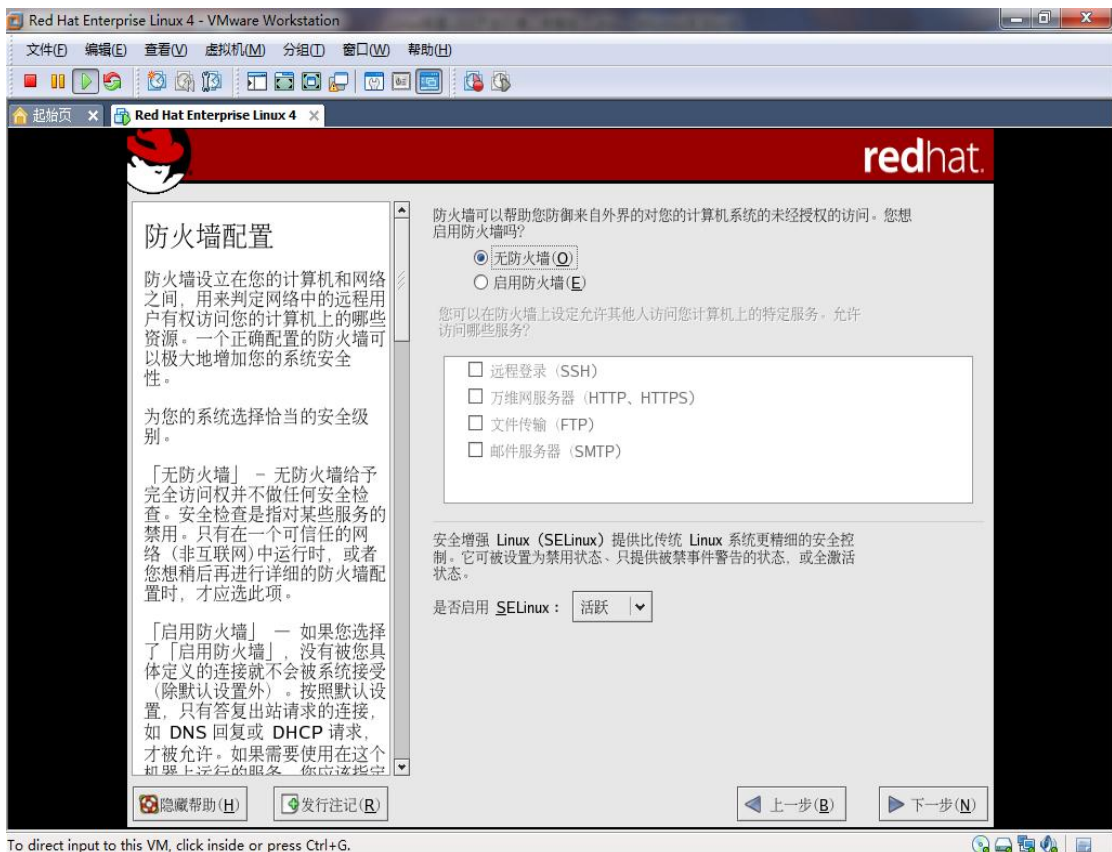
分区设置好后，点击“下一步”按钮出现如下界面：



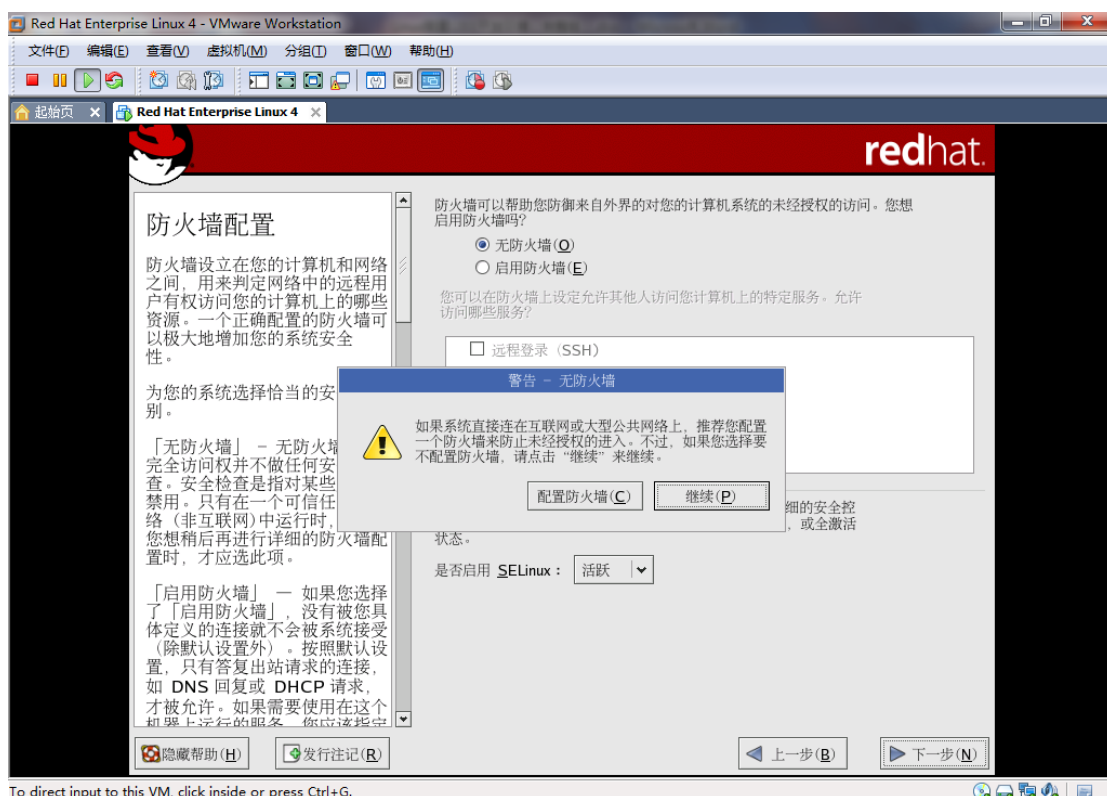
由于我们是自己使用，无需设置安全级别，直接点击“下一步”按钮出现如下界面：



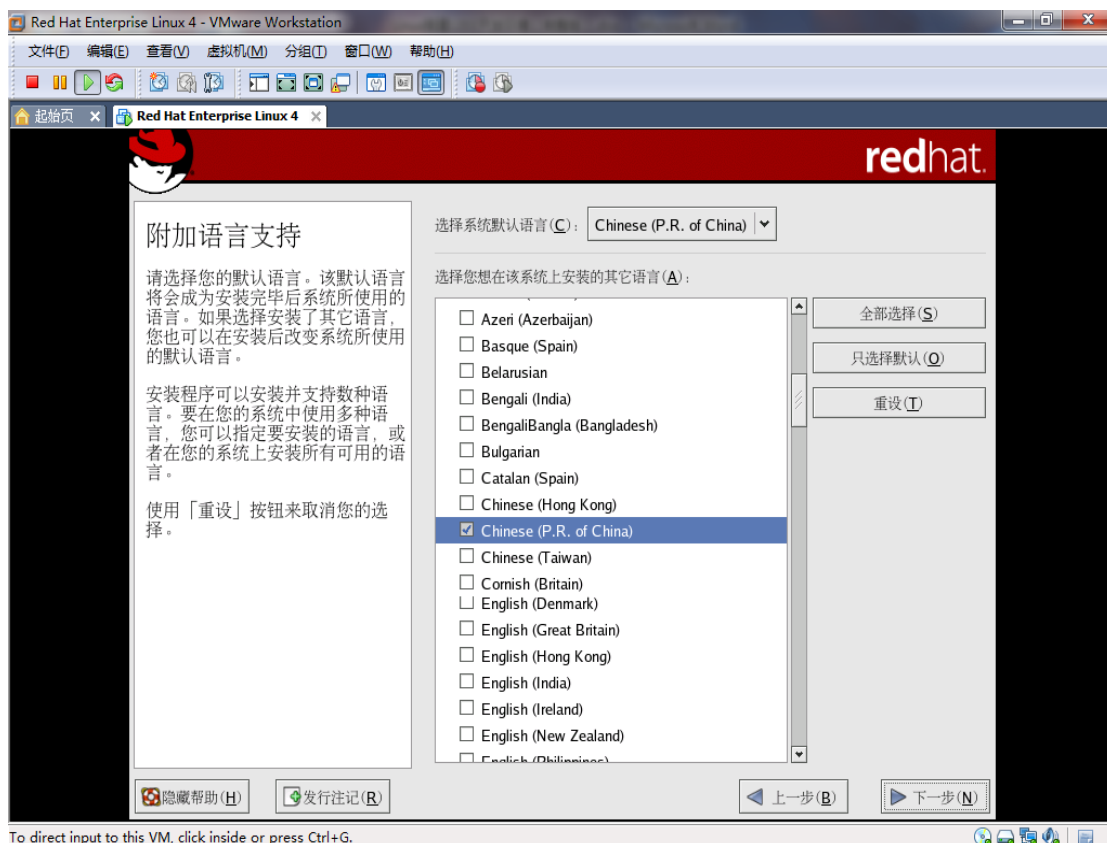
此处默认，后期通过 XWindow 界面设置，点击“下一步”按钮出现如下界面：



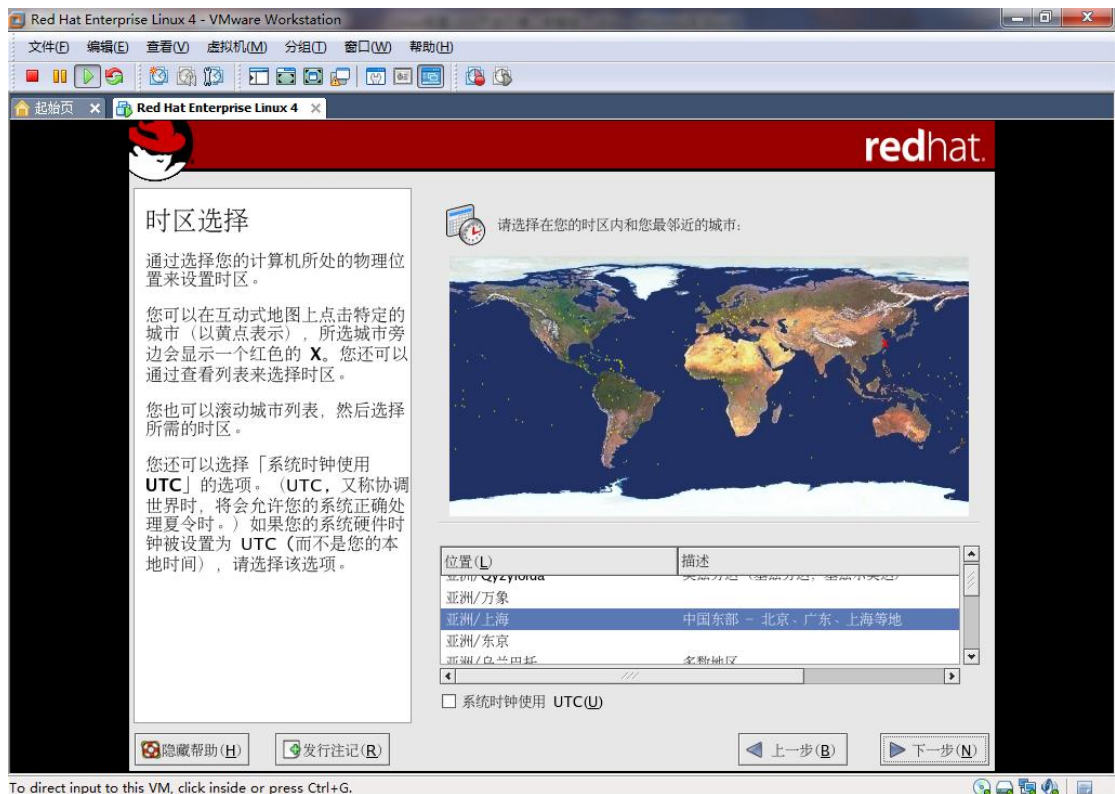
此处选择无防火墙，点击“下一步”按钮出现如下界面：



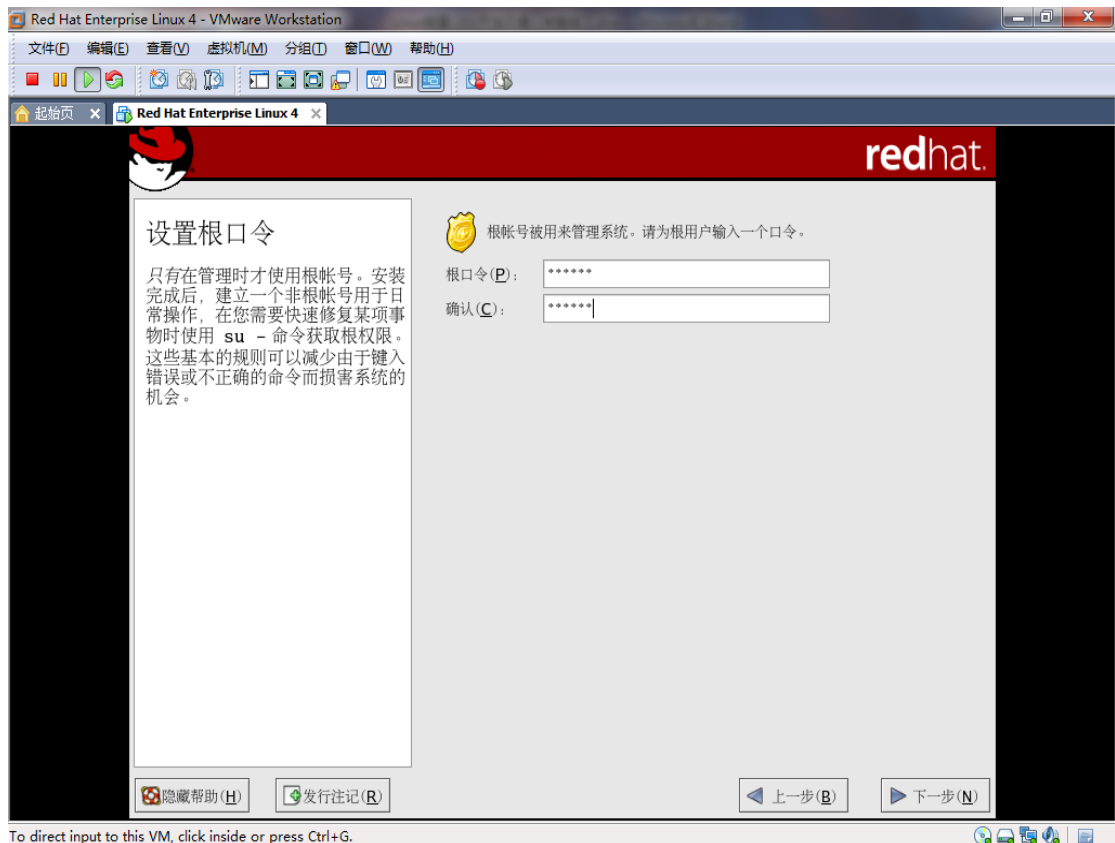
弹出警告，点击“继续”按钮后出现如下界面：



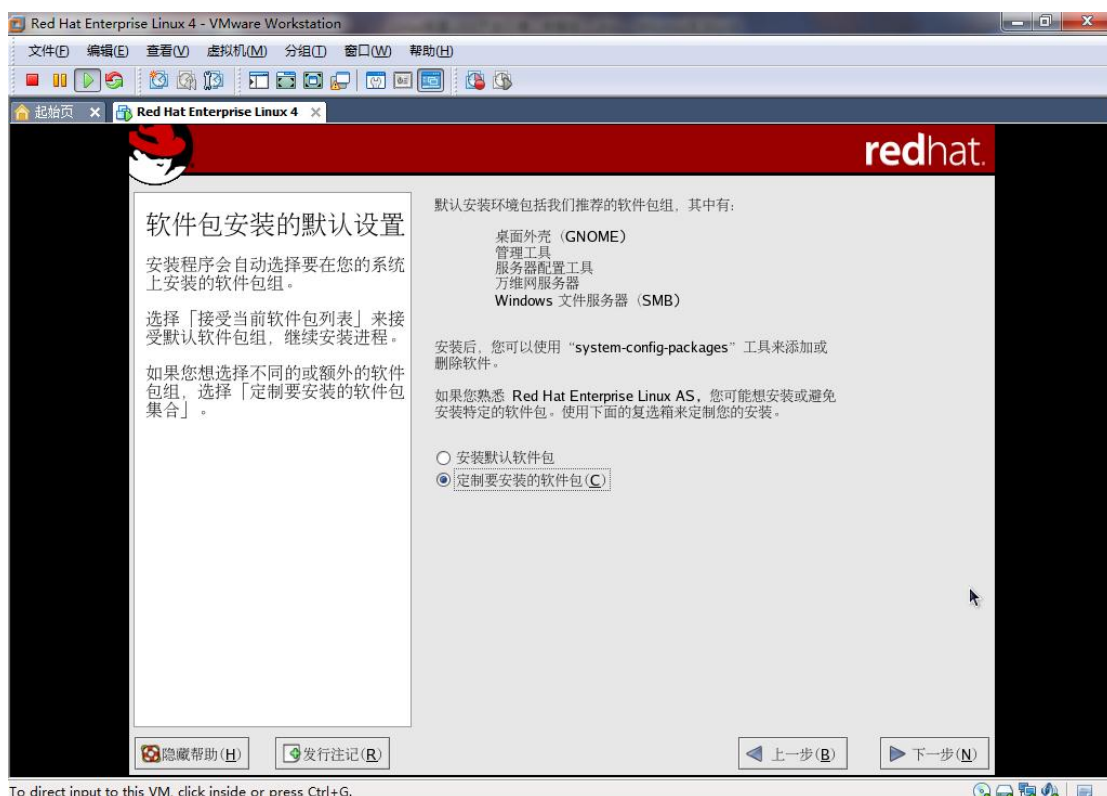
此处无需更改，点击“下一步”按钮出现如下界面：



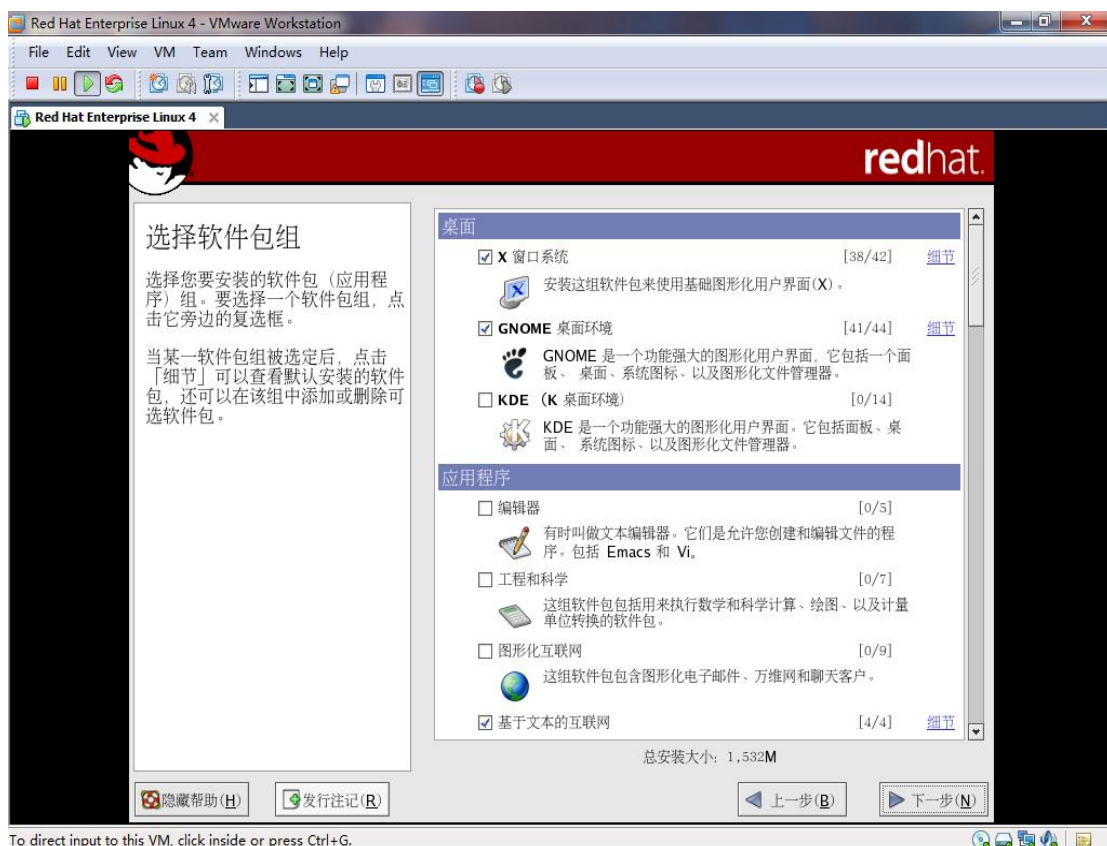
此处无需更改，点击“下一步”按钮出现如下界面：



设置好 root 用户的密码，点击“下一步”按钮出现如下界面：



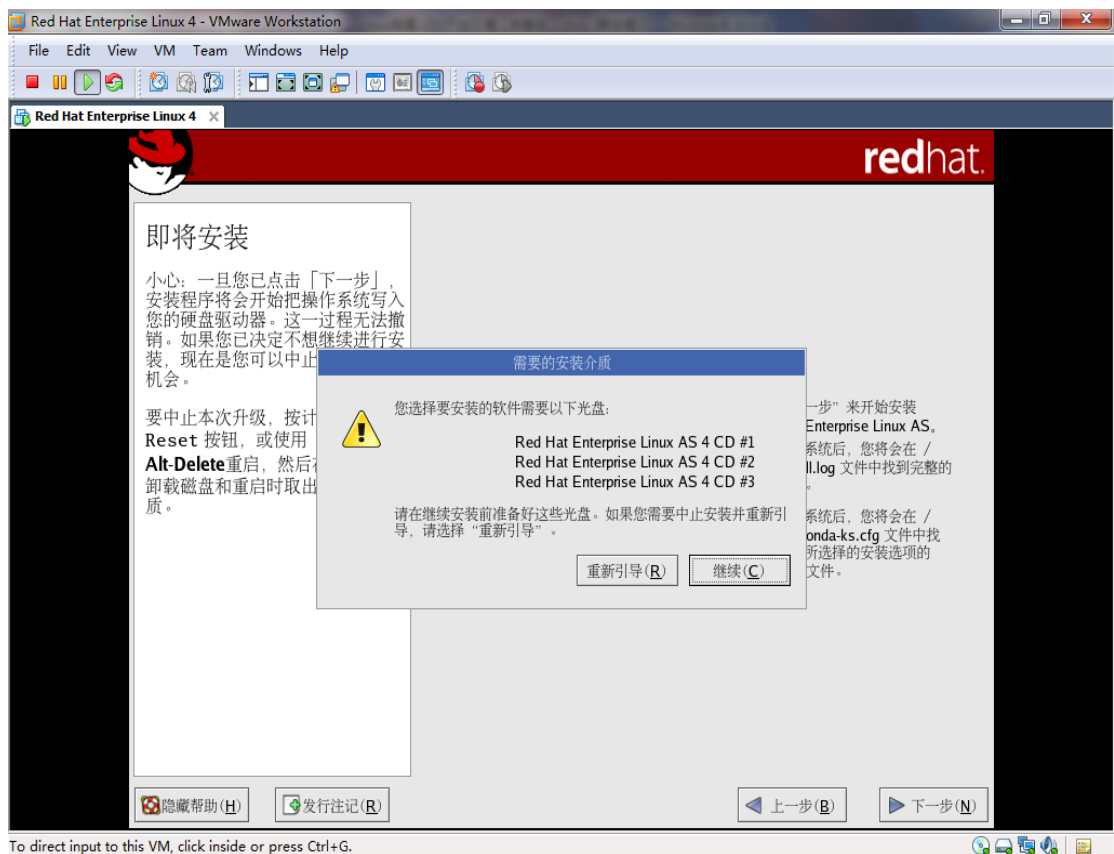
此处选择“定制要安装的软件包”，点击“下一步”按钮出现如下界面：



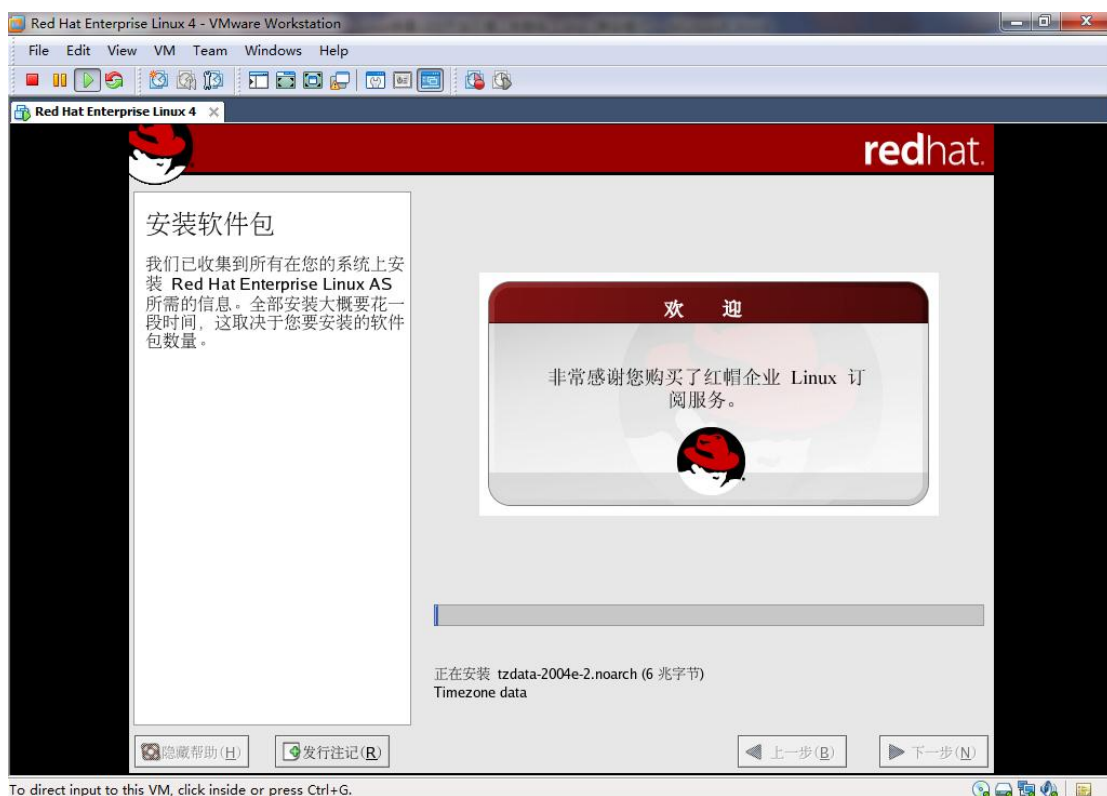
根据需要进行软件安装，点击“下一步”按钮出现如下界面：



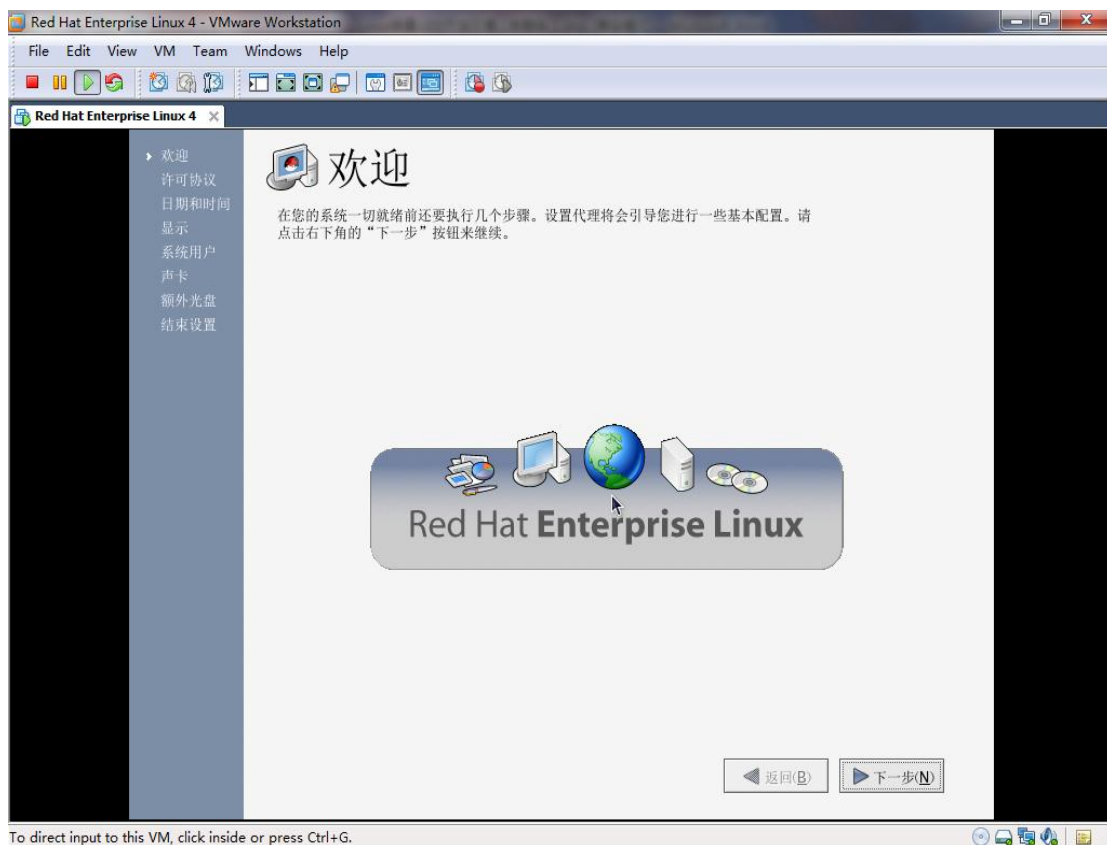
点击“下一步”按钮，出现安装界面：



出现提示点击“继续”按钮，出现如下界面：



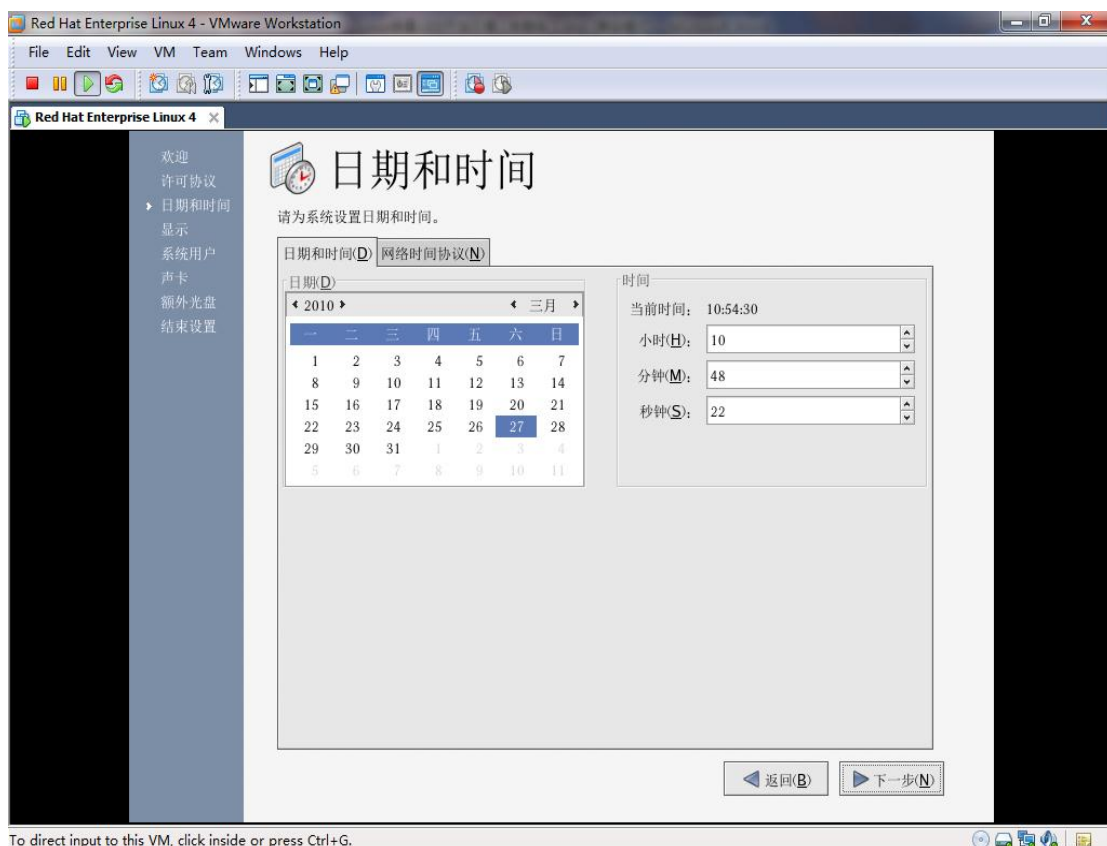
等待安装。。。。。。，安装完成后出现如下界面：



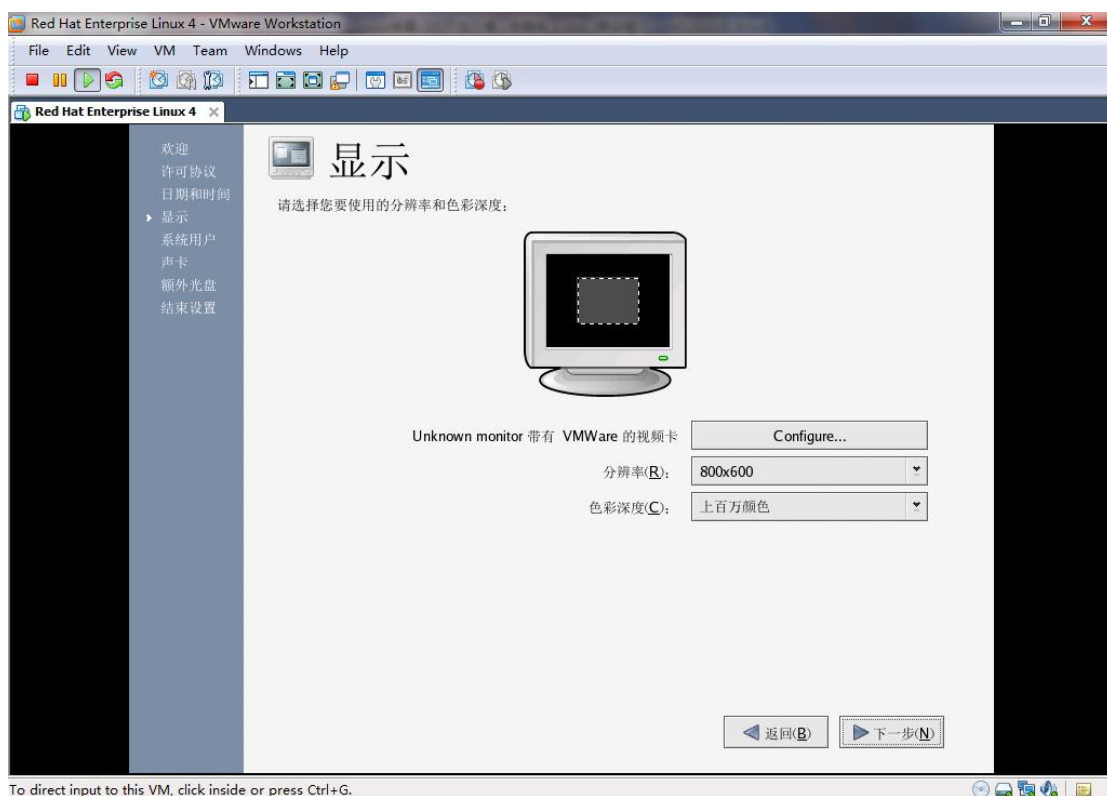
点击“下一步”按钮出现如下界面：



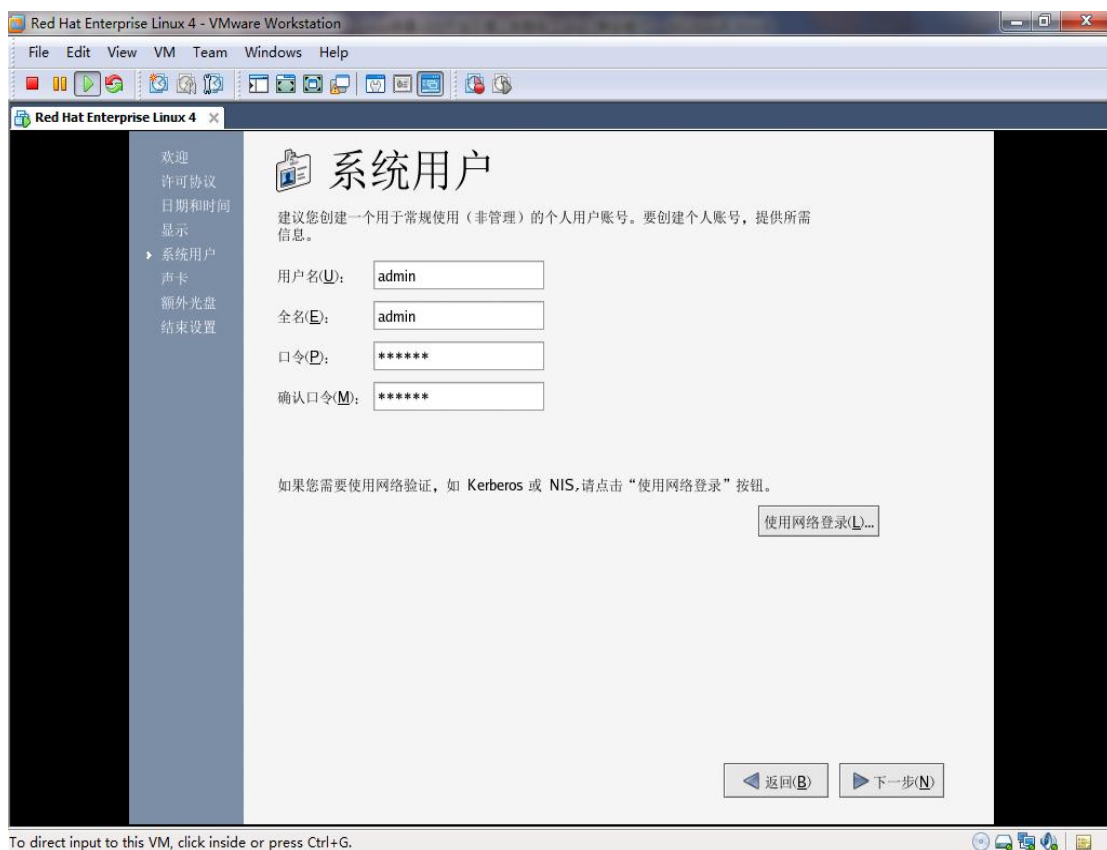
点击“同意协议”，点击“下一步”，出现如下界面：



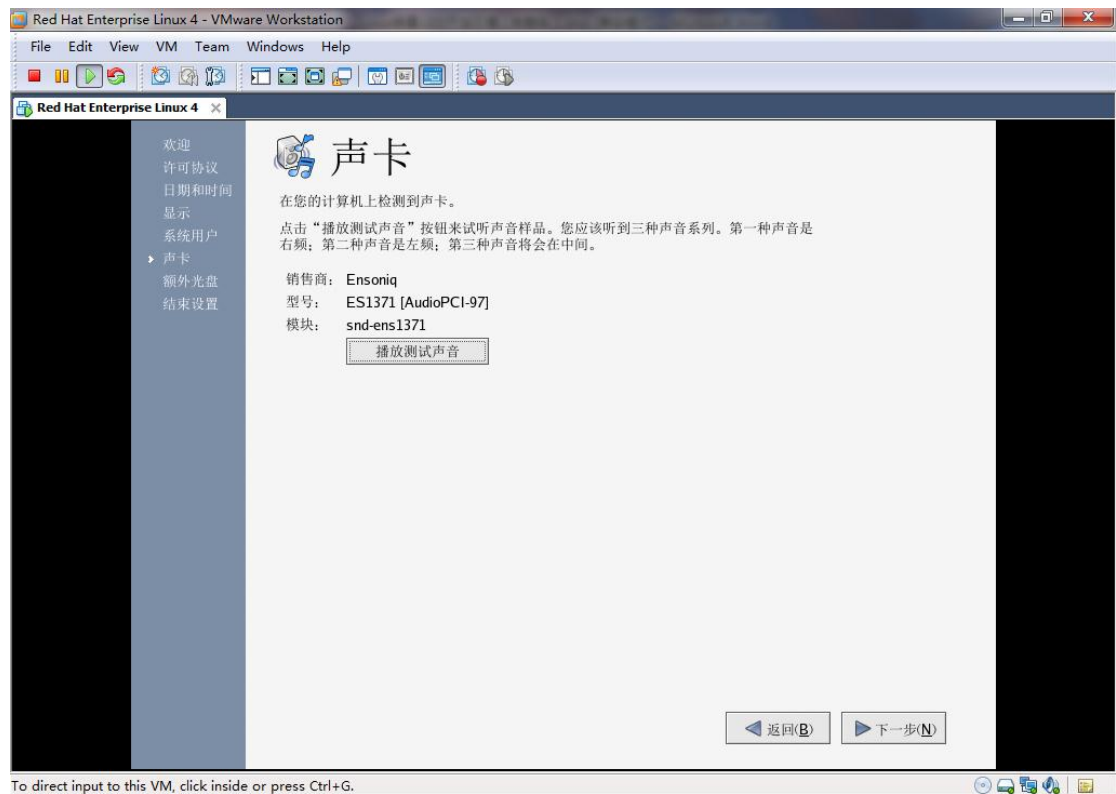
点击“下一步”按钮出现如下界面：



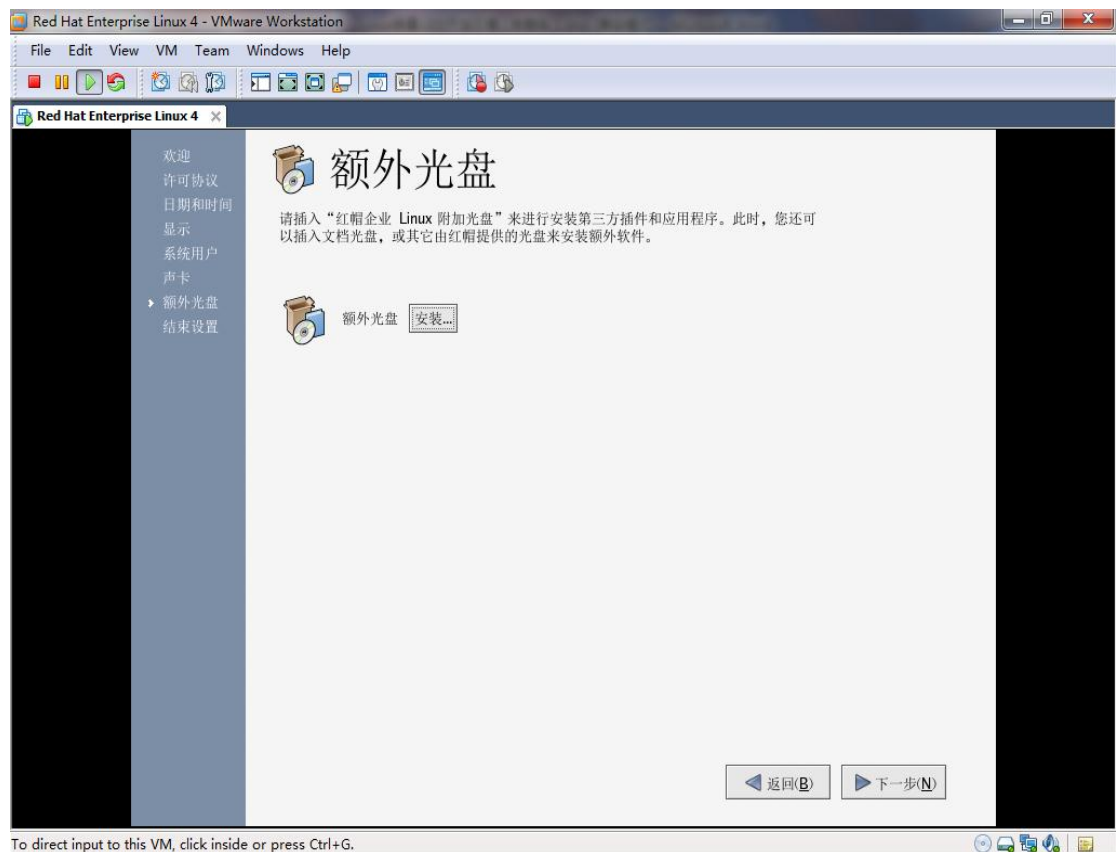
点击“下一步”按钮出现如下界面：



输入用户名和密码，点击“下一步”出现如下界面：



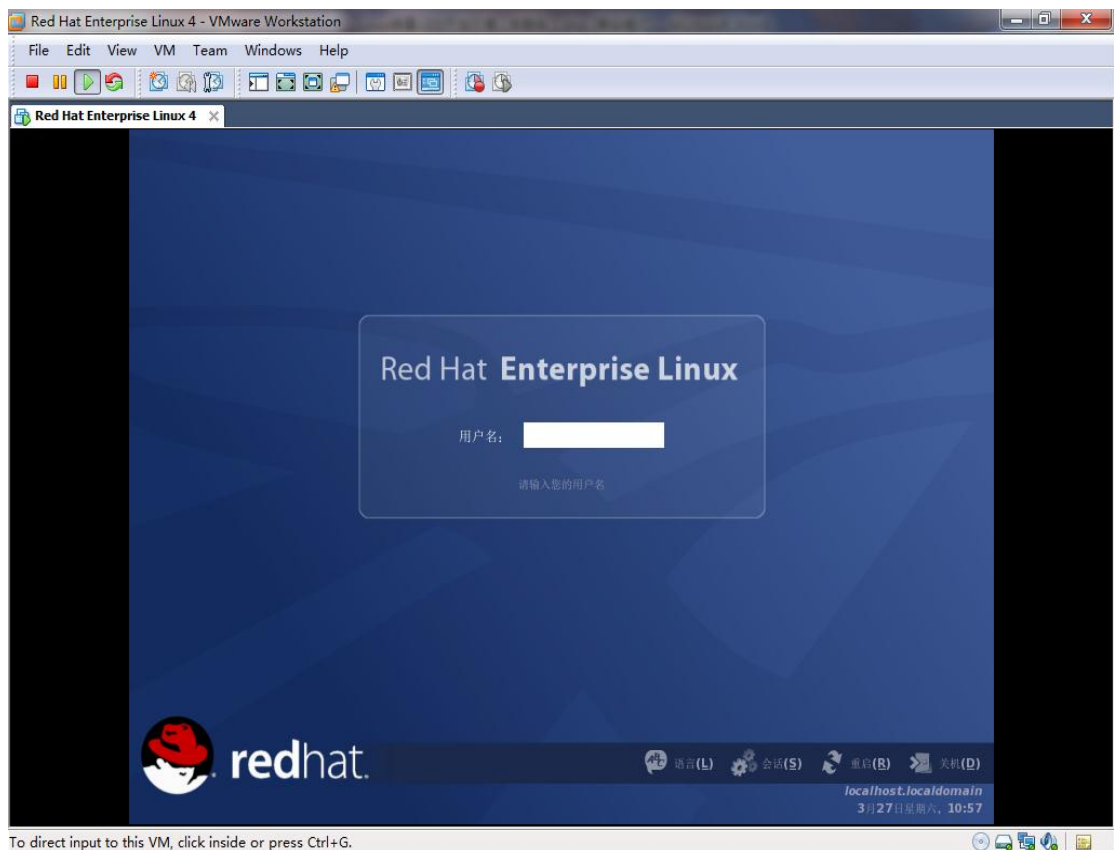
点击“下一步”按钮出现如下界面：



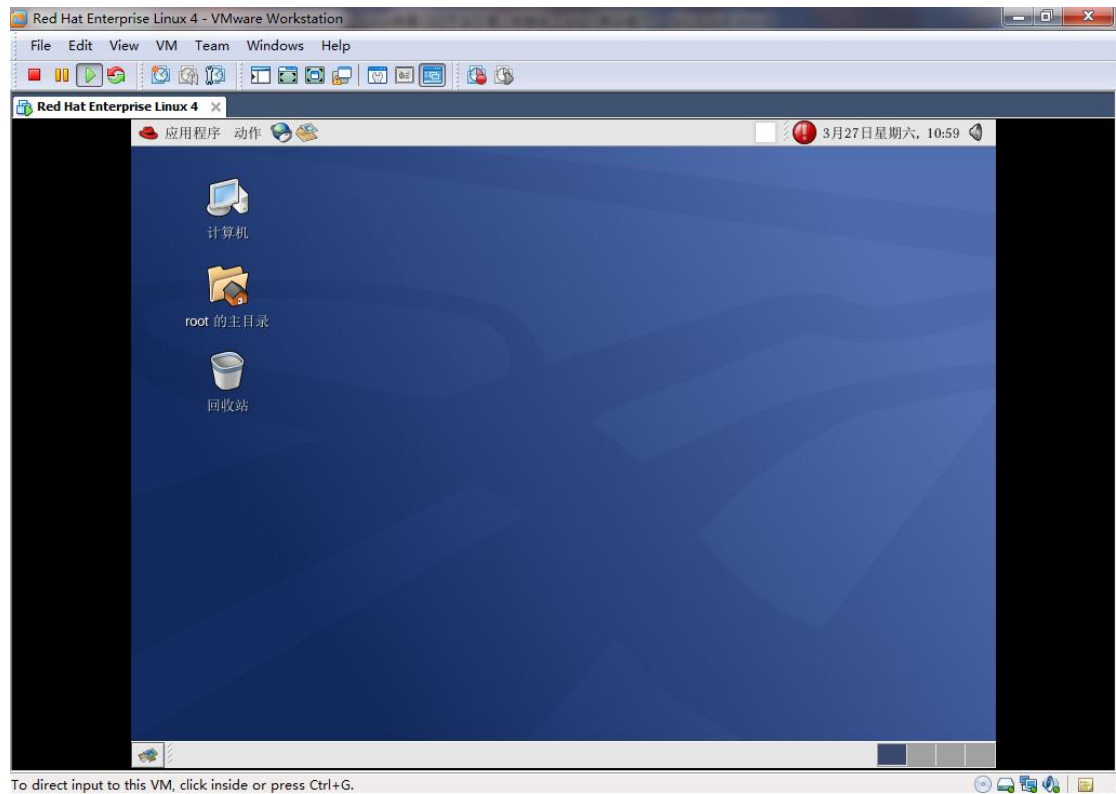
点击“下一步”按钮出现如下界面：



点击“下一步”按钮出现如下界面：



输入用户名和密码，此处以 root 用户登录，因为要设置一下 Linux 系统的相关内容，登录后出现如下界面：

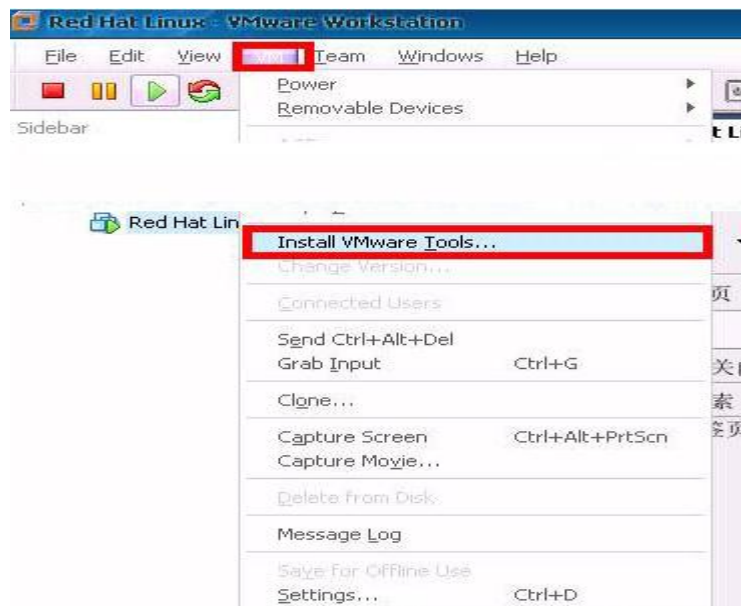


至此，我们的 Linux 就算是安装完成了，但是现在只是安装好了 Linux，并没有什么实质意义，既不能联网也不能与宿主机（Windows）互相通信，要想实现这些功能需要对 Linux 进行一些更为细致的配置，下面我将讲解如何在 Linux 下上网以及如何实现 Windows 与 Linux 进行通信。

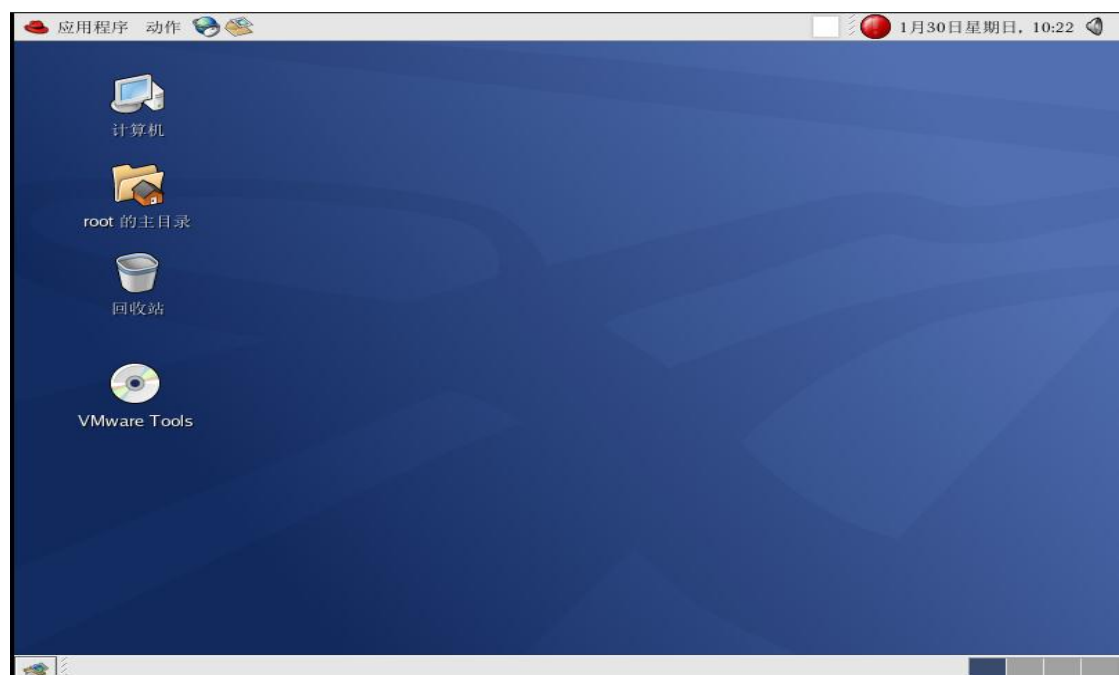
LINUX 网络环境配置

本阶段主要讲解如何在 Linux 下进行上网, 以及如何实现 Linux 与宿主机进行相互通信, 这部分是最重要的一部分, 没有这部分后面的软件安装操作将无法进行。闲话少絮, 咱们直接进入主题。

要想实现 Linux 上网, 第一步我们需要安装 VMware 为我们提供的工具, 具体步骤如下:



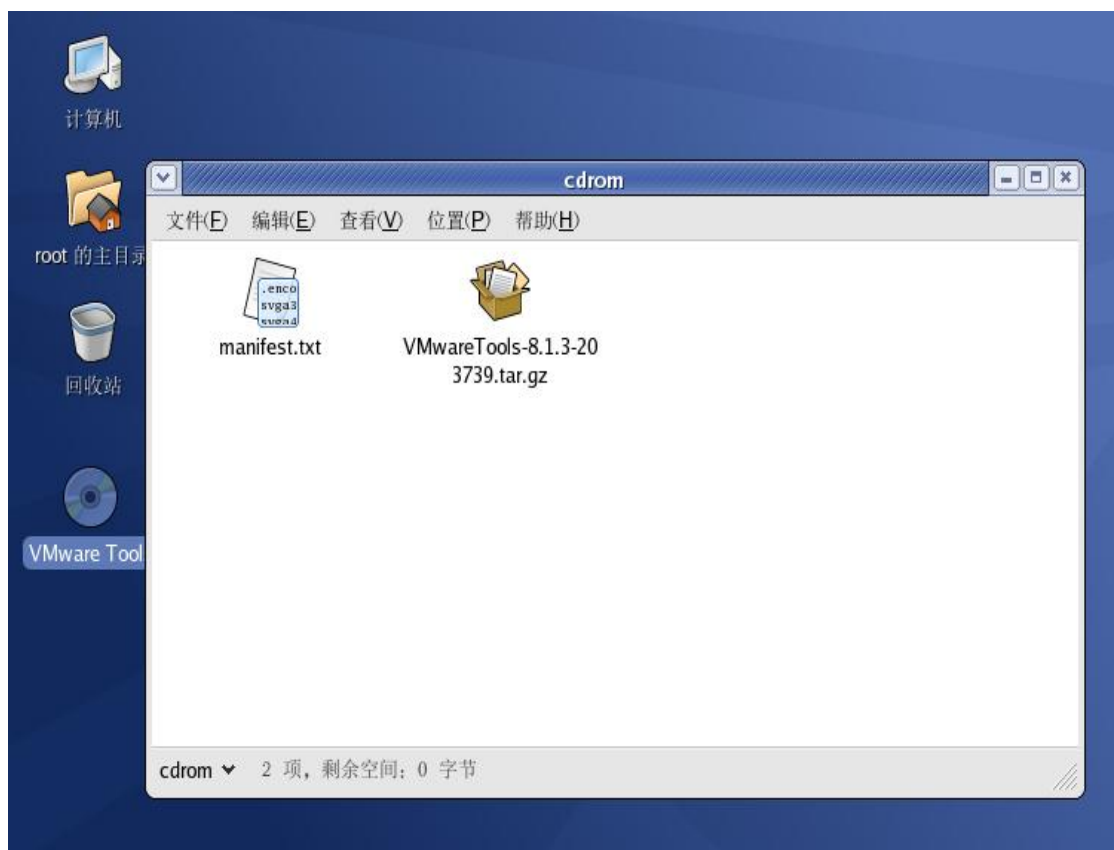
选择 VM→Install WMware Tools...菜单后, Linux 将自动挂载光驱, 界面如下:



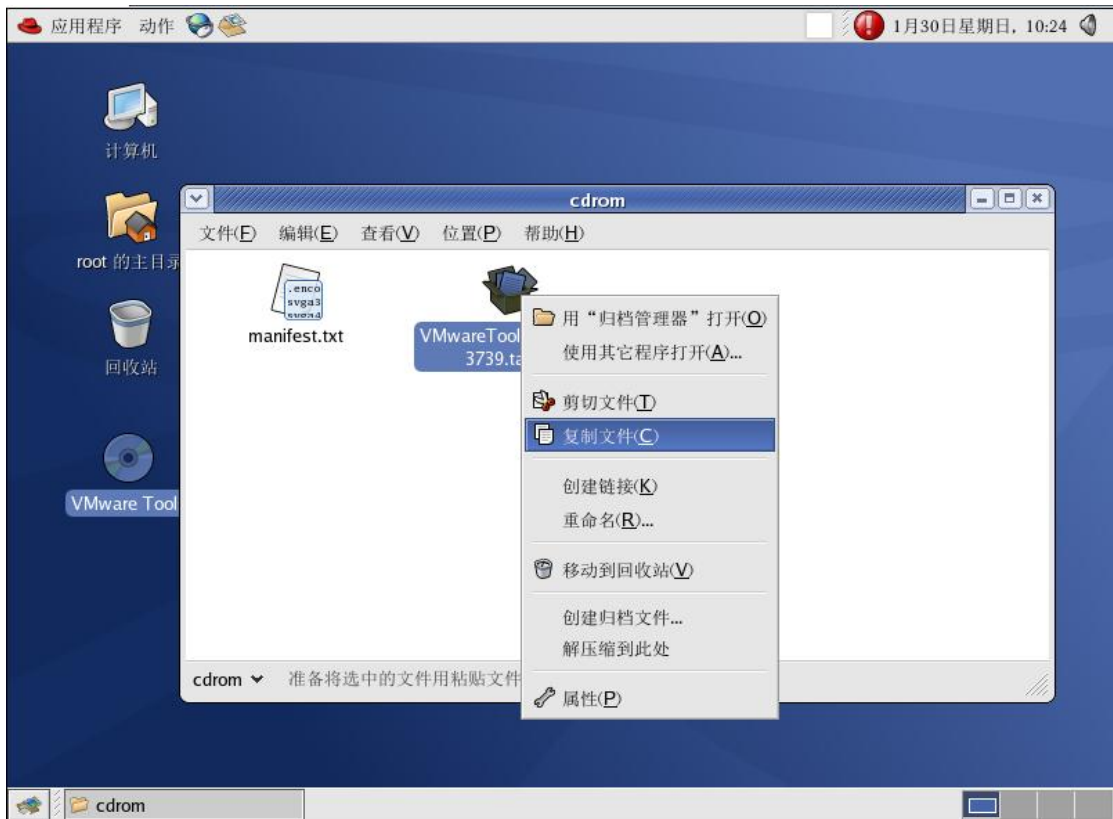
右键在 VMware Toos 上单击，出现如下界面：



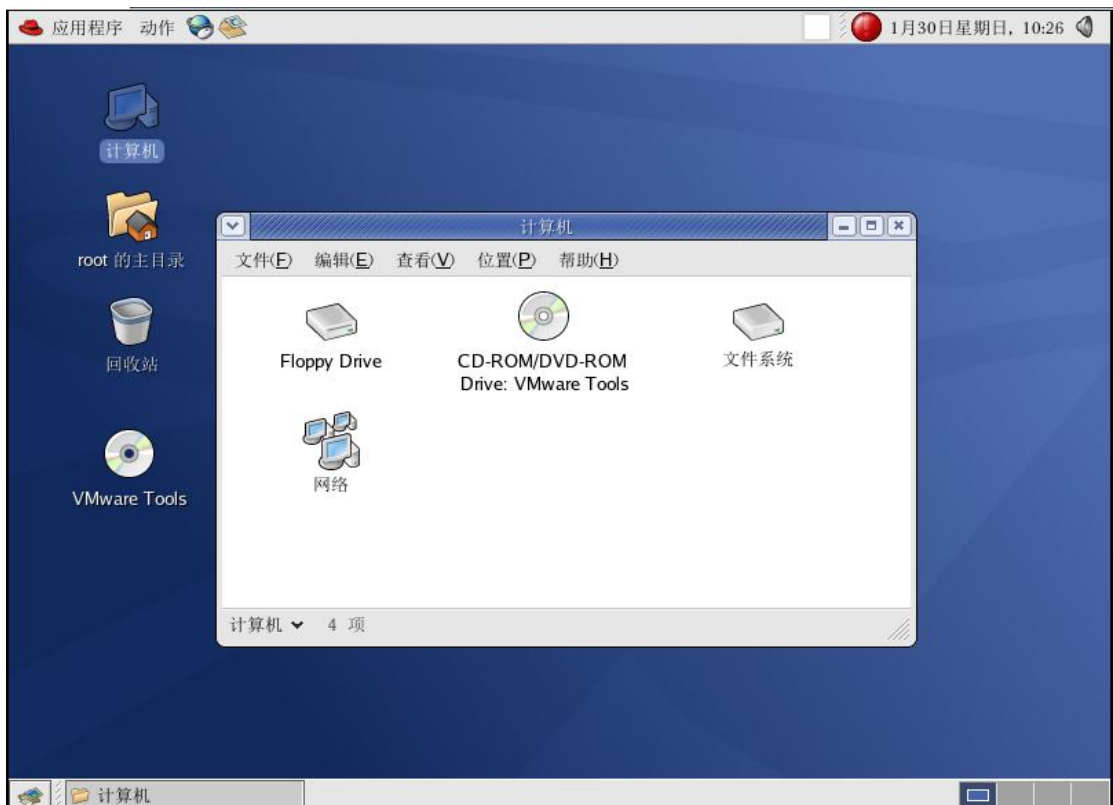
单击“打开”，进入 Linux 光驱后，出现如下界面：



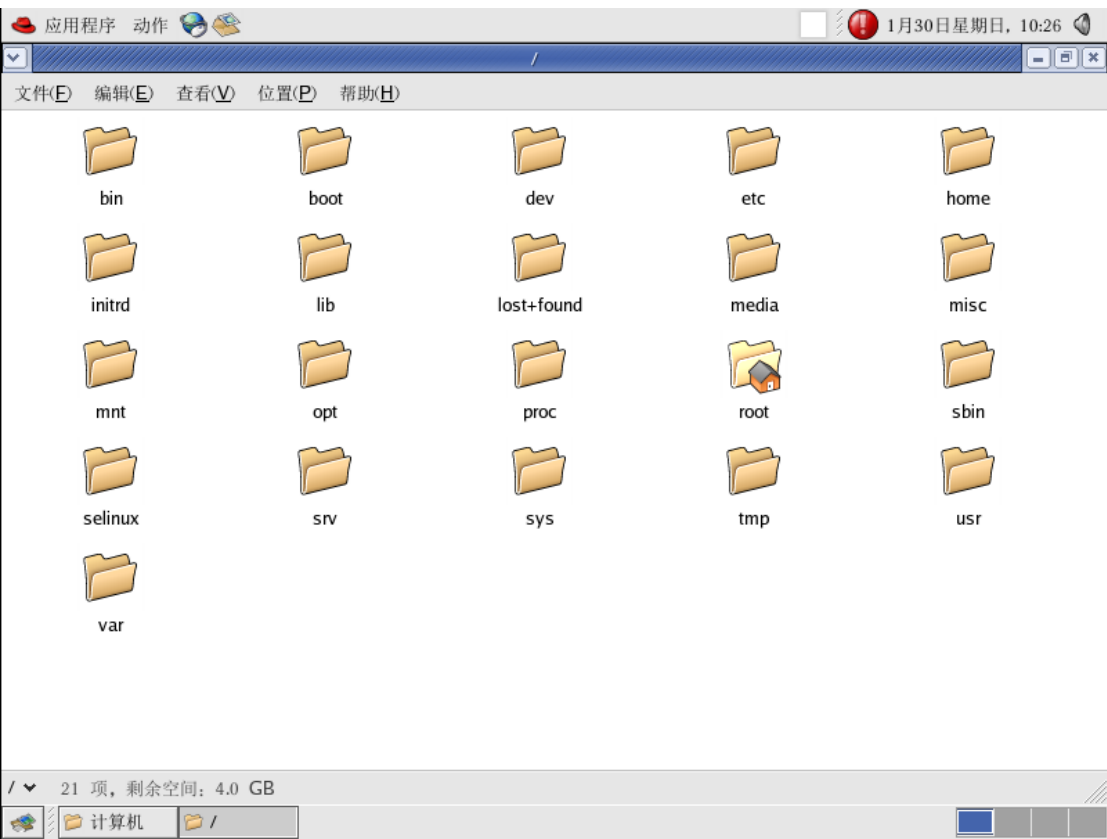
右键单击 WMwareTools-xxx.tar.gz 文件，选择“复制文件”：



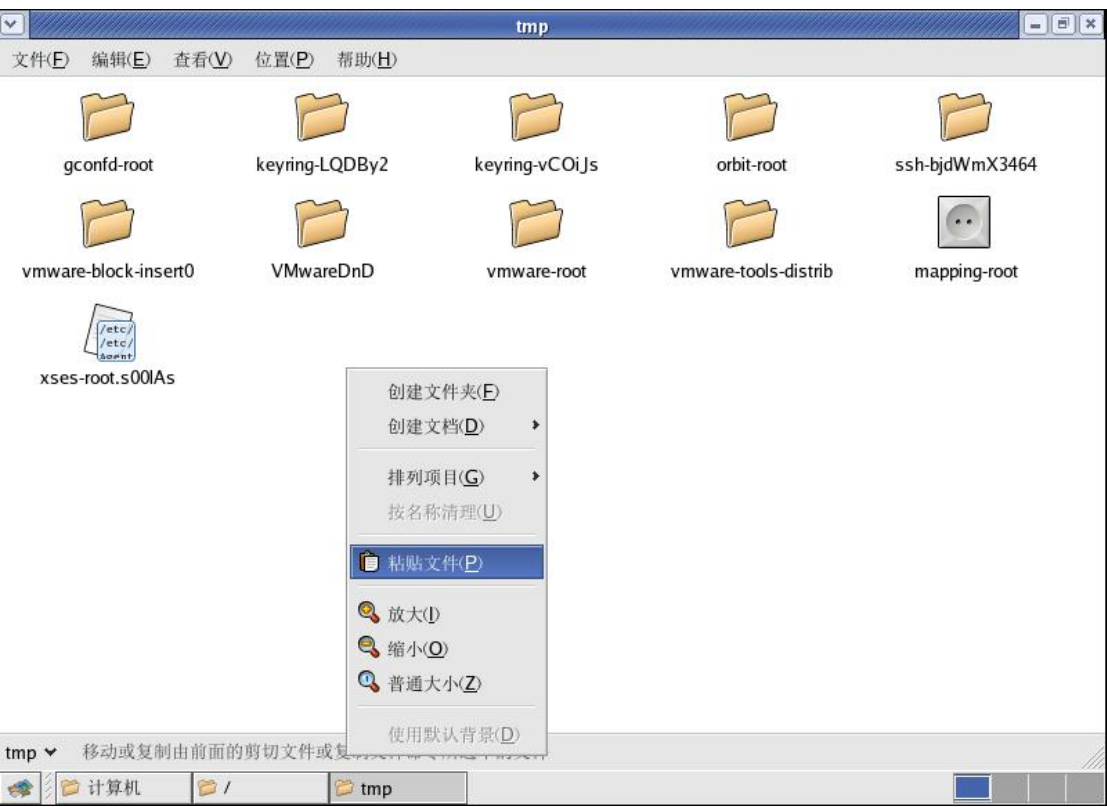
复制后，点击关闭按钮，右键单击“计算机”，选择“打开”，出现如下界面：

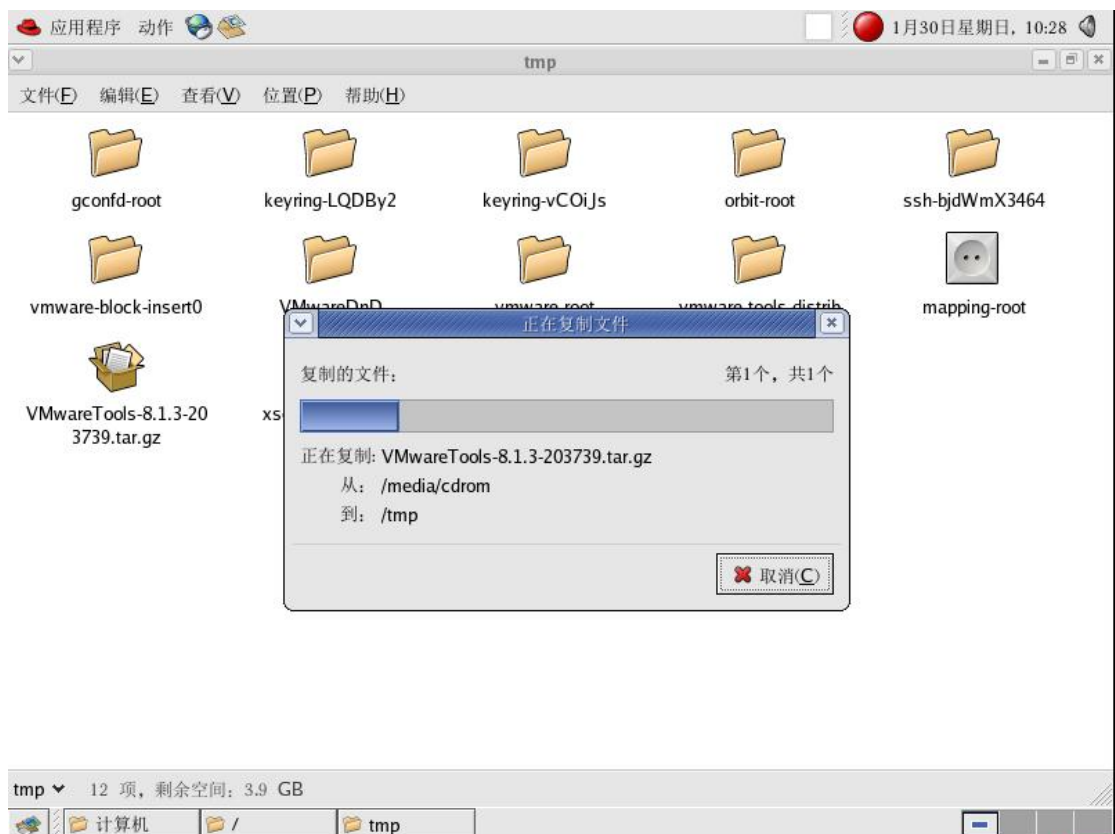


双击“文件系统”图标，弹出如下界面：



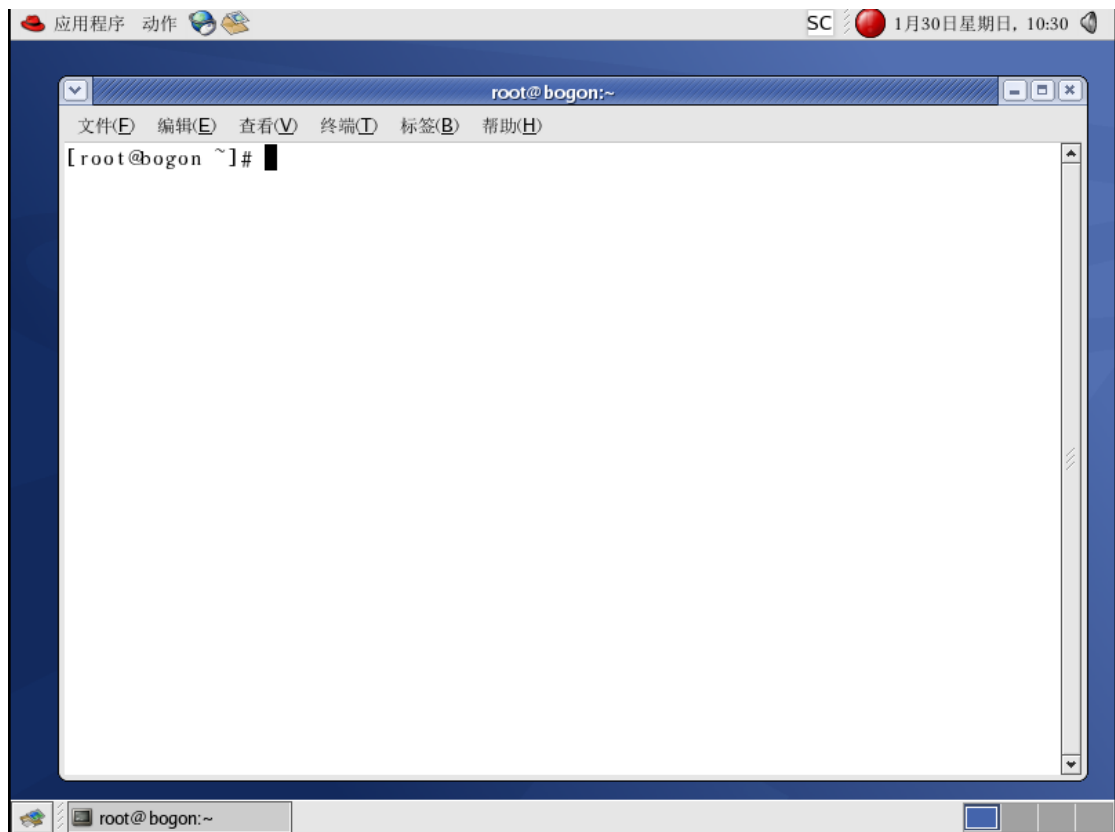
双击“tmp”目录，进入该目录后，单击右键，选择“粘贴文件”：



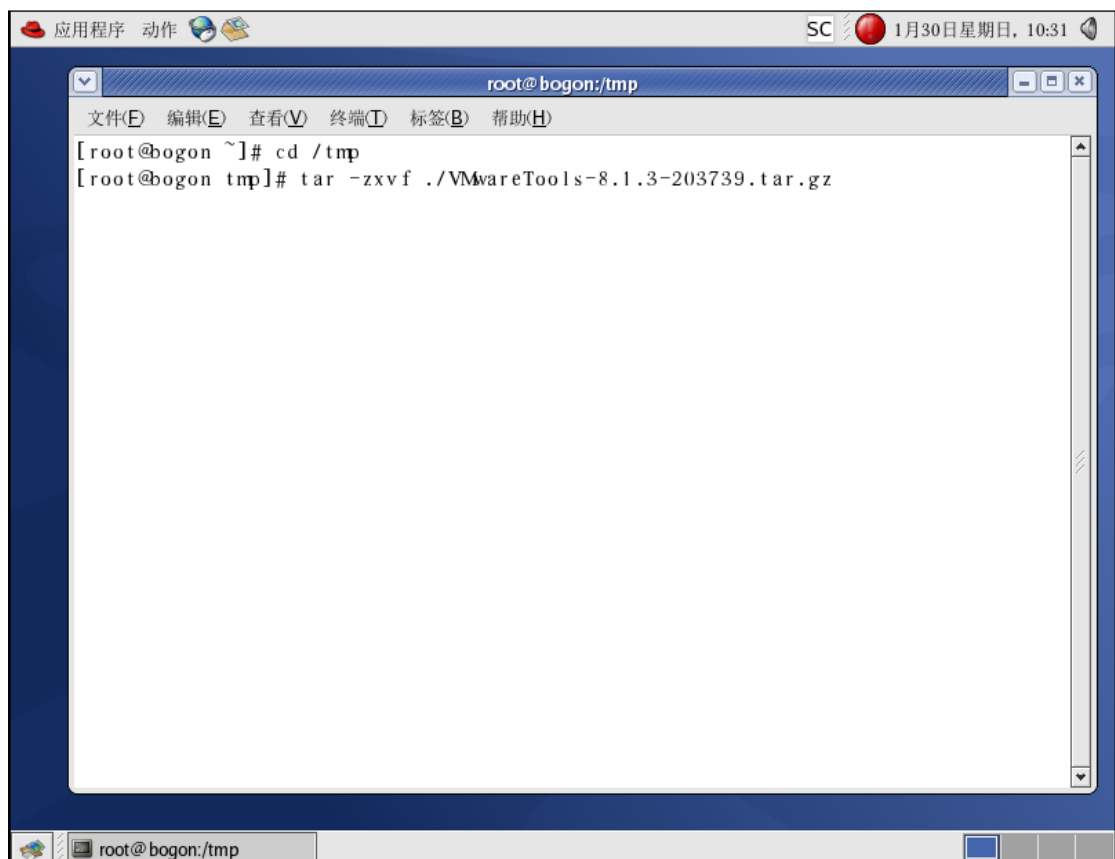


粘贴完成后，关闭所有已经打开的窗口，在主界面右键，选择“打开终端”菜单，界面如下：

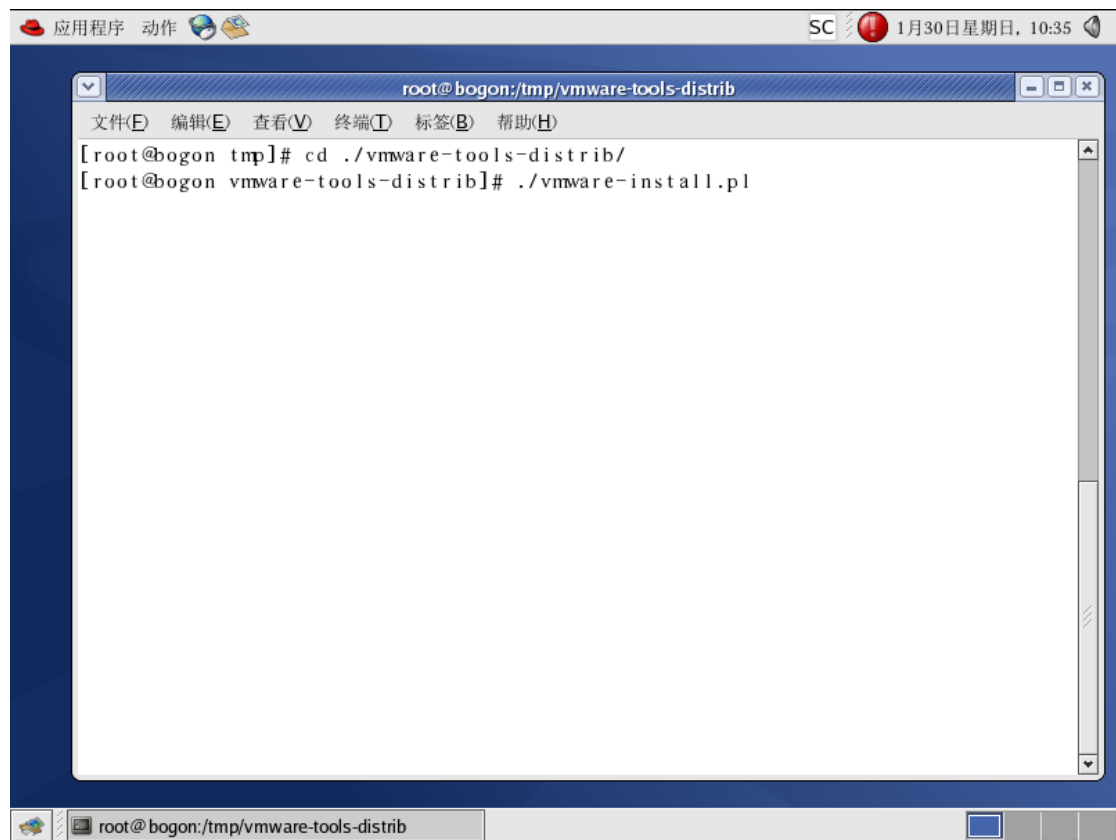




在终端输入如下图所示命令：

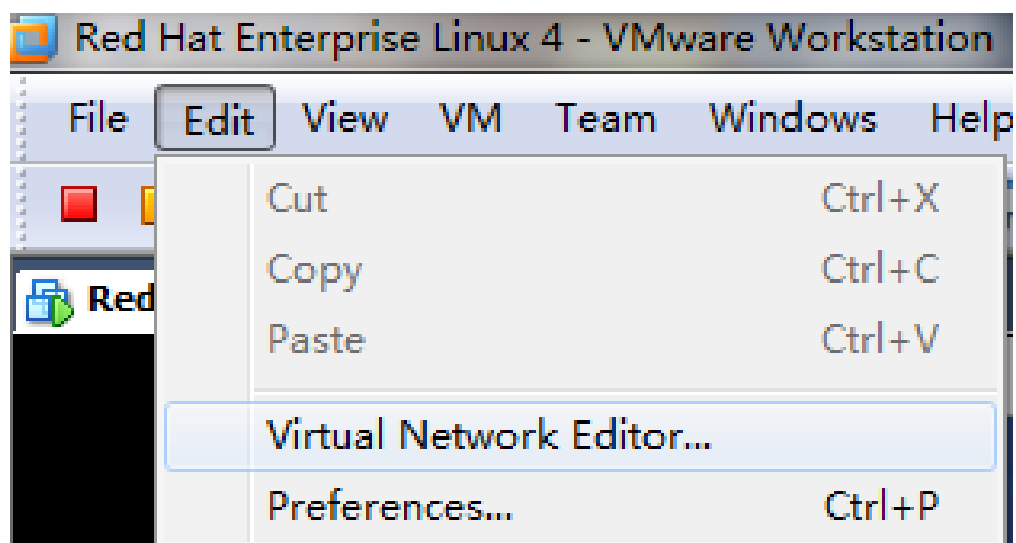


待解压缩完成后，继续在命令窗口输入如下图所示命令：

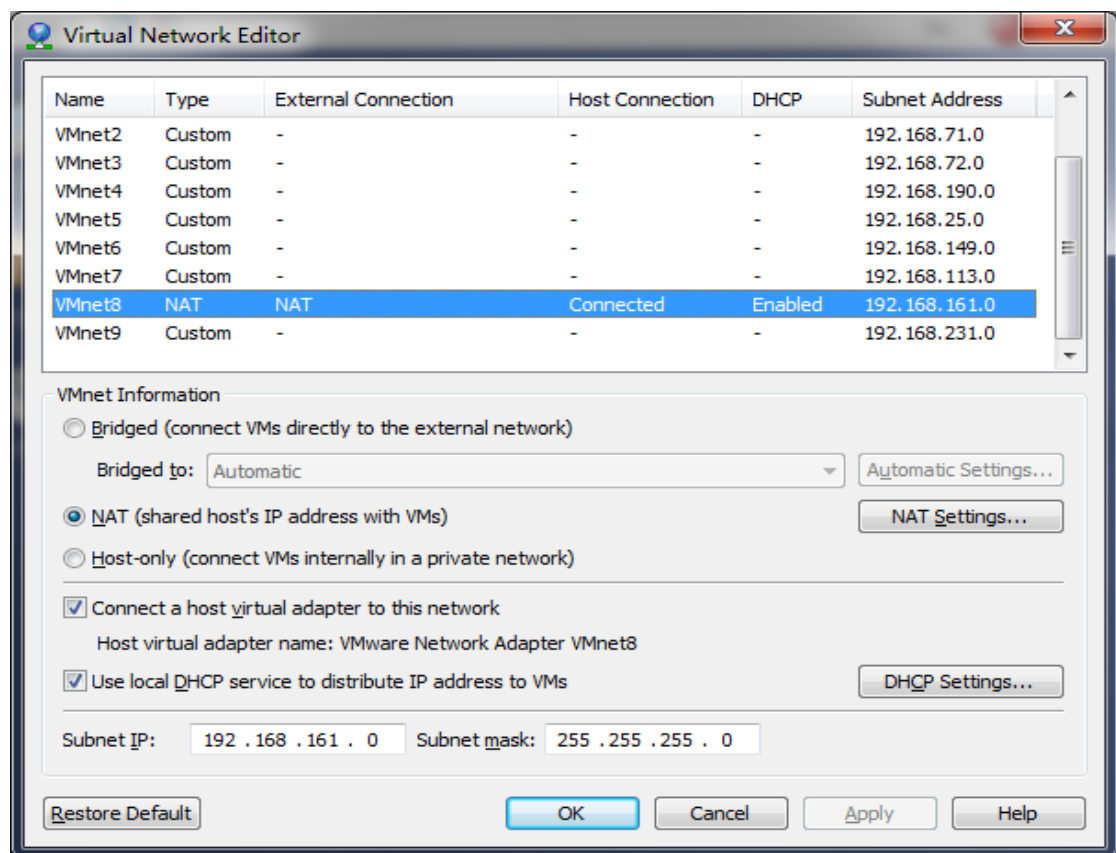


回车后，开始安装 VMware 提供的工具，弹出的所有确认直接回车即可。等待安装完成，至此 Linux 的配置暂时告一段落，下面我们回到虚拟机。

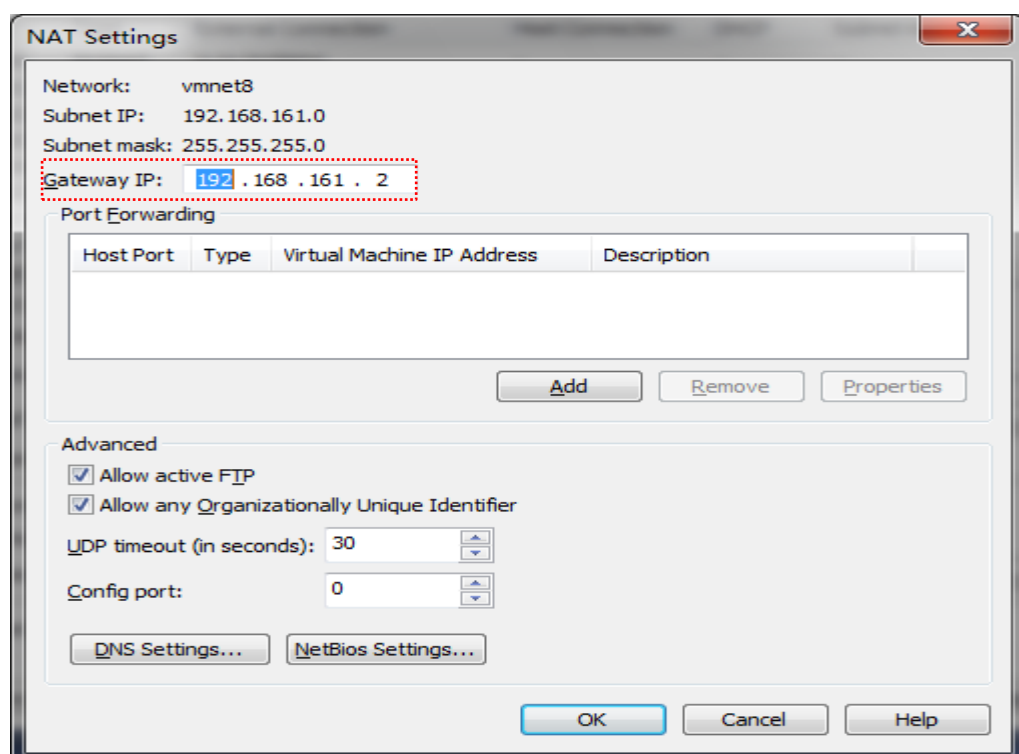
VMware 为我们提供了两块虚拟网卡：分别是 VMware Network Adapter VMnet1 和 VMware Network Adapter VMnet8，这两块虚拟网卡一块采用 Bridge 方式一块采用 NAT 方式，我们要用 VMnet8 这块网卡，所以先要把 VMnet1 禁用掉，然后配置虚拟机，具体步骤如下：



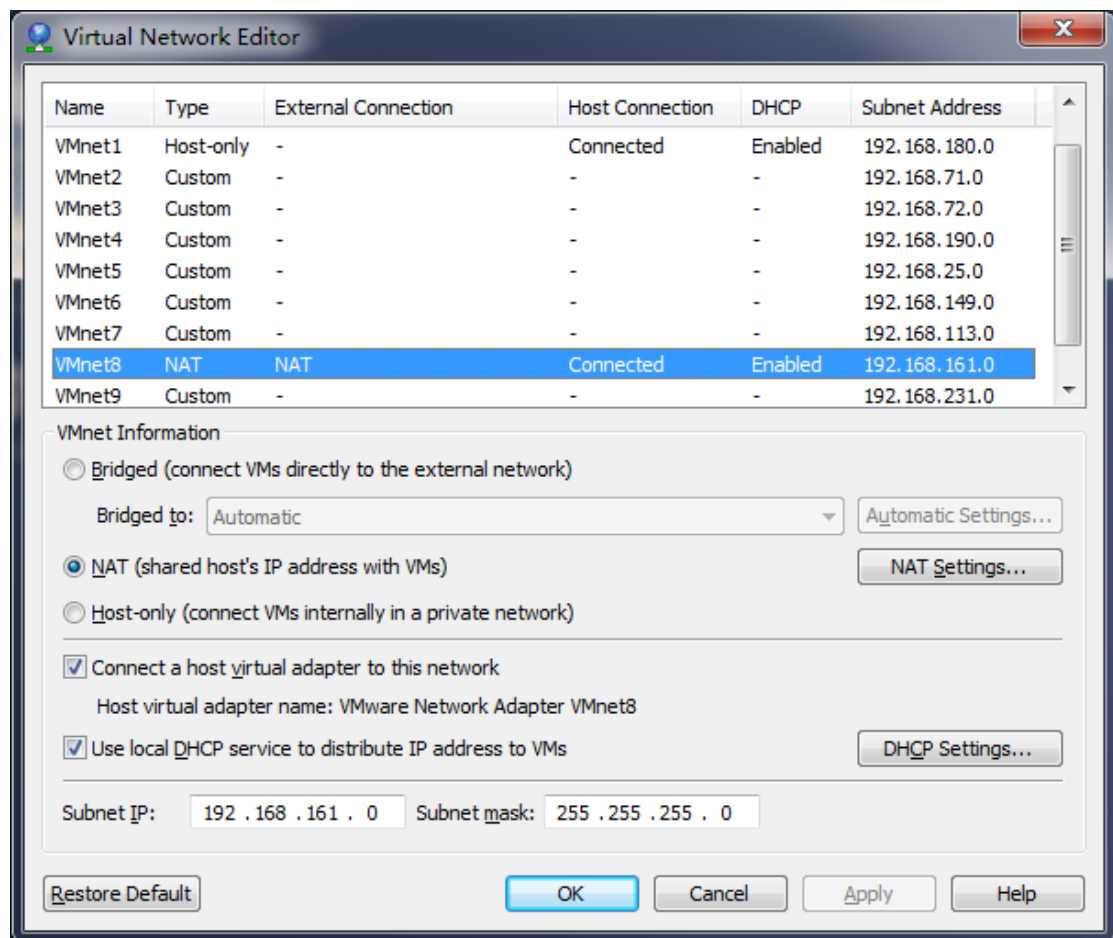
选择 Edit→Virtual Network Editor...后，弹出如下界面：



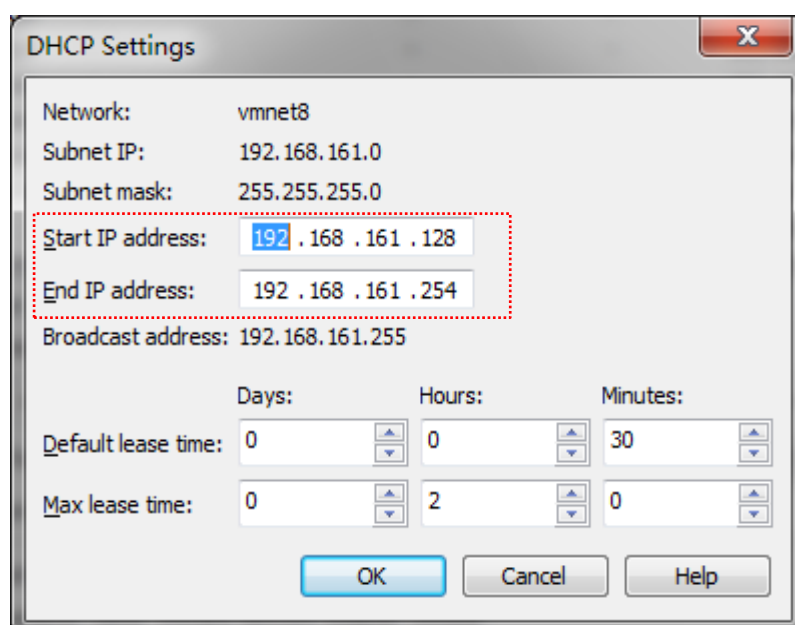
我们一定要选中 VMnet8，然后点击“NAT Settings...”按钮，弹出如下界面：



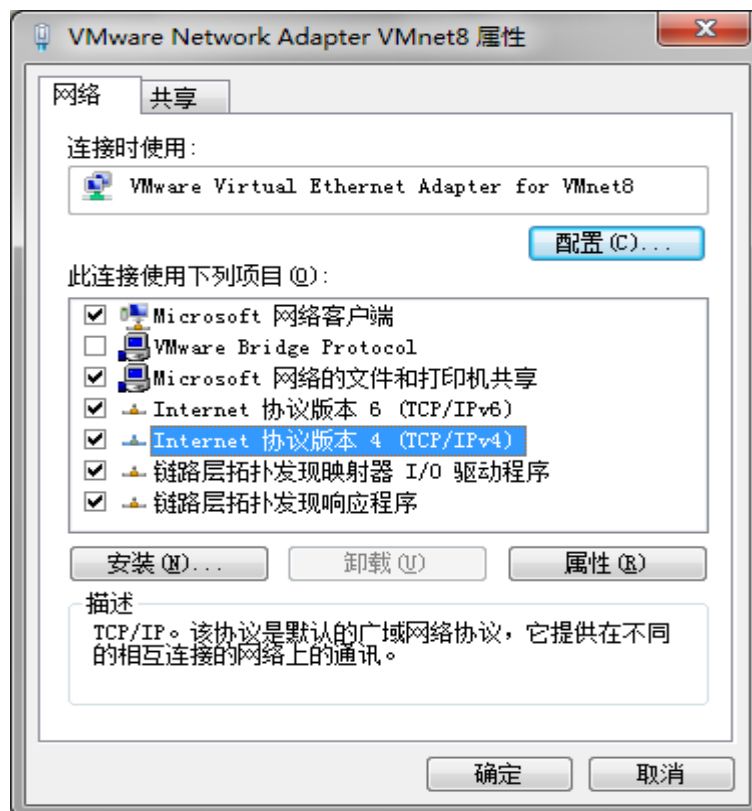
记录下红色框框住的 Gateway IP，一定要记住这个地址，一会儿我们需要用它来配置我们的虚拟网卡的默认网关。记录后，点击 OK。



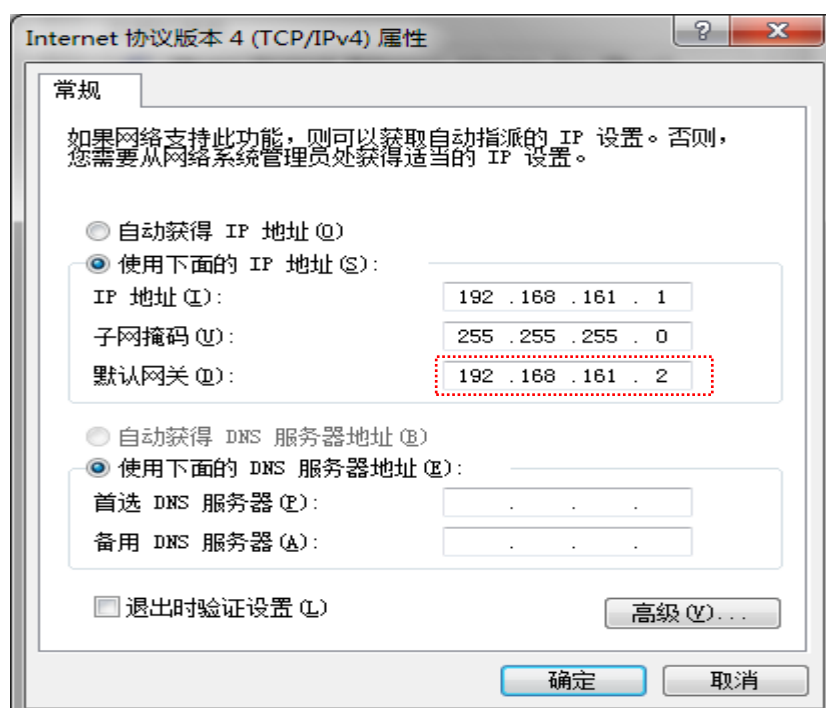
再点击“DHCP Settings...”按钮，弹出如下界面：



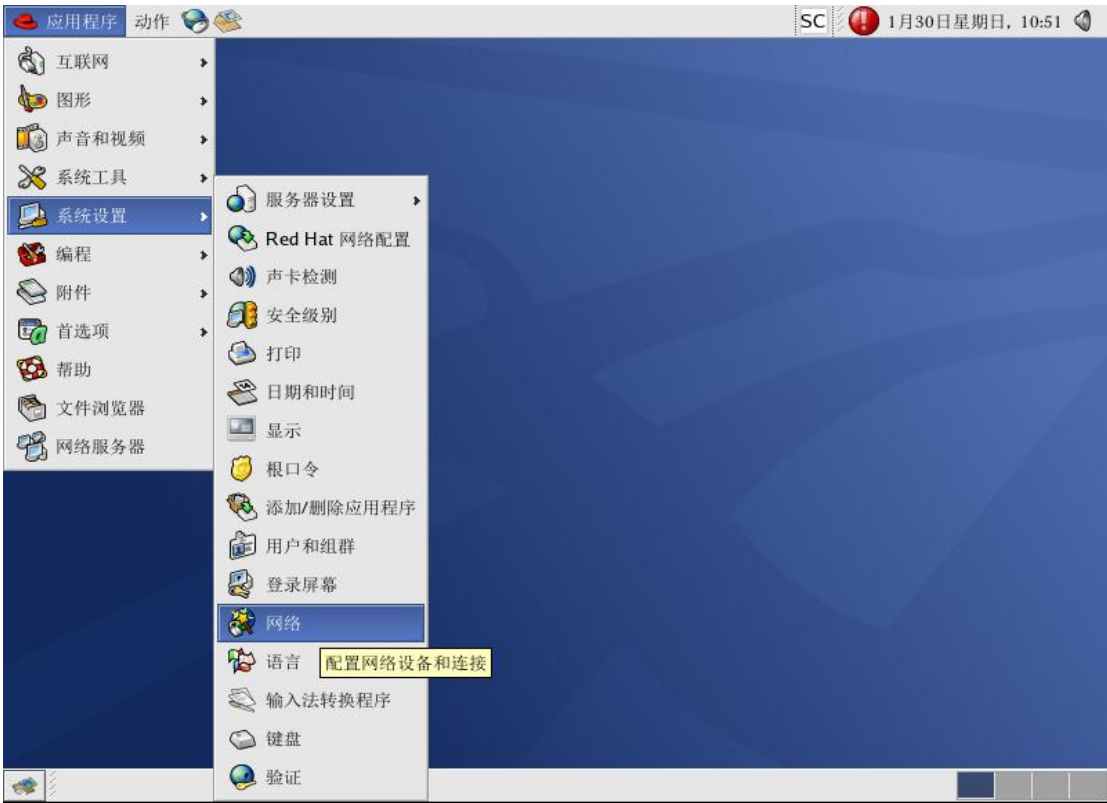
出于个人隐私，故将 DNS 的 IP 地址隐去，各位请自行记下自己的 DNS 地址，记录下后，我们就要对虚拟网卡进行配置了。找到 VMware Network Adapter VMnet8，在上面单击右键选择属性。



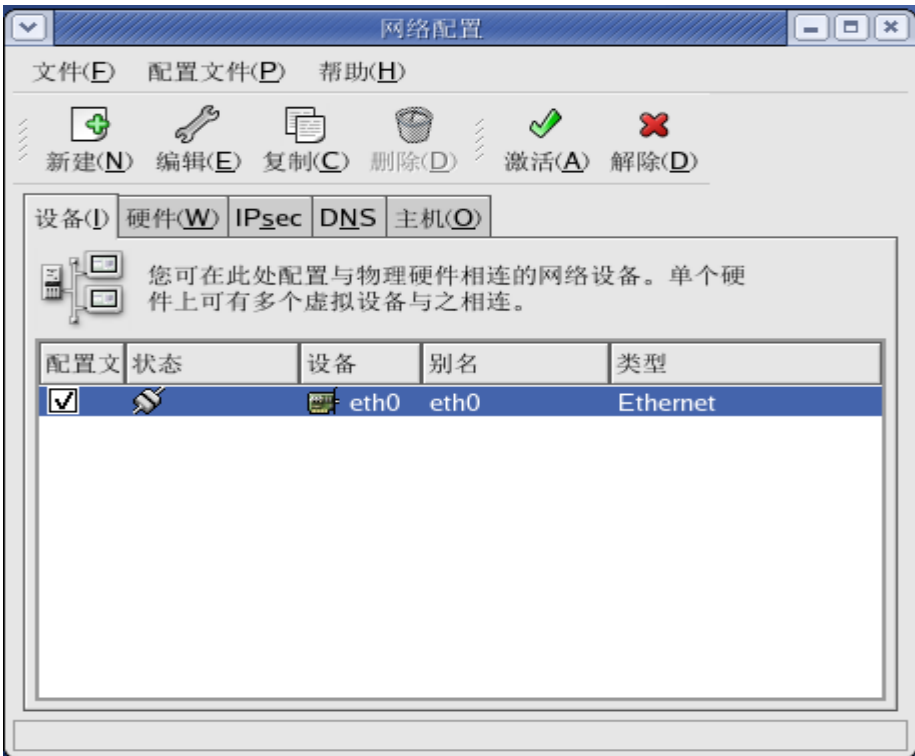
双击 Internet 协议版本 4，出现如下界面：



只需在默认网关处填写刚刚虚拟机为我们提供的 VMnet8 的默认网关地址即可，点击“确定”按钮。至此，宿主机的配置告一段落，下面回到 Linux 操作系统，按照如下步骤操作：



选择应用程序→系统设置→网络，弹出如下界面：



点击“编辑”按钮，弹出如下界面：

以太网设备

常规(G) 路由(R) 硬件设备(H)

别名(N): eth0

☒ 当计算机启动时激活设备(A)

☐ 允许所有用户启用和禁用该设备(U)

☐ 为此界面启用IPv6配置

☐ 自动获取 IP 地址设置使用: dhcp

DHCP 设置

主机名(可选)(H):

☒ 自动从提供商处获取 DNS 信息

☒ 静态设置的 IP 地址:

手工设置 IP 地址

地址(A): 192.168.161.130

子网掩码(S): 255.255.255.0

默认网关(a)地址: 192.168.161.2

确定(O) 取消(C)

Linux 默认采用的是自动获取 IP 地址设置使用，我们要进行手工配置，根据前面记录下的内容填写红色框标上的内容，填写完毕后，点击“确定”按钮。

网络配置

文件(E) 配置文件(P) 帮助(H)

新建(N) 编辑(E) 复制(C) 删除(D) 激活(A) 解除(D)

设备(I) 硬件(W) IPsec DNS 主机(O)

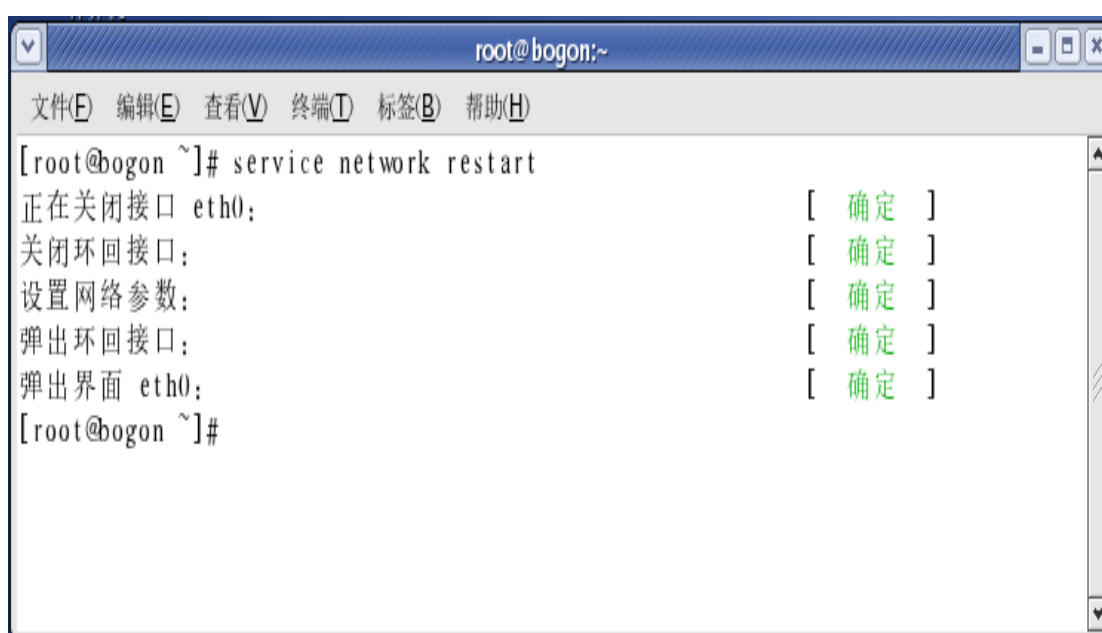
您可在此处配置与物理硬件相连的网络设备。单个硬件上可有多个虚拟设备与之相连。

配置文	状态	设备	别名	类型
<input checked="" type="checkbox"/>	up	eth0	eth0	Ethernet

点击“DNS”选项卡，弹出如下界面：



只需在主 DNS 和第二 DNS 处填写刚才在宿主机看到的 DNS 地址即可，配置完成后，点击“X”关闭按钮，弹出确认对话框，点击确认。此时我们的 Linux 网络并没有真正的生效，启动一个命令窗口，输入如下命令：



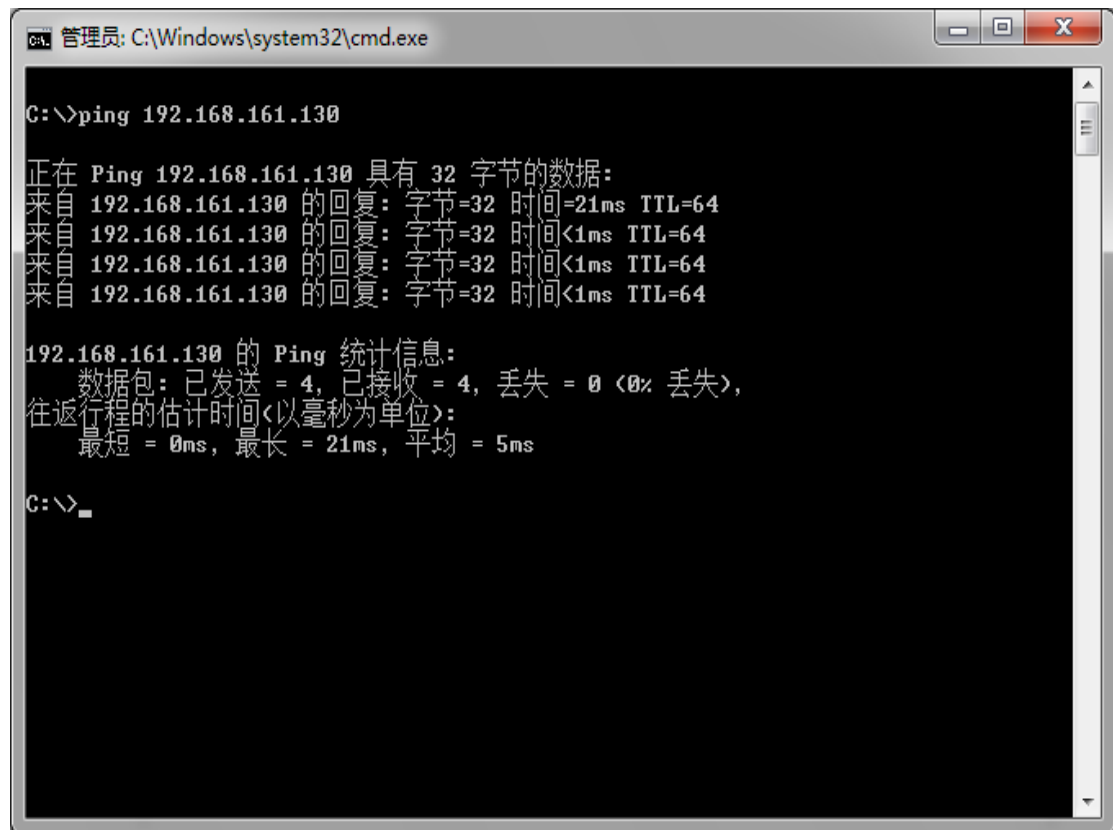
当看到全部为绿色的“确定”的时候，证明我们的网络已经配置成功，下面我们实验一下，在命令窗口输入 `ping www.baidu.com`，看看能否正常 ping 通。

```
root@bogon:~  
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@bogon ~]# ping www.baidu.com  
PING www.a.shifen.com (61.135.169.125) 56(84) bytes of data:  
64 bytes from 61.135.169.125: icmp_seq=0 ttl=128 time=30.8 ms  
64 bytes from 61.135.169.125: icmp_seq=1 ttl=128 time=31.6 ms  
64 bytes from 61.135.169.125: icmp_seq=2 ttl=128 time=30.3 ms  
  
--- www.a.shifen.com ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2001ms  
rtt min/avg/max/mdev = 30.301/30.946/31.644/0.585 ms, pipe 2  
[root@bogon ~]#
```

OK，可以正常 ping 通，下面打开一个 Linux 内置的 Firefox 浏览器，看看能否正常打开网页。



下面我们在看一下在宿主机下能否正常 ping 通 Linux 系统 IP，启动 dos 窗口，输入如下命令：



```
C:\Windows\system32\cmd.exe

C:\>ping 192.168.161.130

正在 Ping 192.168.161.130 具有 32 字节的数据:
来自 192.168.161.130 的回复: 字节=32 时间=21ms TTL=64
来自 192.168.161.130 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.161.130 的回复: 字节=32 时间<1ms TTL=64
来自 192.168.161.130 的回复: 字节=32 时间<1ms TTL=64

192.168.161.130 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 0ms, 最长 = 21ms, 平均 = 5ms

C:\>
```

可以看到我们的 Linux 网络已经配置成功，而且宿主机已经和 Linux 之间可以进行正常的通信了，网络配置是我纠结了好久的问题，经过不懈努力终于搞定了，有好的东西当然不能自己独享，必须拿出来和各位同仁共享一下。在下面我将讲解定制 Linux 环境。

定制 LINUX 操作系统

我们知道，Linux 是一个基于 Unix 的操作系统，Unix 是一个基于命令行的操作系统，所以我们在使用 Linux 的时候不会直接使用 XWindow 窗口，都是基于命令行方式使用，在本阶段我主要讲解如何定制 Linux。

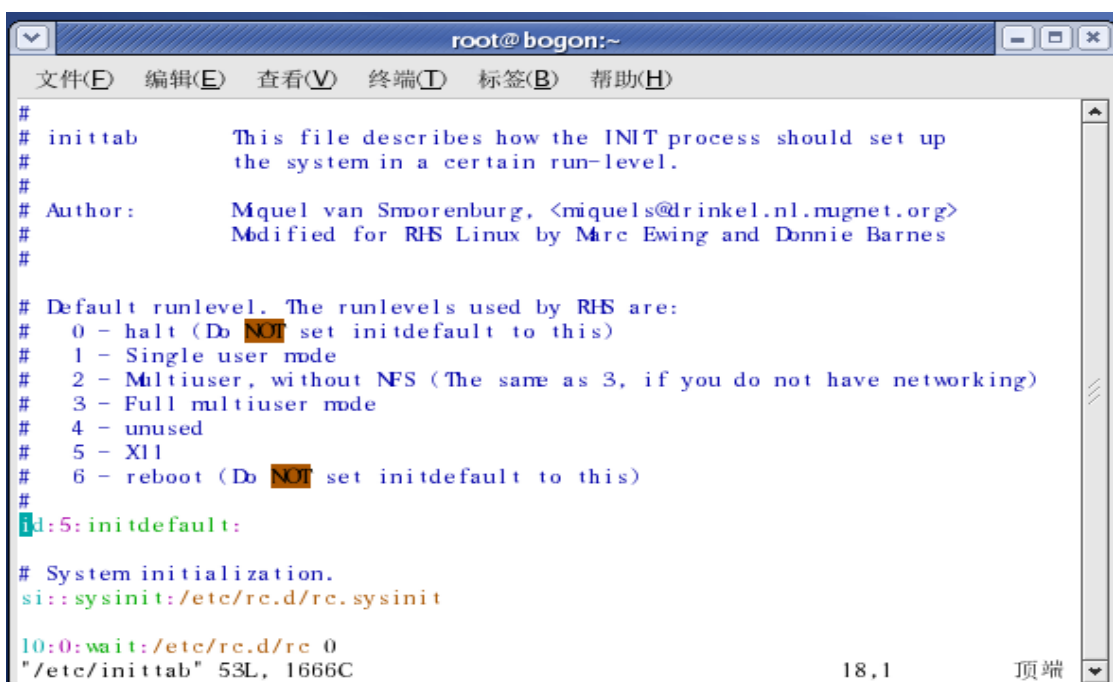
✚ 修改 Linux 系统启动方式

打开一个 Linux 终端，输入如下命令：



```
root@bogon:~  
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
[root@bogon ~]# vi /etc/inittab
```

回车后，显示如下界面：



```
root@bogon:~  
文件(E) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)  
#  
# inittab          This file describes how the INIT process should set up  
#                  the system in a certain run-level.  
#  
# Author:         Mquiel van Smoorenburg, <miquels@drinkel.nl.nugnet.org>  
#                 Modified for RHEL Linux by Marc Ewing and Donnie Barnes  
#  
# Default runlevel. The runlevels used by RHEL are:  
#  0 - halt (Do NOT set initdefault to this)  
#  1 - Single user mode  
#  2 - Multiuser, without NFS (The same as 3, if you do not have networking)  
#  3 - Full multiuser mode  
#  4 - unused  
#  5 - X11  
#  6 - reboot (Do NOT set initdefault to this)  
#  
id:5:initdefault:  
# System initialization.  
si::sysinit:/etc/rc.d/rc.sysinit  
10:0:wait:/etc/rc.d/rc 0  
"/etc/inittab" 53L, 1666C  
18,1 顶端
```

我们看到，默认的启动方式值为 5，我们需要将其改为 3，具体步骤如下：

1: 输入键盘的“i”键，进入插入模式，将如下内容修改为：

```
# Default runlevel. The runlevels used by RHS are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:5:initdefault:
```

```
# Default runlevel. The runlevels used by RHS are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:
```

2: 修改完后，按“Esc”键，然后按“Shift+:”组合键，界面如下：

```
# Default runlevel. The runlevels used by RHS are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:3:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

10:0:wait:/etc/rc.d/rc 0
:
```

3: 在:后输入 wq，保存退出

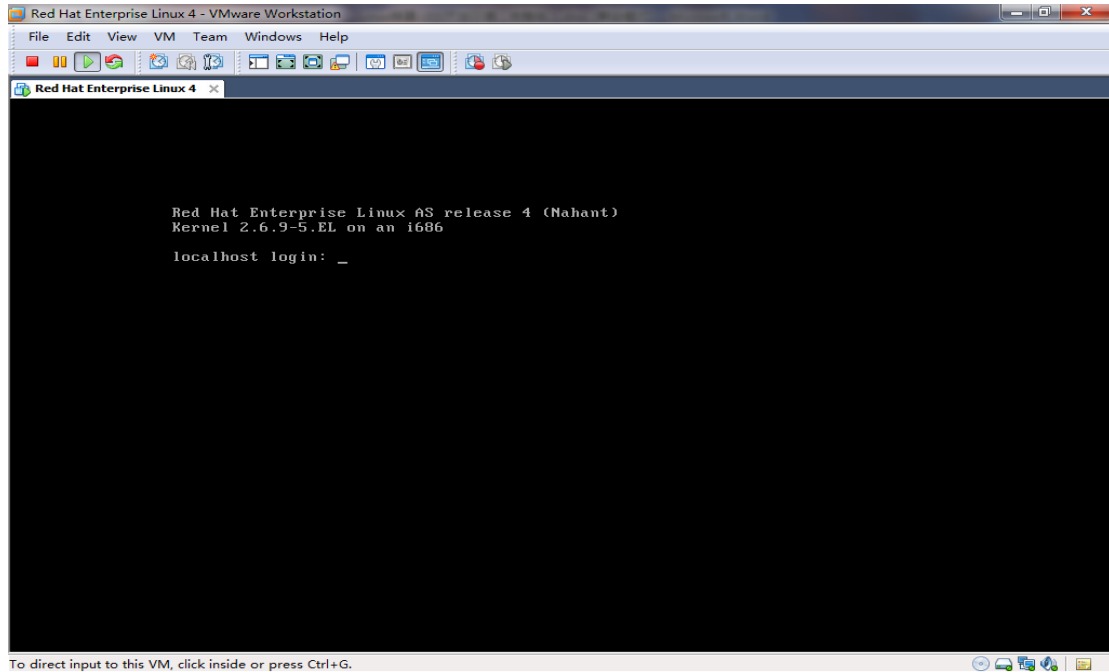
```
id:3:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

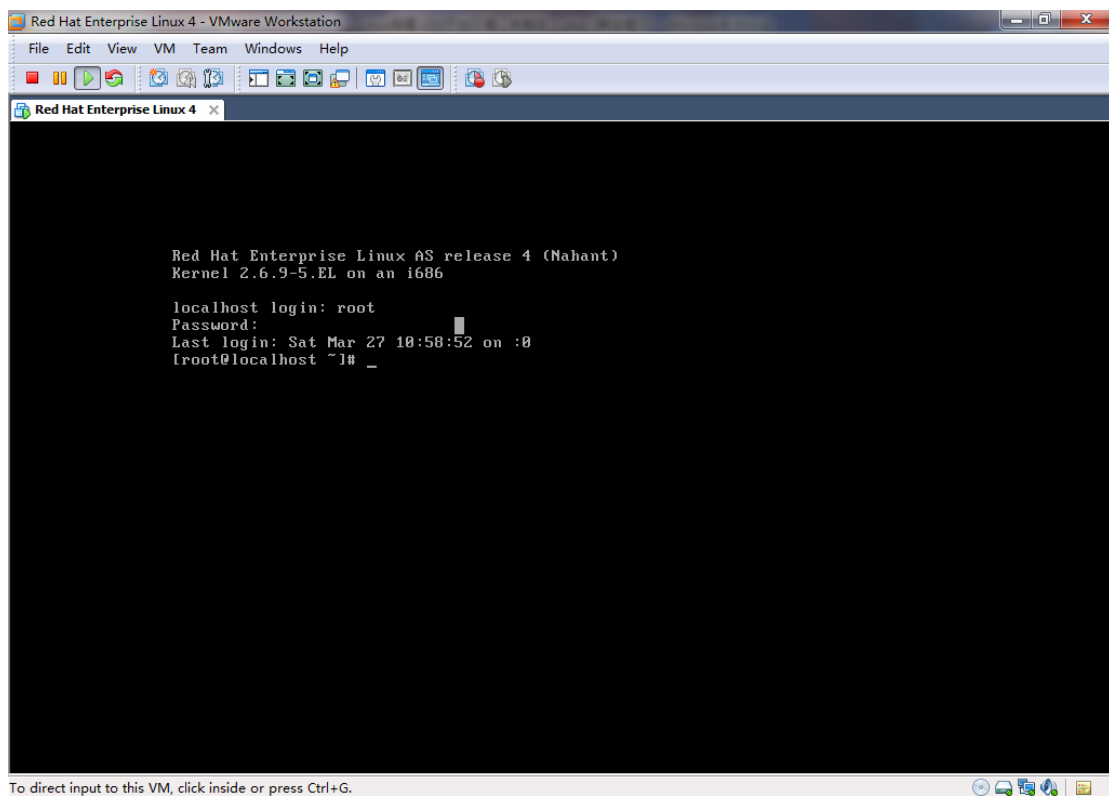
10:0:wait:/etc/rc.d/rc 0
:wq
```


4: 修改完后, 输入 `reboot`, 重启 Linux 系统。

启动后的 Linux 界面如下:



输入用户名和密码, 登录 Linux。



🚦 修改 Linux 字符集

1: 在进入 Linux 系统后，输入如下命令：

```
Red Hat Enterprise Linux AS release 4 (Nahant)
Kernel 2.6.9-5.EL on an i686

bogon login: root
Password:
Last login: Sun Jan 30 10:17:36 on tty1
You have mail.
[root@bogon ~]# vi /etc/sysconfig/i18n_
```

2: 同上操作步骤，进入编辑模式，将默认内容修改为如下内容：

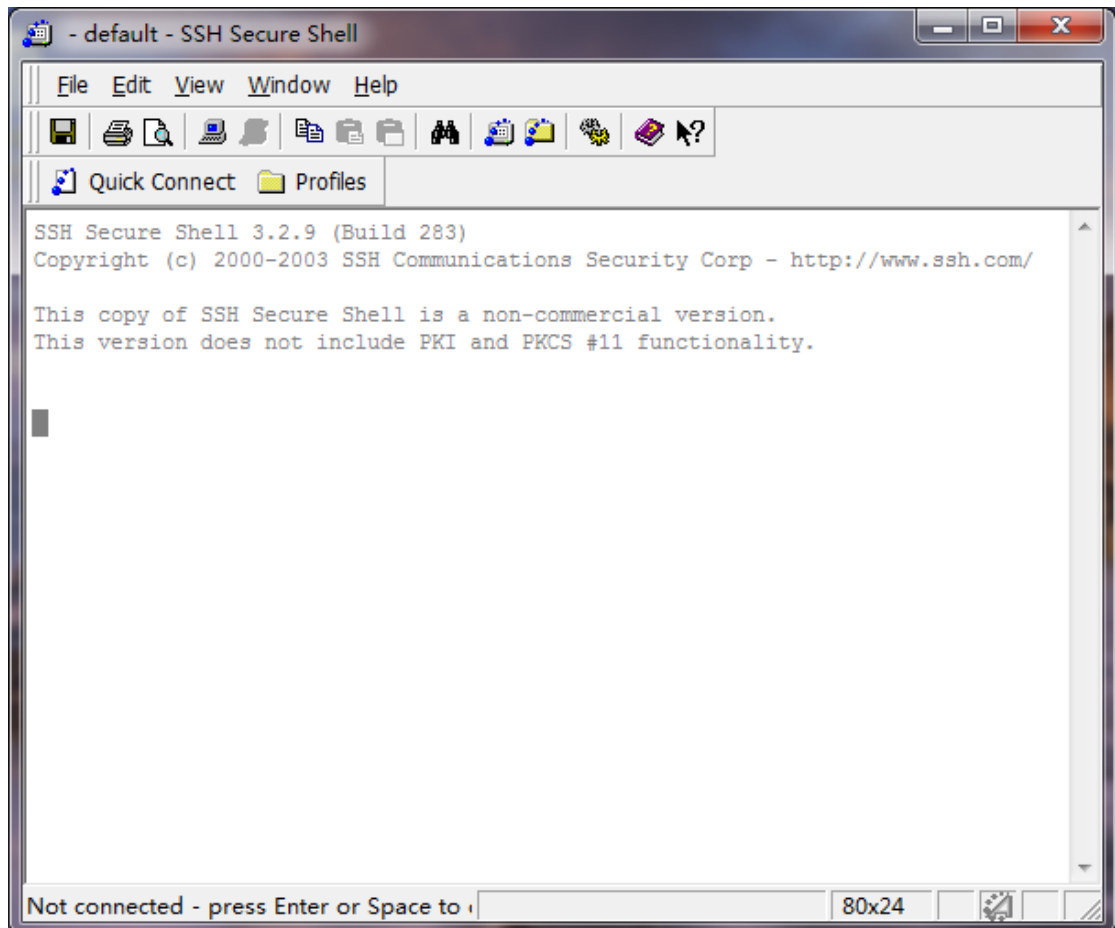
```
LANG="zh_CN.GB18030"
LANGUAGE="zh_CN.GB18030:zh_CN.GB2312:zh_CN"
SUPPORTED="zh_CN.UTF-8:zh_CN:zh:en_US.UTF-8:en_US:en"
SYSFONT="lat1-sun16"
```

3: 修改完成后，保存退出，输入 reboot 重新引导 Linux 启动。

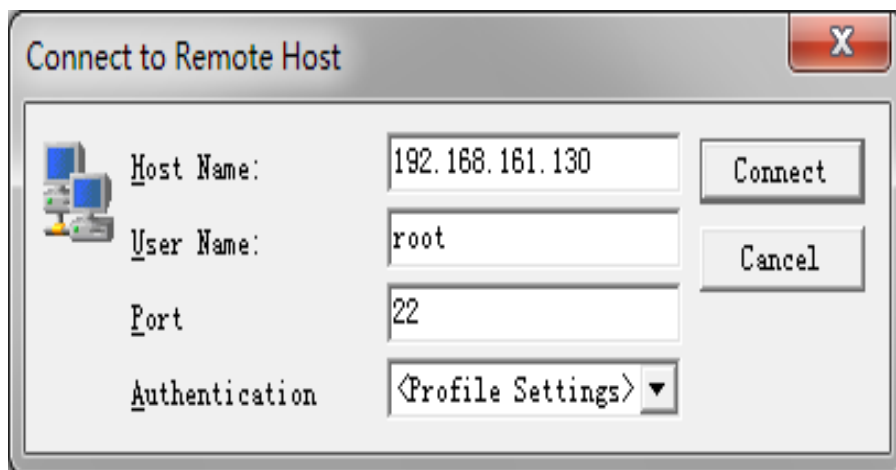
至此，Linux 操作系统的定制工作完成，在下面的章节中将主要讲解软件的安装，个人认为最重要的是 Oracle 数据库的安装，下面我将逐一讲解。

安装 ORACLE 数据库

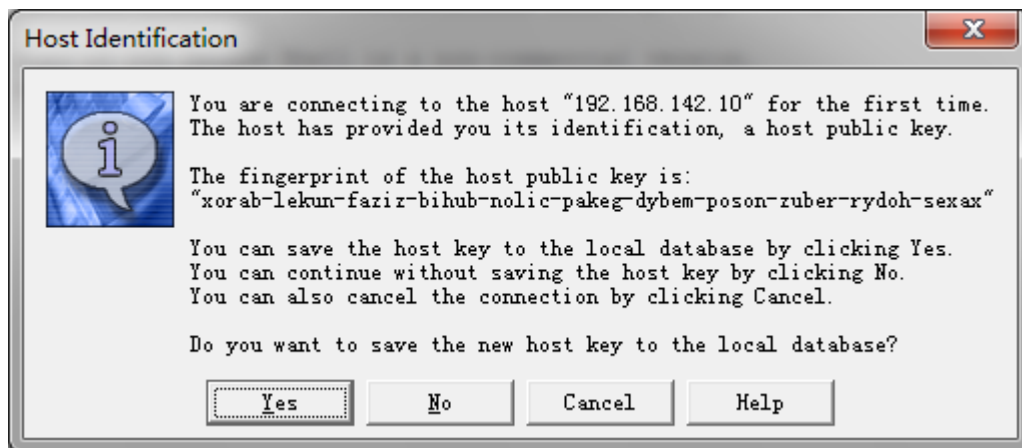
1、启动 SSH 客户端软件，界面如下：



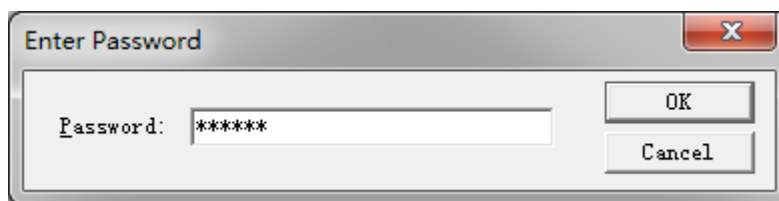
点击“Quick Connect”按钮，出现如下界面：



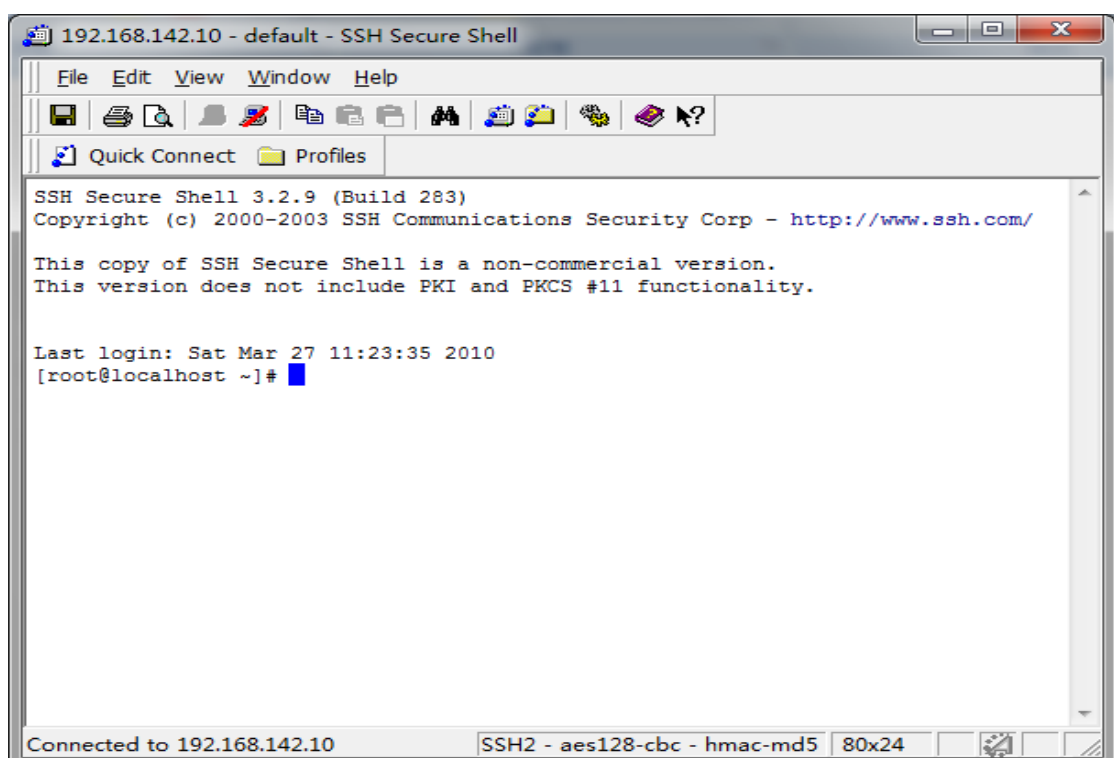
输入刚才配置好的 Linux 主机 IP 和用户名，点击 Connect 按钮，如果连接成功出现如下界面：



点击 “Yes” 按钮即可，出现如下界面：

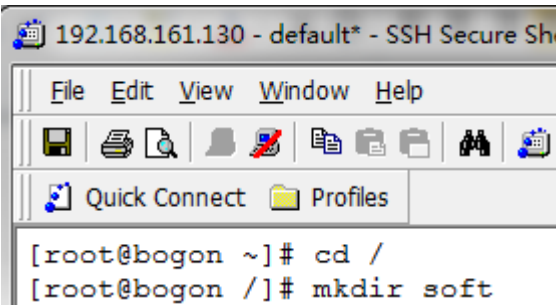


输入 root 用户的密码，点击 OK 按钮，如果验证通过出现如下界面：



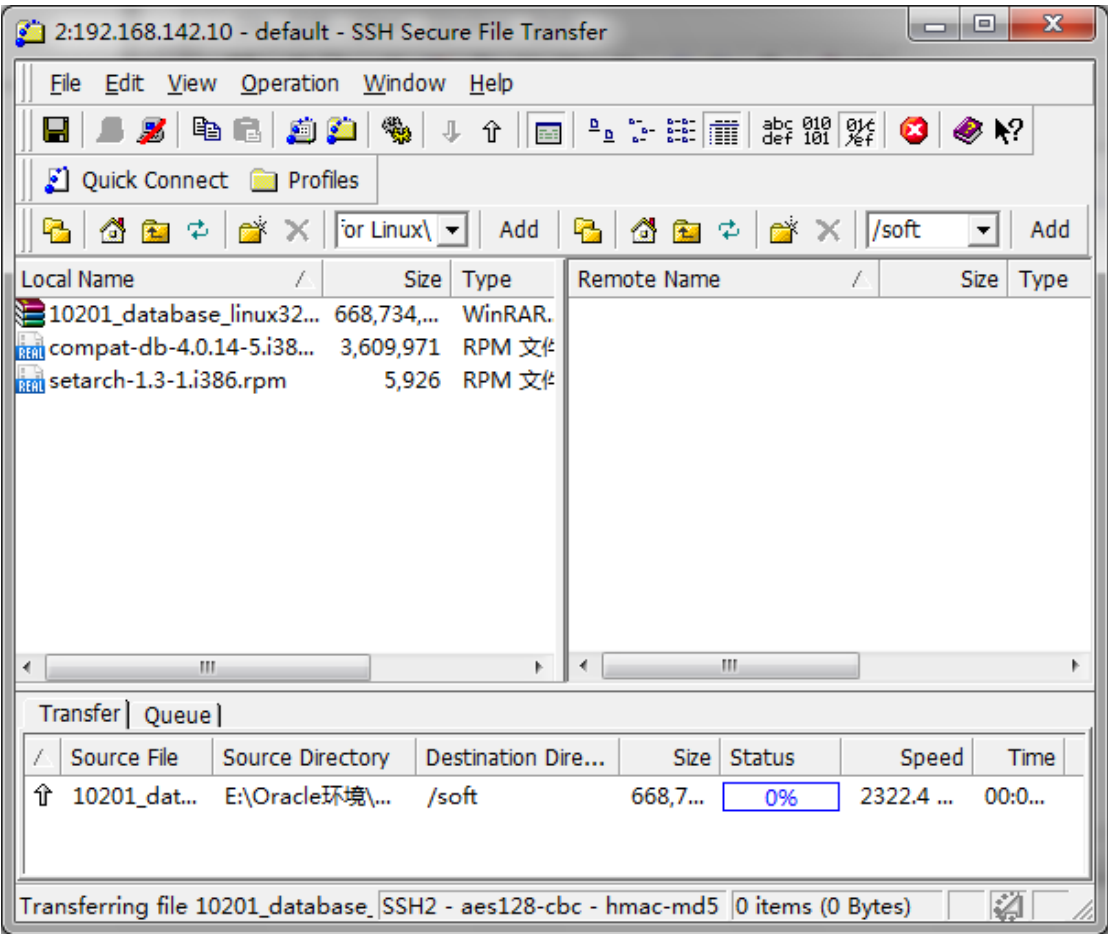
2、将所需文件上传到 Linux 系统

(1) 首先在 SSH 中键入如下 Shell 脚本



```
192.168.161.130 - default* - SSH Secure Shell  
File Edit View Window Help  
Quick Connect Profiles  
[root@bogon ~]# cd /  
[root@bogon /]# mkdir soft
```

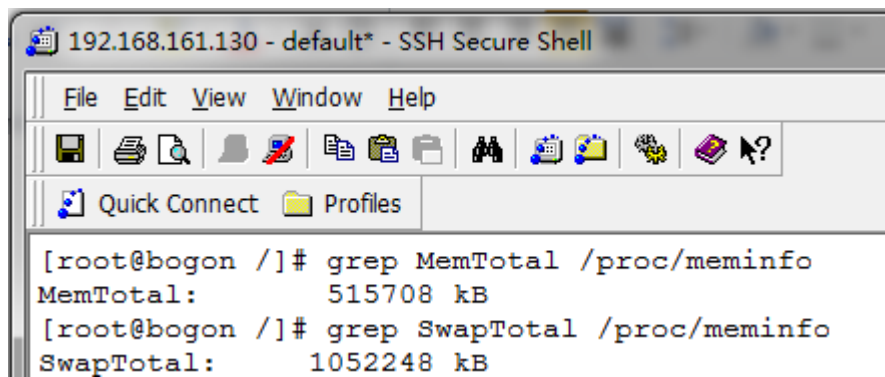
(2) 点击 SSH 客户端软件的  按钮，出现如下界面：



将要上传的文件从左侧窗口拖动到右侧即可。

3、针对 Oracle 数据库配置 Linux 系统，以下操作全部采用 root 用户

(1) 检查内存和交换分区是否满足需求



The screenshot shows a terminal window titled "192.168.161.130 - default* - SSH Secure Shell". The window has a menu bar with "File", "Edit", "View", "Window", and "Help". Below the menu bar is a toolbar with various icons. The terminal content shows the following commands and output:

```
[root@bogon /]# grep MemTotal /proc/meminfo
MemTotal:      515708 kB
[root@bogon /]# grep SwapTotal /proc/meminfo
SwapTotal:     1052248 kB
```

(2) 确认 Linux 系统中已经安装了如下组件

binutils-2.15.92.0.2-13.EL4

compat-db-4.1.25-9

compat-libstdc++-296-2.96-132.7.2

control-center-2.8.0-12

gcc-3.4.3-22.1.EL4

gcc-c++-3.4.3-22.1.EL44

glibc-2.3.4-2.9

glibc-common-2.3.4-2.9

gnome-libs-1.4.1.2.90-44.1

libstdc++-3.4.3-22.1

libstdc++-devel-3.4.3-22.1

make-3.80-5

pdksh-5.2.14-30

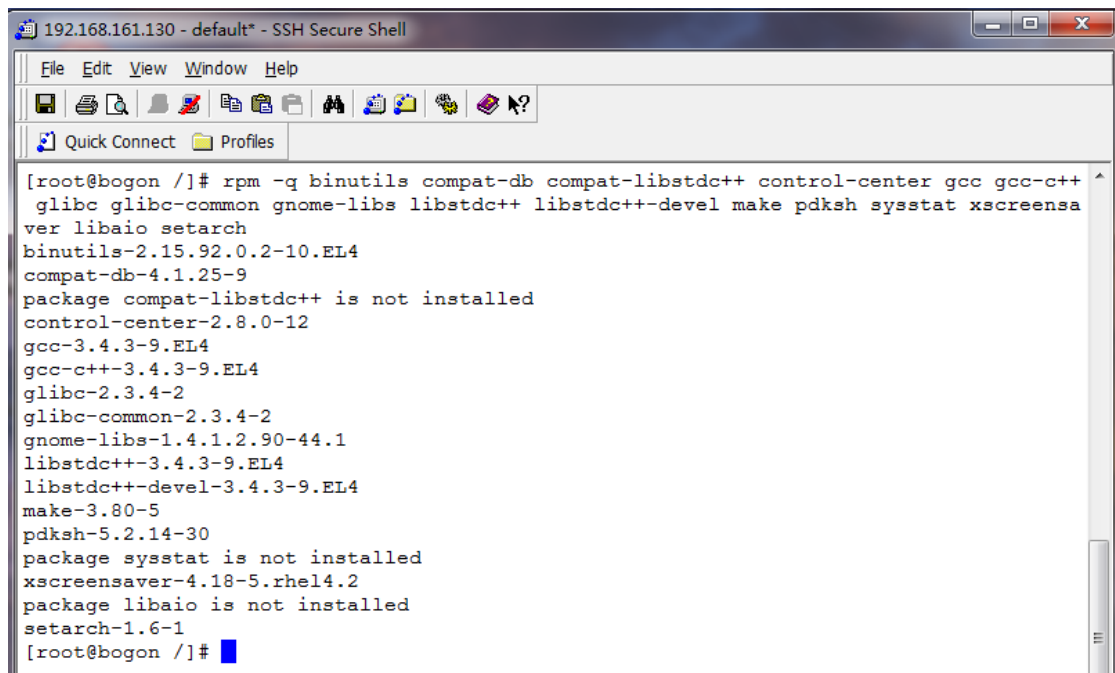
sysstat-5.0.5-1

xscreensaver-4.18-5.rhel4.2

libaio-0.3.96

setarch-1.6-1

(3) 检测软件包

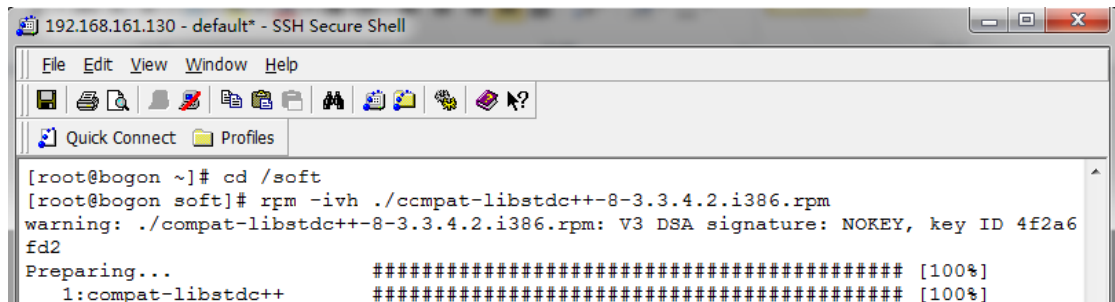


```
192.168.161.130 - default* - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon /]# rpm -q binutils compat-db compat-libstdc++ control-center gcc gcc-c++
glibc glibc-common gnome-libs libstdc++ libstdc++-devel make pdksh sysstat xscreensa
ver libaio setarch
binutils-2.15.92.0.2-10.EL4
compat-db-4.1.25-9
package compat-libstdc++ is not installed
control-center-2.8.0-12
gcc-3.4.3-9.EL4
gcc-c++-3.4.3-9.EL4
glibc-2.3.4-2
glibc-common-2.3.4-2
gnome-libs-1.4.1.2.90-44.1
libstdc++-3.4.3-9.EL4
libstdc++-devel-3.4.3-9.EL4
make-3.80-5
pdksh-5.2.14-30
package sysstat is not installed
xscreensaver-4.18-5.rhel4.2
package libaio is not installed
setarch-1.6-1
[root@bogon /]#
```

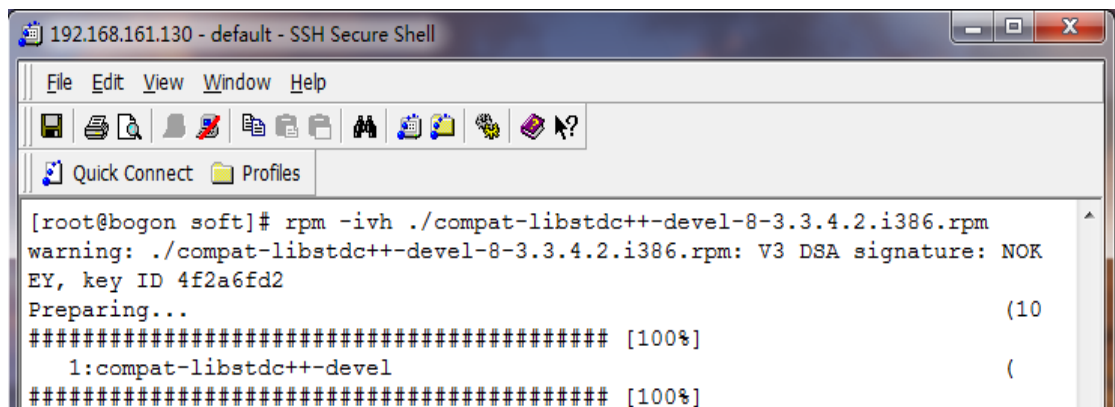
经检测我缺少 compat-libstdc++、sysstat 和 libaio-0.3.96 没有安装，将安装文件上传到/soft 目录下。

(4) 安装缺失软件



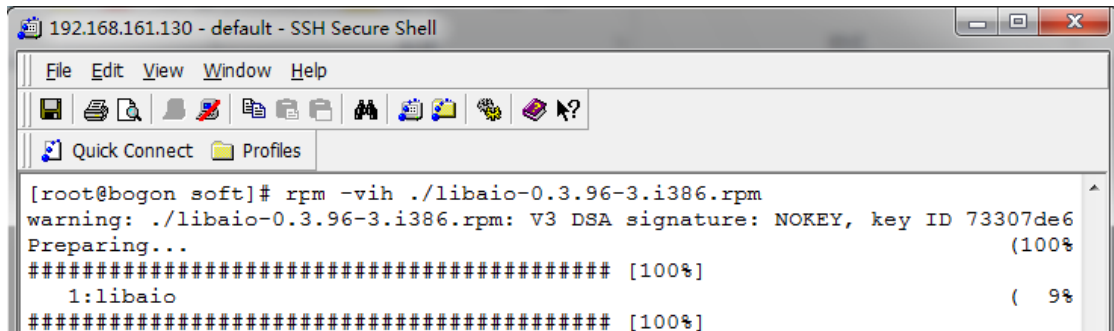
```
192.168.161.130 - default* - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon ~]# cd /soft
[root@bogon soft]# rpm -ivh ./compat-libstdc++-8-3.3.4.2.i386.rpm
warning: ./compat-libstdc++-8-3.3.4.2.i386.rpm: V3 DSA signature: NOKEY, key ID 4f2a6
fd2
Preparing... ##### [100%]
1:compat-libstdc++ ##### [100%]
```



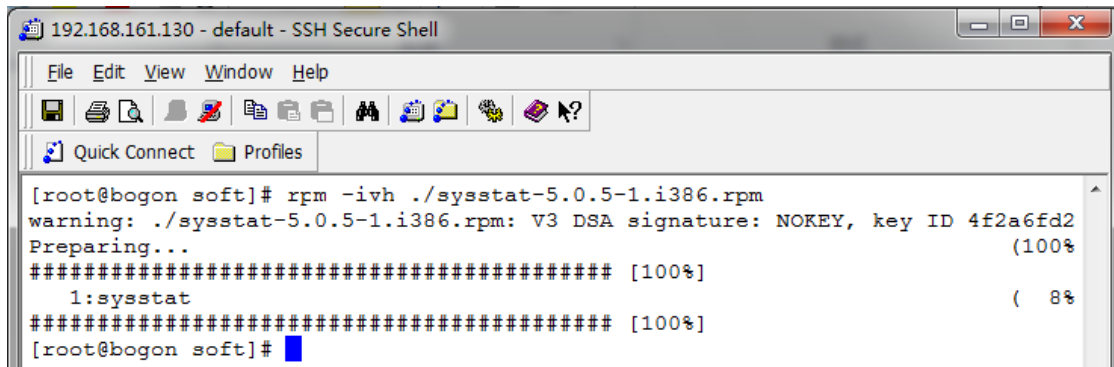
```
192.168.161.130 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon soft]# rpm -ivh ./compat-libstdc++-devel-8-3.3.4.2.i386.rpm
warning: ./compat-libstdc++-devel-8-3.3.4.2.i386.rpm: V3 DSA signature: NOKE
EY, key ID 4f2a6fd2
Preparing... (10 ##### [100%]
1:compat-libstdc++-devel ( ##### [100%]
```

```
192.168.161.130 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

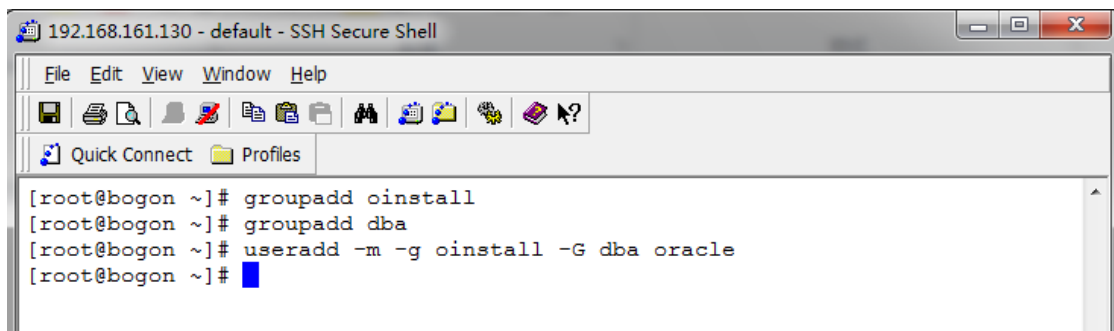
[root@bogon soft]# rpm -vih ./libaio-0.3.96-3.i386.rpm
warning: ./libaio-0.3.96-3.i386.rpm: V3 DSA signature: NOKEY, key ID 73307de6
Preparing... (100%)
##### [100%]
1:libaio ( 9%)
##### [100%]
```



```
192.168.161.130 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon soft]# rpm -ivh ./sysstat-5.0.5-1.i386.rpm
warning: ./sysstat-5.0.5-1.i386.rpm: V3 DSA signature: NOKEY, key ID 4f2a6fd2
Preparing... (100%)
##### [100%]
1:sysstat ( 8%)
##### [100%]
[root@bogon soft]#
```

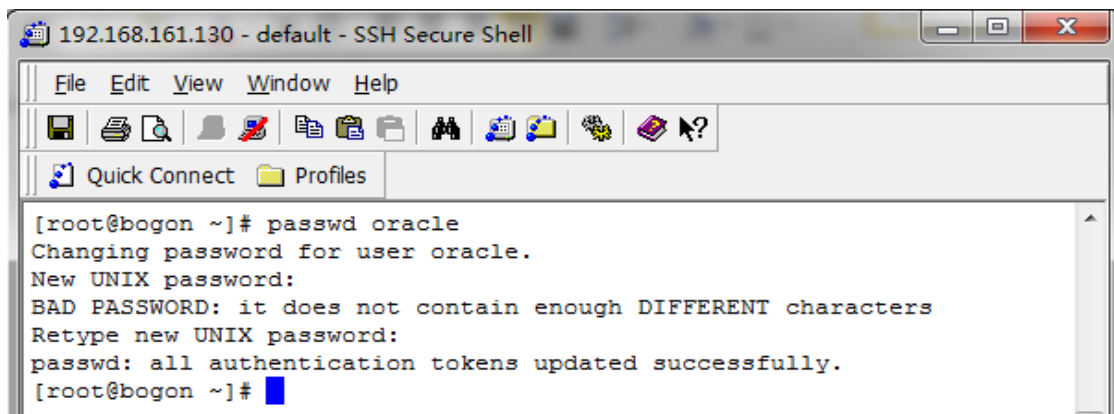
(5) 创建 Oracle 用户和组



```
192.168.161.130 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon ~]# groupadd oinstall
[root@bogon ~]# groupadd dba
[root@bogon ~]# useradd -m -g oinstall -G dba oracle
[root@bogon ~]#
```

(6) 更改 Oracle 用户密码

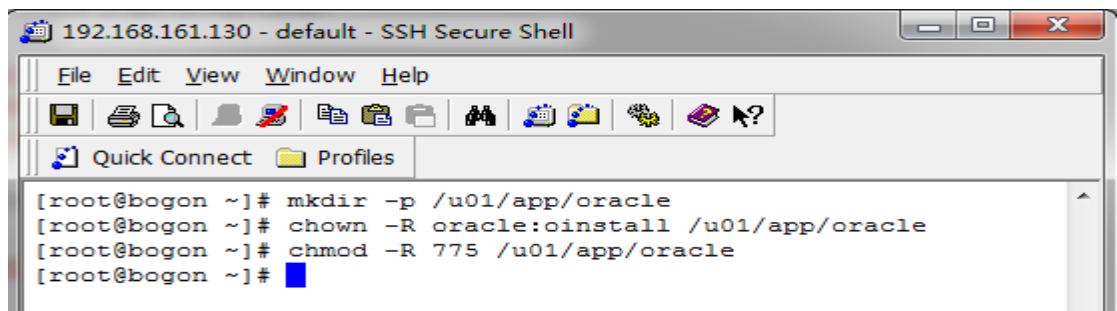


```
192.168.161.130 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon ~]# passwd oracle
Changing password for user oracle.
New UNIX password:
BAD PASSWORD: it does not contain enough DIFFERENT characters
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
[root@bogon ~]#
```

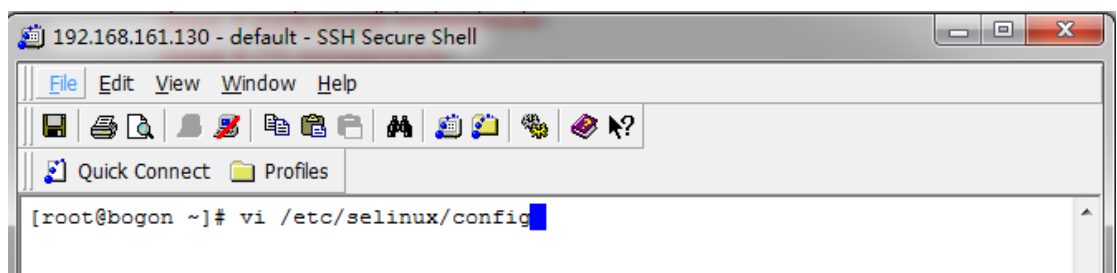
输入两次 Oracle 用户的密码。

(7) 创建 Oracle 安装时的相关目录



```
192.168.161.130 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon ~]# mkdir -p /u01/app/oracle
[root@bogon ~]# chown -R oracle:oinstall /u01/app/oracle
[root@bogon ~]# chmod -R 775 /u01/app/oracle
[root@bogon ~]#
```

(8) 关闭 SELINUX



```
192.168.161.130 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon ~]# vi /etc/selinux/config
```

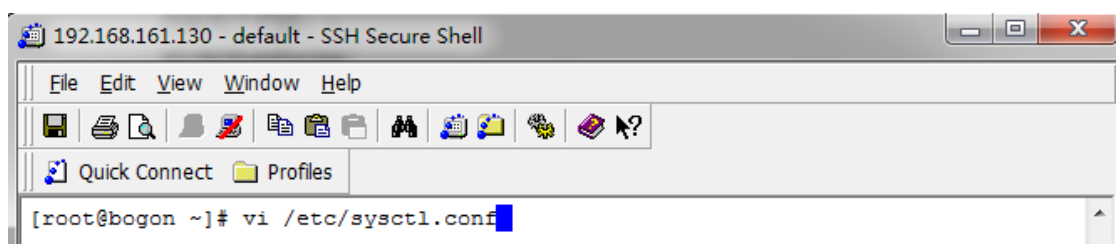
找到如下内容:

SELINUX=enforcing

改为

SELINUX=disabled

(9) 配置 Linux 内核参数

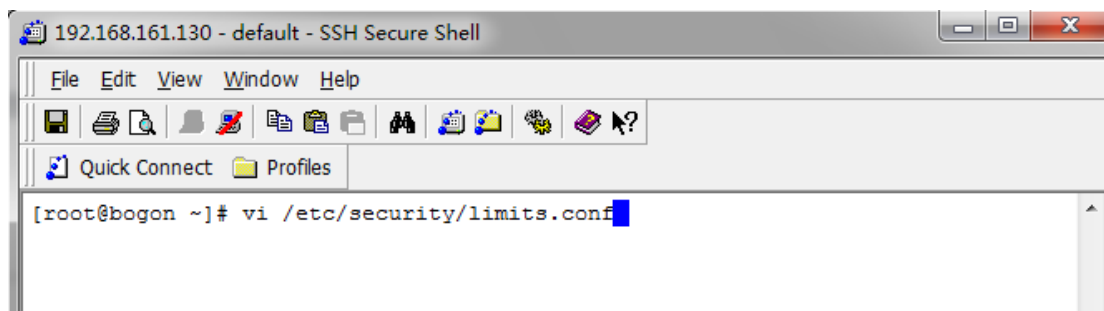


```
192.168.161.130 - default - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon ~]# vi /etc/sysctl.conf
```

在该文件的最末尾添加如下内容:

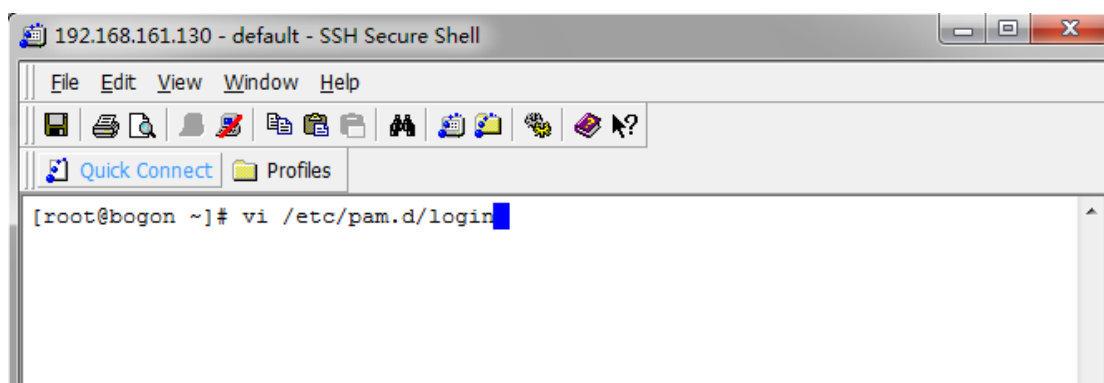
```
kernel.shmall = 2097152
kernel.shmmax = 536870912
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
fs.file-max = 65536
net.ipv4.ip_local_port_range = 1024 65000
net.core.rmem_default=262144
net.core.wmem_default=262144
net.core.rmem_max=262144
net.core.wmem_max=262144
```

(10) 为 Oracle 用户设置 Shell limits



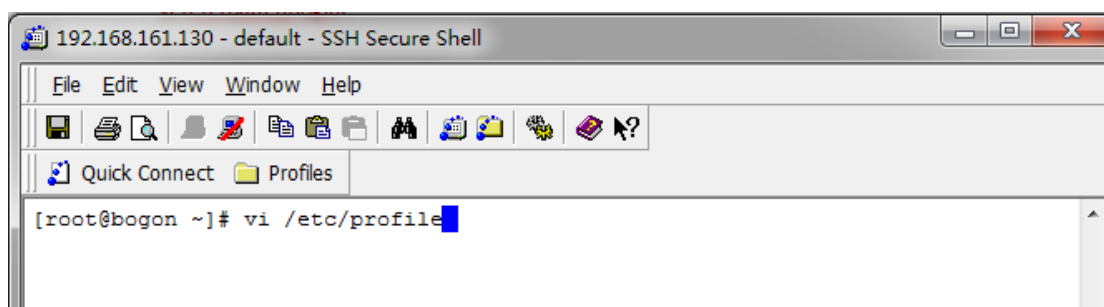
在该文件的最末尾添加如下内容:

```
oracle soft nproc 2047
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
```



在该文件的最末尾添加如下内容:

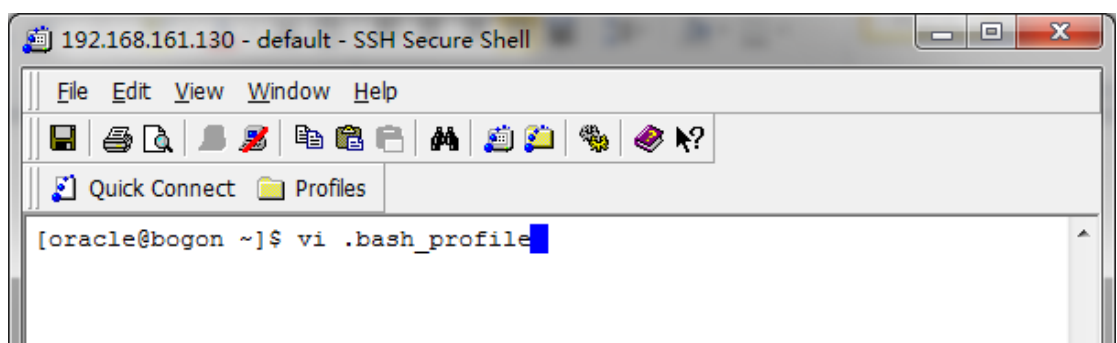
```
session required /lib/security/pam_limits.so
session required pam_limits.so
```



在该文件末尾添加如下内容：

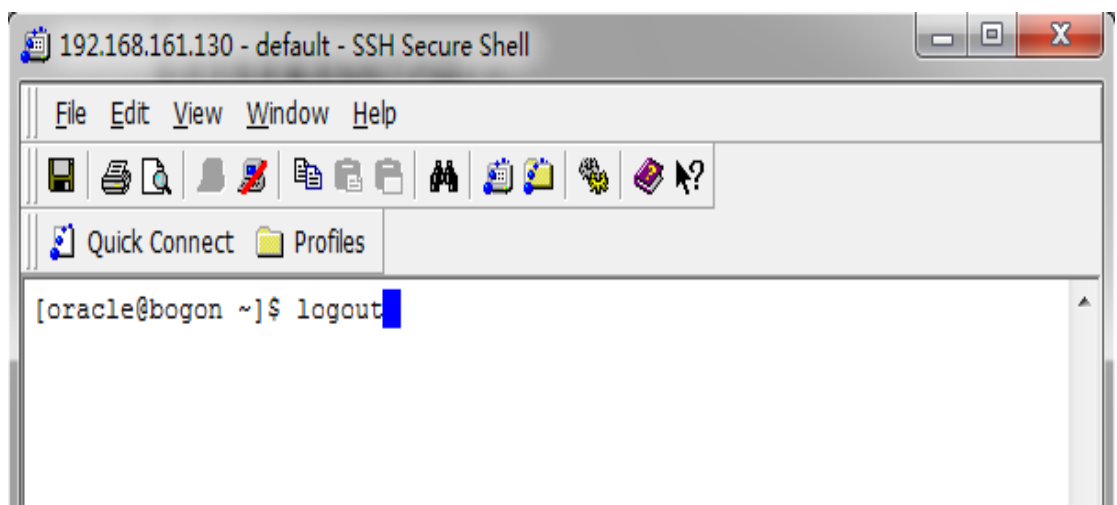
```
if [ $USER = "oracle" ]; then
if [ $SHELL = "/bin/ksh" ]; then
ulimit -p 16384
ulimit -n 65536
else
ulimit -u 16384 -n 65536
fi
fi
```

4、使用 oracle 用户登录 Linux，设置相关内容



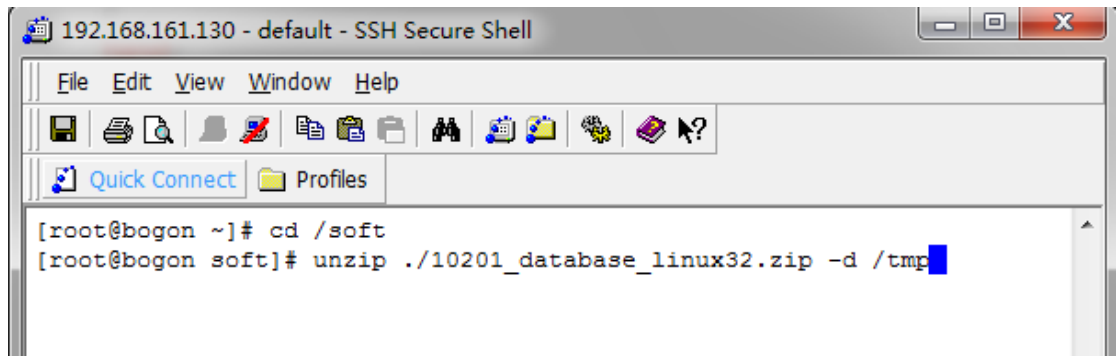
在该文件末尾添加如下内容：

```
umask 022
export ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=$ORACLE_BASE/product/10.2.0/db_1
export ORACLE_SID=orcl
export PATH=$PATH:$HOME/bin:$ORACLE_HOME/bin
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:/usr/lib
```

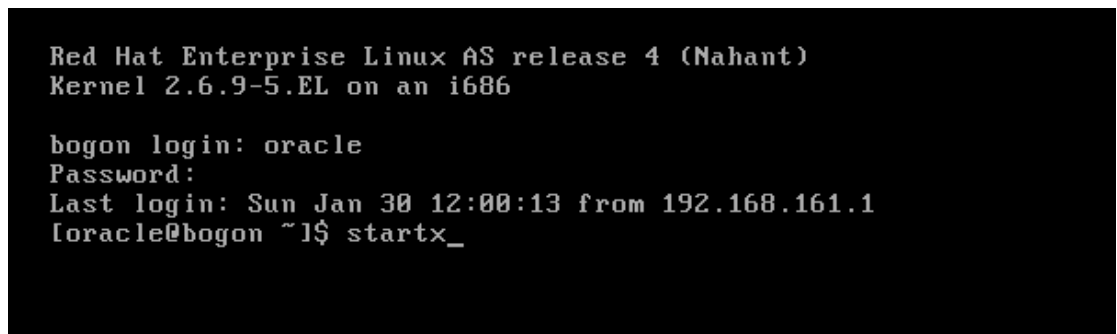


5、开始安装 Oracle

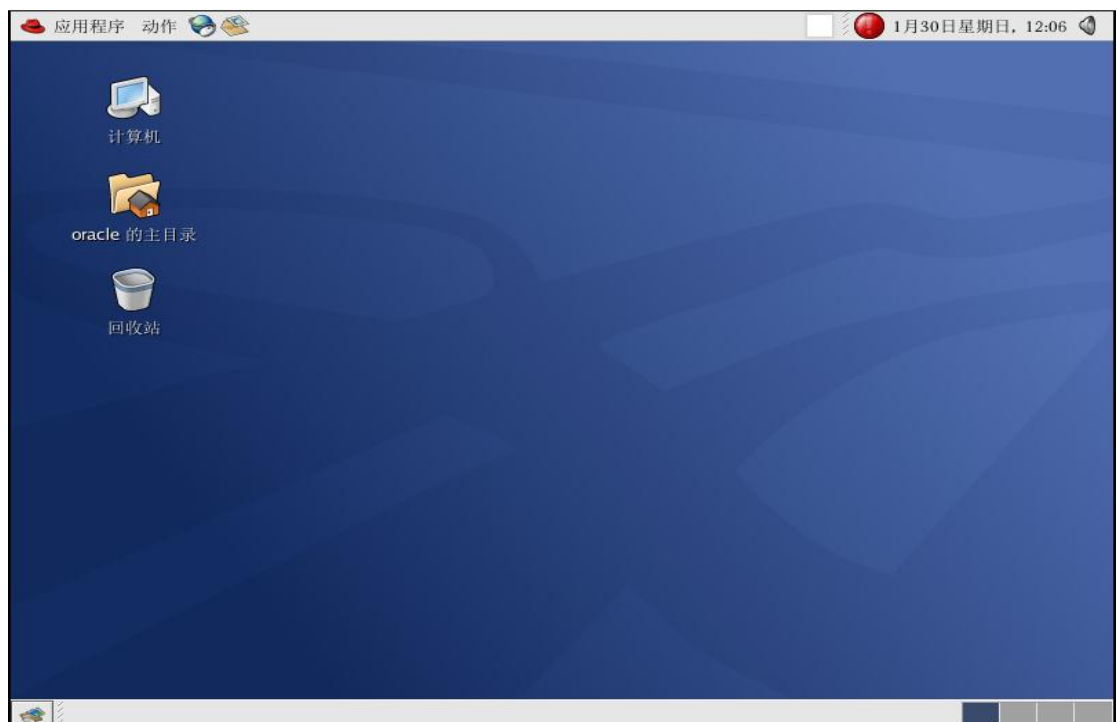
(1) 以 root 用户登录 Linux 系统，输入如下 Shell



(2) 切换到虚拟机，以 oracle 用户登录 Linux，输入如下 Shell



输入 startx 后出现 XWindow 界面，界面如下：



启动一个终端，键入如下 Shell 脚本：

```
oracle@bogon:/tmp/database
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[oracle@bogon ~]$ cd /tmp
[oracle@bogon tmp]$ cd ./database/
[oracle@bogon database]$ ./runInstaller &
```

输入如上 Shell 脚本后，将出现如下界面：

```
oracle@bogon:/tmp/database
文件(F) 编辑(E) 查看(V) 终端(T) 标签(B) 帮助(H)
[oracle@bogon ~]$ cd /tmp
[oracle@bogon tmp]$ cd ./database/
[oracle@bogon database]$ ./runInstaller &
[1] 3791
[oracle@bogon database]$ 正在启动 Oracle Universal Installer...

正在检查安装程序要求...

检查操作系统版本：必须是redhat-3, SuSE-9, redhat-4, UnitedLinux-1.0, asianux-1 o
r asianux-2
                                通过

所有安装程序要求均已满足。

准备从以下地址启动 Oracle Universal Installer /tmp/OraInst12011-01-30_12-08-12
PM 请稍候...
```

Oracle Database 10g

选择安装方法

☒ **基本安装(I)**
使用标准配置选项(需要输入的内容最少)执行完整的 Oracle Database 10g 安装。此选项使用文件系统进行存储, 并将一个口令用于所有数据库帐户。

Oracle 主目录位置(L): 浏览(B)...

安装类型(T):

UNIX DBA 组(G):

☒ 创建启动数据库(附加 720MB)(S)

全局数据库名(G):

数据库口令(P): 确认口令(C):


此口令将用于 SYS, SYSTEM, SYSMAN 和 DBSNMP 帐户。

☐ **高级安装(A)**
可以选择高级选项, 例如: 为 SYS, SYSTEM, SYSMAN 和 DBSNMP 帐户使用不同口令, 选择数据库字符集, 产品语言, 自动备份, 定制安装以及备用存储选项(例如自动存储管理)。

帮助(H) 上一步(B) **下一步(N)** 安装(I) 取消

ORACLE

- (3) 看到如上图所示界面后，根据实际情况修改全局数据库名，并输入数据库密码。



The screenshot shows the 'Select Installation Method' window of the Oracle Database 10g Universal Installer. The 'Basic Installation' radio button is selected. The window contains several input fields and a checkbox. The 'Oracle Main Directory Location' is set to '/u01/app/oracle/product/10.2.0/db_1'. The 'Installation Type' is set to 'Enterprise Edition (1.3GB)'. The 'UNIX DBA Group' is set to 'oinstall'. The checkbox 'Create Startup Database (adds 720MB)' is checked. The 'Global Database Name' is set to 'orcl'. The 'Database Password' and 'Confirm Password' fields both contain '*****'. A note below these fields states: 'This password is used for SYS, SYSTEM, SYSMAN and DBSNMP users.' The 'Advanced Installation' radio button is unselected. At the bottom, there are buttons for 'Help', 'Previous Step', 'Next Step', 'Install', and 'Cancel'.

Oracle Database 10g

选择安装方法

☒ 基本安装(I)

使用标准配置选项(需要输入的内容最少)执行完整的 Oracle Database 10g 安装。此选项使用文件系统进行存储, 并将一个口令用于所有数据库帐户。

Oracle 主目录位置(L): /u01/app/oracle/product/10.2.0/db_1 浏览(B)...

安装类型(T): 企业版 (1.3GB)

UNIX DBA 组(G): oinstall

☒ 创建启动数据库 (附加 720MB)(S)

全局数据库名(G): orcl

数据库口令(P): ***** 确认口令(C): *****

此口令将用于 SYS, SYSTEM, SYSMAN 和 DBSNMP 帐户。

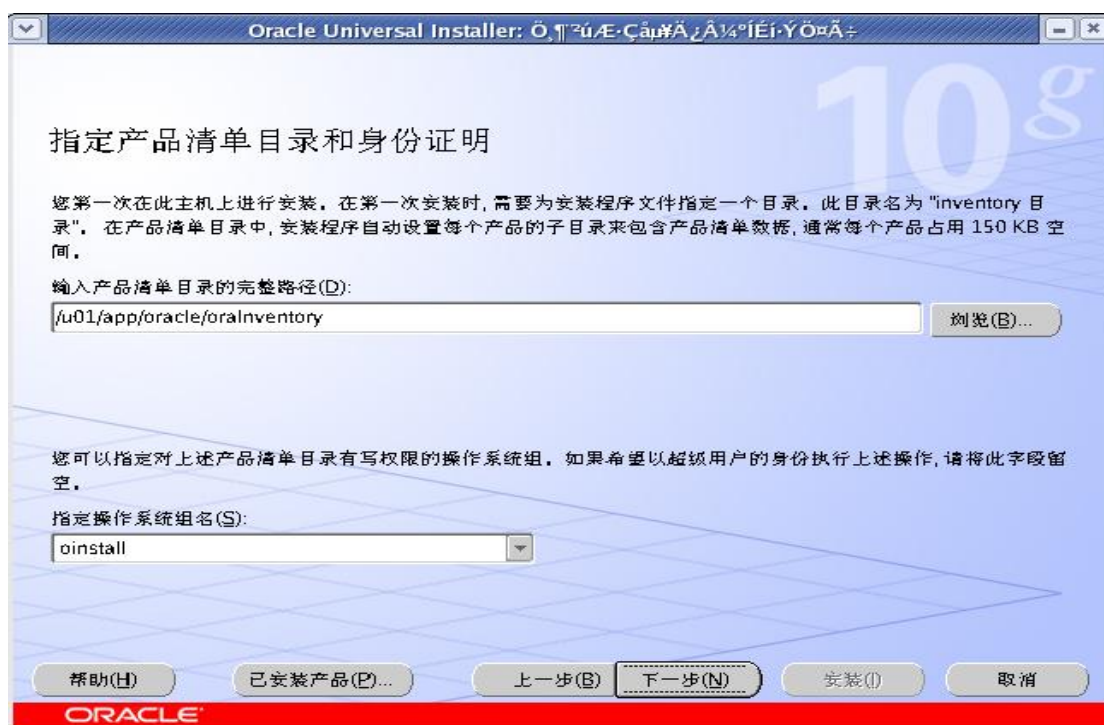
☐ 高级安装(A)

可以选择高级选项, 例如: 为 SYS, SYSTEM, SYSMAN 和 DBSNMP 帐户使用不同口令, 选择数据库字符集, 产品语言, 自动备份, 定制安装以及备用存储选项 (例如自动存储管理)。

帮助(H) 上一步(B) 下一步(N) 安装(I) 取消

ORACLE

点击“下一步”出现如下界面:



The screenshot shows the 'Specify Product List Directory and Identity' window of the Oracle Universal Installer. The window contains a text box for the 'Product List Directory Full Path' set to '/u01/app/oracle/orainventory'. There is a 'Browse(B)...' button next to it. Below this, there is a text box for 'Specify Operating System Group(S)' set to 'oinstall'. At the bottom, there are buttons for 'Help', 'Already Installed Products(P)...', 'Previous Step', 'Next Step', 'Install', and 'Cancel'.

Oracle Universal Installer: 指定产品清单目录和身份证明

指定产品清单目录和身份证明

您第一次在此主机上进行安装。在第一次安装时, 需要为安装程序文件指定一个目录。此目录名为 "inventory 目录"。在产品清单目录中, 安装程序自动设置每个产品的子目录来包含产品清单数据, 通常每个产品占用 150 KB 空间。

输入产品清单目录的完整路径(D): /u01/app/oracle/orainventory 浏览(B)...

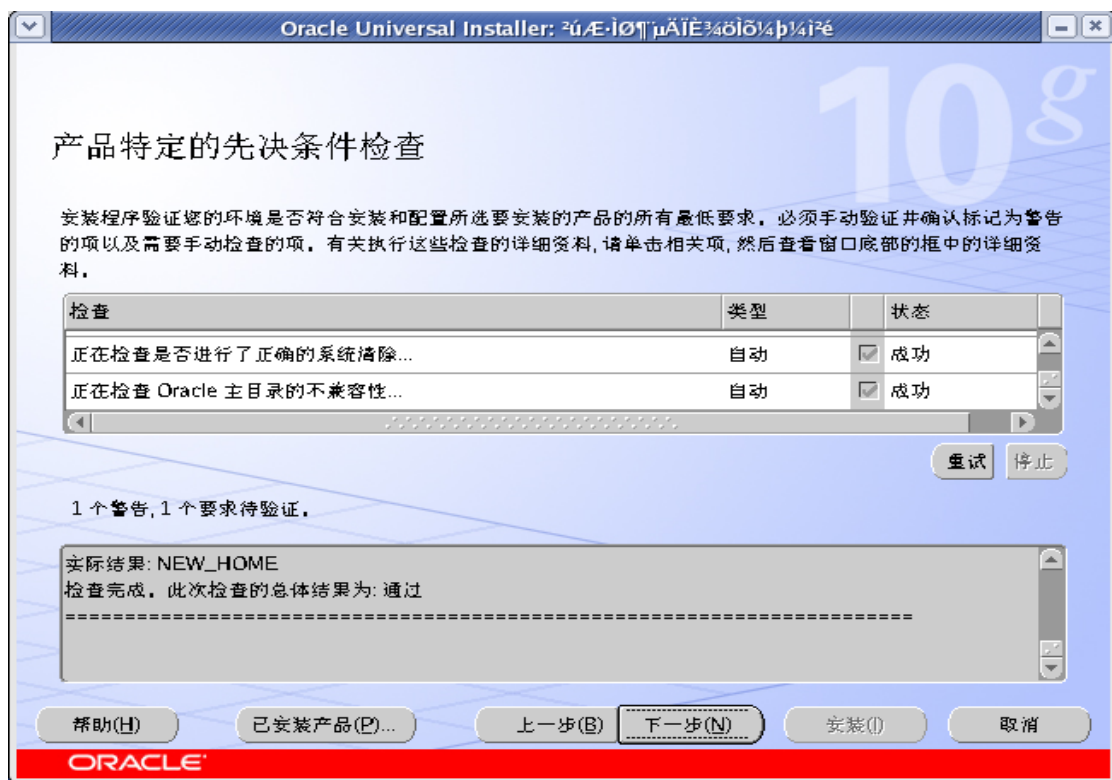
您可以指定对上述产品清单目录有写权限的操作系统组。如果希望以超级用户的身份执行上述操作, 请将此字段留空。

指定操作系统组名(S): oinstall

帮助(H) 已安装产品(P)... 上一步(B) 下一步(N) 安装(I) 取消

ORACLE

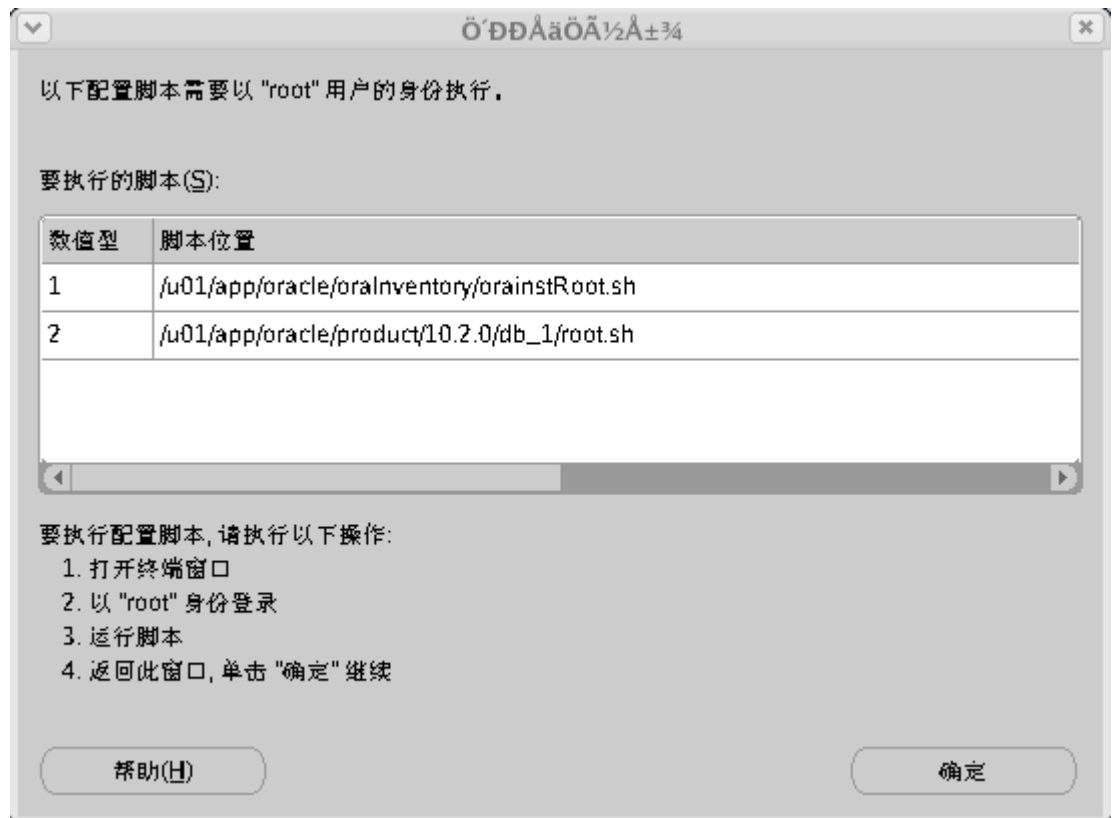
此处不需修改，直接点击“下一步”出现如下界面：



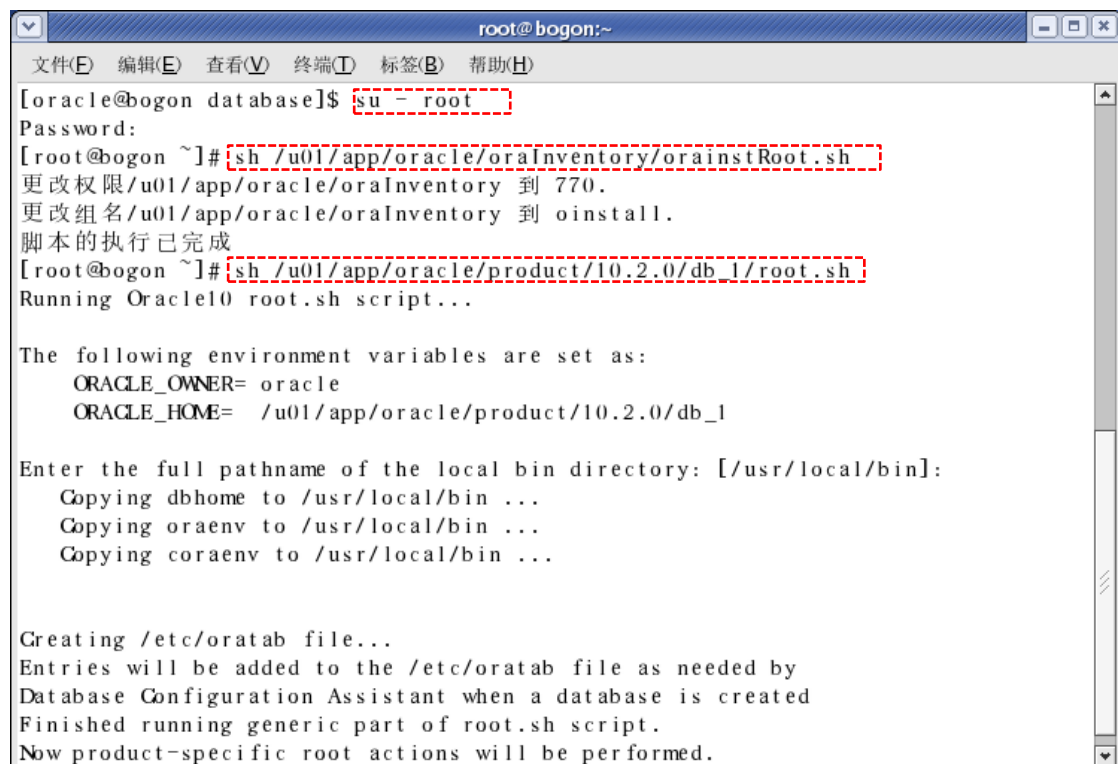
等待检测后，点击“下一步”按钮，出现如下界面：



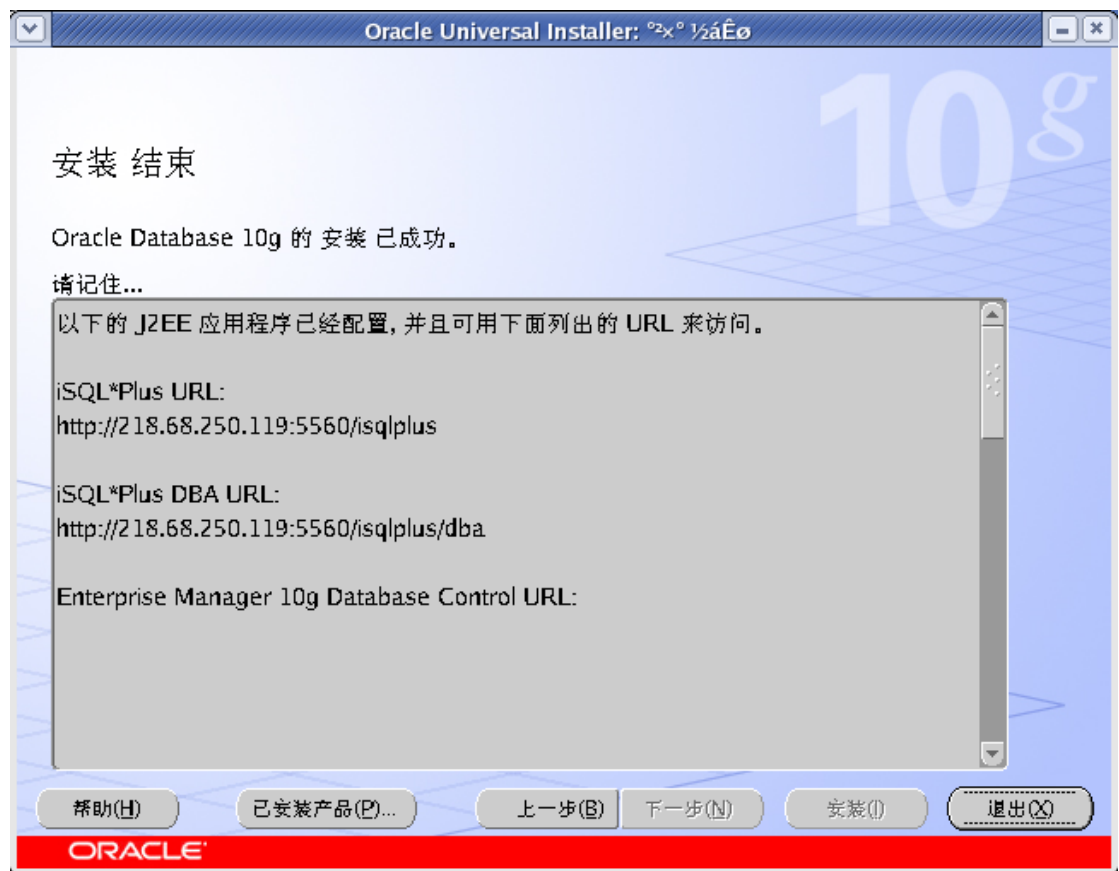
确认一下，点击“下一步”按钮，当安装过程出现如下界面的时候



启动一个命令窗口，输入红色框标识的命令：



运行脚本后，并且换到 oracle 用户，点击“确定”按钮，出现如下界面：



点击“退出”按钮即可。在终端输入 reboot 重新引导 Linux 系统。

6、在 Linux 下管理 Oracle 数据库

- (1) 首先以 oracle 用户登录 Linux 系统
- (2) 常用命令

启动监听器

```
lsnrctl start
```

停止监听器

```
lsnrctl stop
```

查看监听器状态

```
lsnrctl status
```

启动数据库

```
sqlplus "/as sysdba"
```

```
startup
```

```
# 停止数据库
shutdown immediate
```

(3) 启用 scott 用户

由于 Oracle10G 默认没有启用 scott 用户，需要手动启用，键入如下命令：

```
[oracle@bogon ~]$ lsnrctl start
```

```
[oracle@bogon ~]$ sqlplus "/as sysdba"
```

```
SQL> startup
```

```
SQL> conn sys as sysdba;
Enter password:
Connected.
SQL>
```

```
SQL> alter user scott account unlock;
```

```
SQL> commit;
```

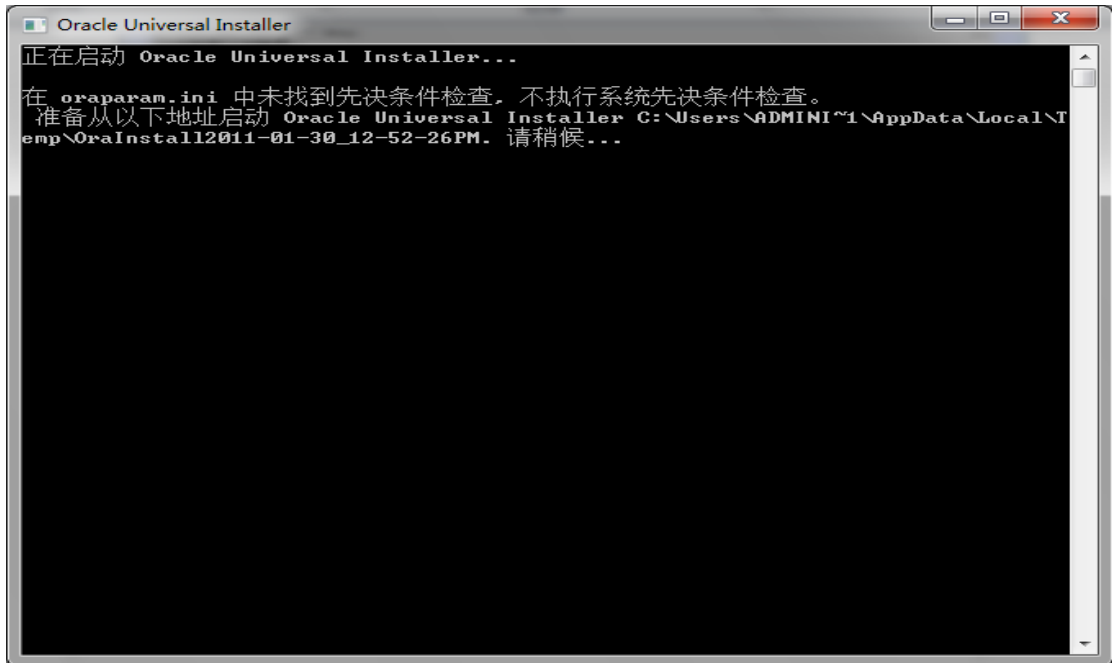
```
SQL> conn scott/tiger;
ERROR:
ORA-28001: the password has expired
```

```
Changing password for scott
New password:
Retype new password:
Password changed
Connected.
```

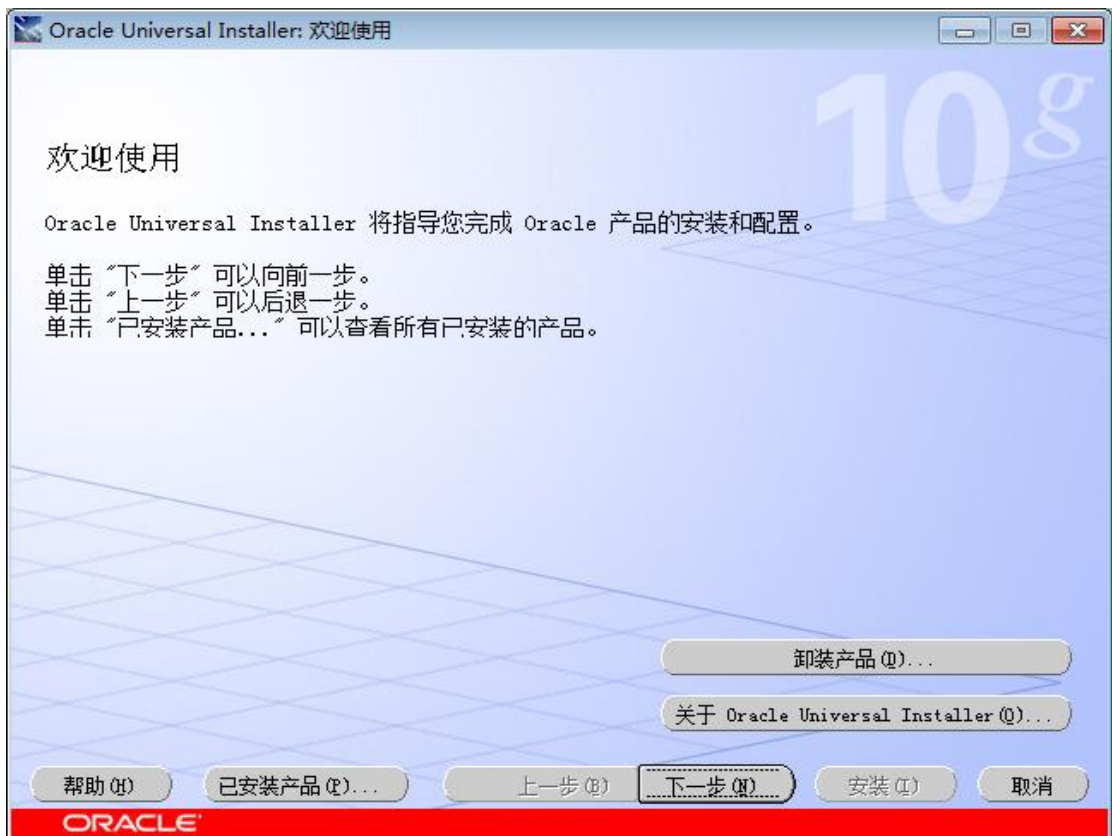
当使用 scott 用户登录 SQL*Plus 的时候，Oracle 会检测 scott 用户的密码，需要为 scott 用户重置密码，输入两次一样的就可以了，我输入的是 tiger。

7、安装 Oracle10G 客户端

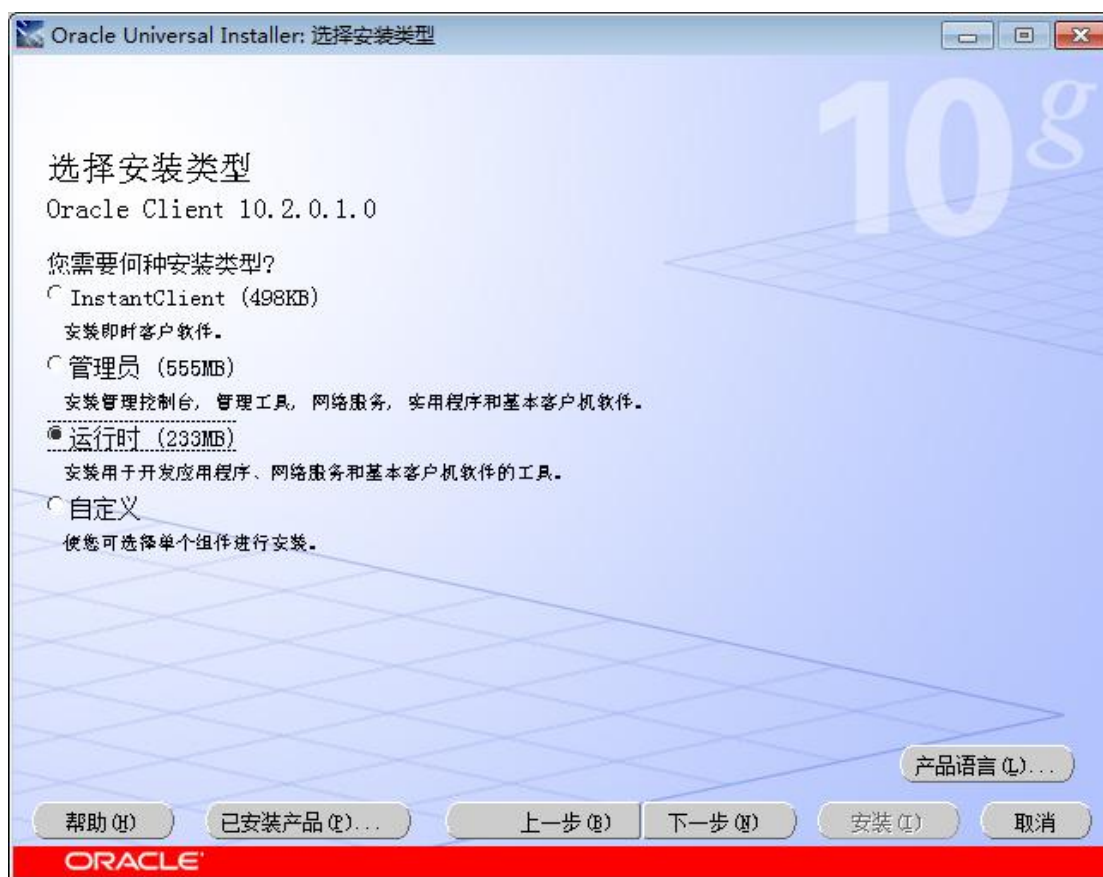
回到 Windows 系统，解压缩下载的 10201_client_win32.zip，双击 setup.exe 出现如下界面：



检测后，出现如下界面：



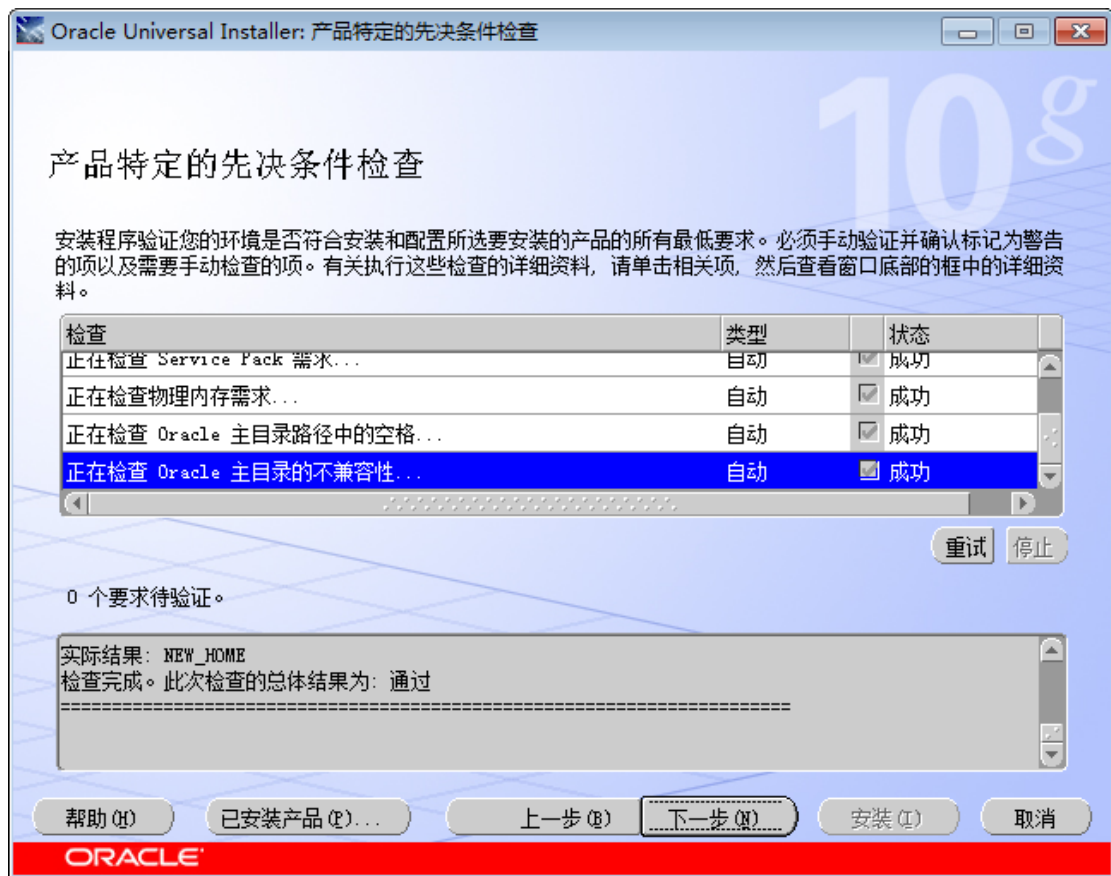
直接点击“下一步”按钮，出现如下界面：



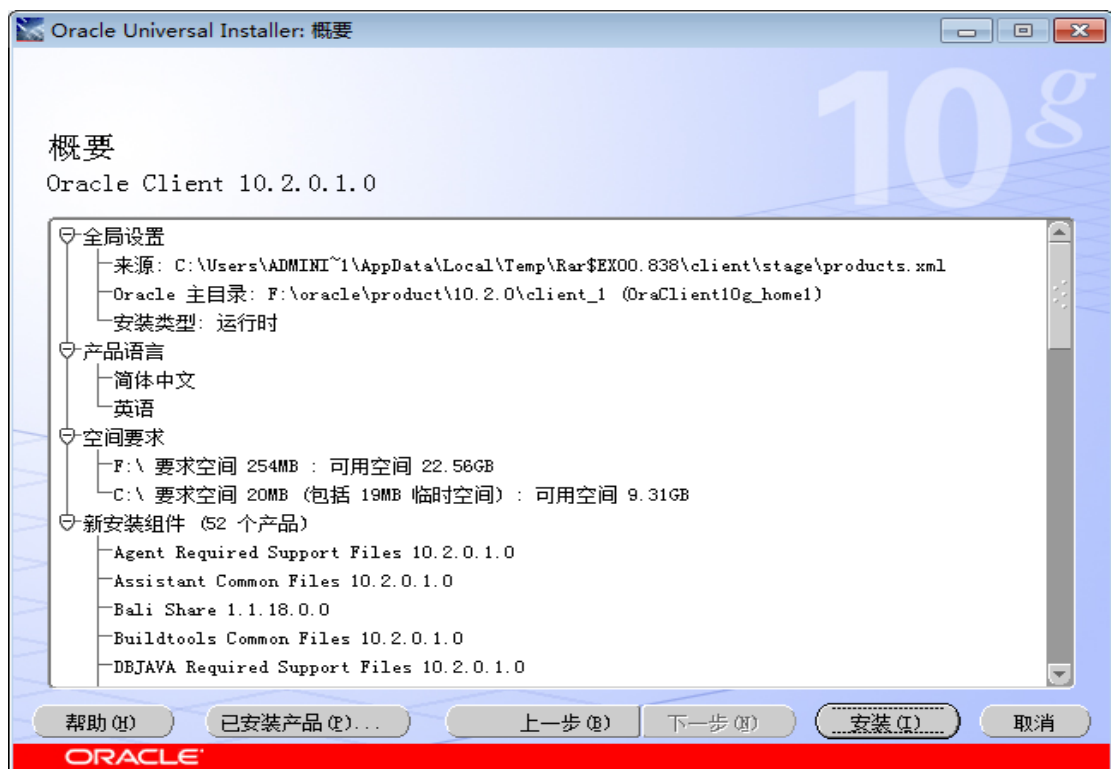
此处，选择“运行时”即可，点击“下一步”按钮，出现如下界面：



根据情况修改 Oracle 的安装路径，点击“下一步”按钮出现如下界面：



待检测通过后，直接点击“下一步”按钮，出现如下界面：



点击“安装”按钮，出现如下界面：



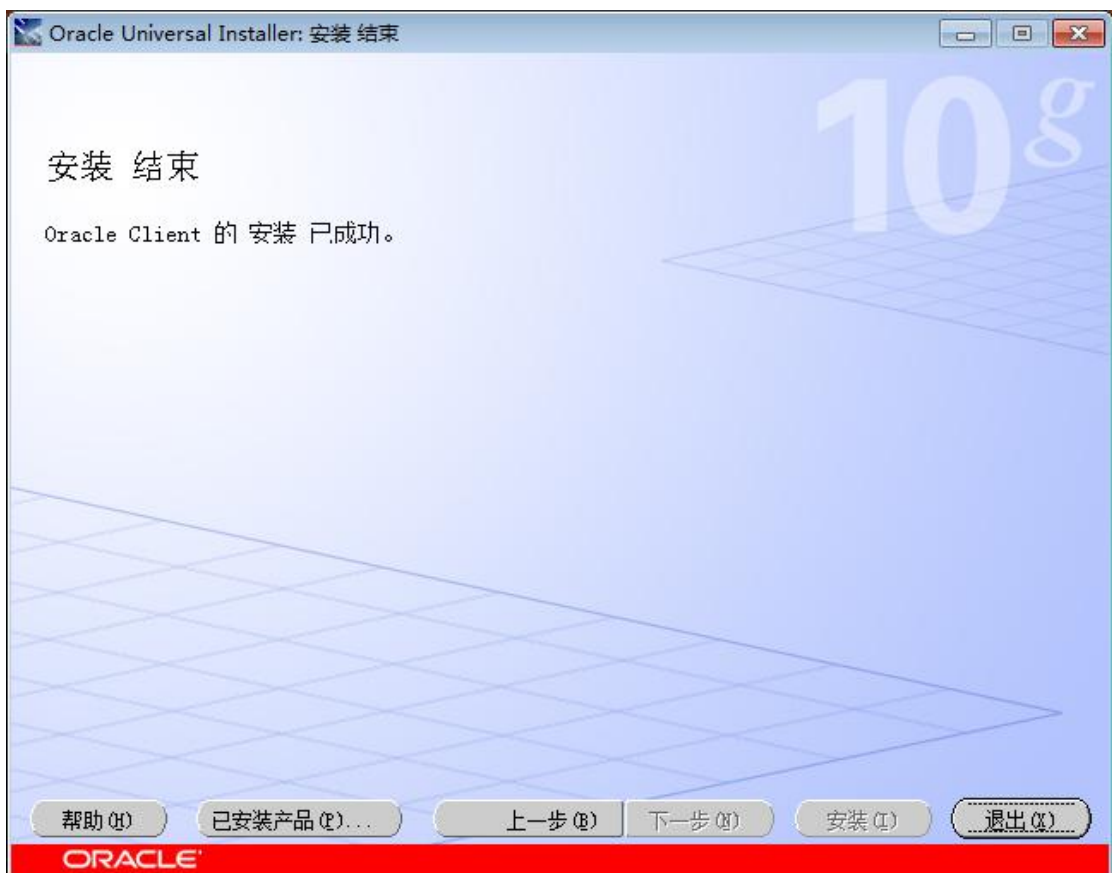
当安装过程中，出现如下界面的时候：



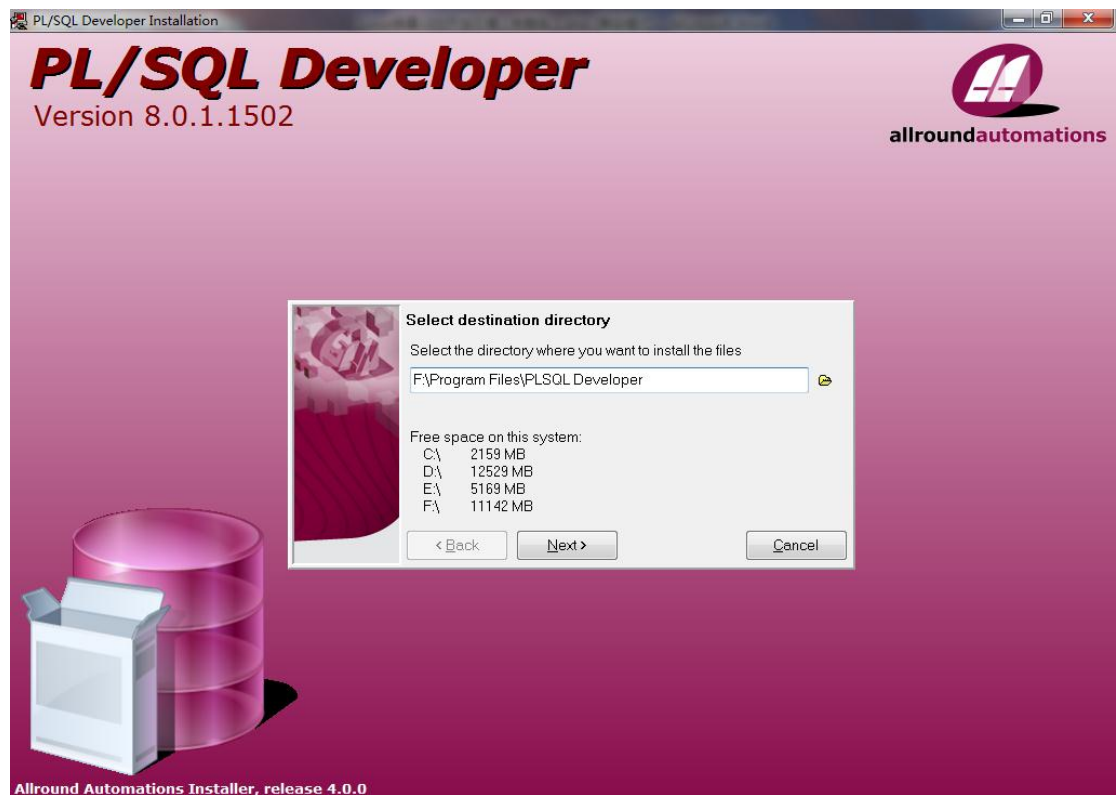
直接点击“下一步”，出现如下界面：



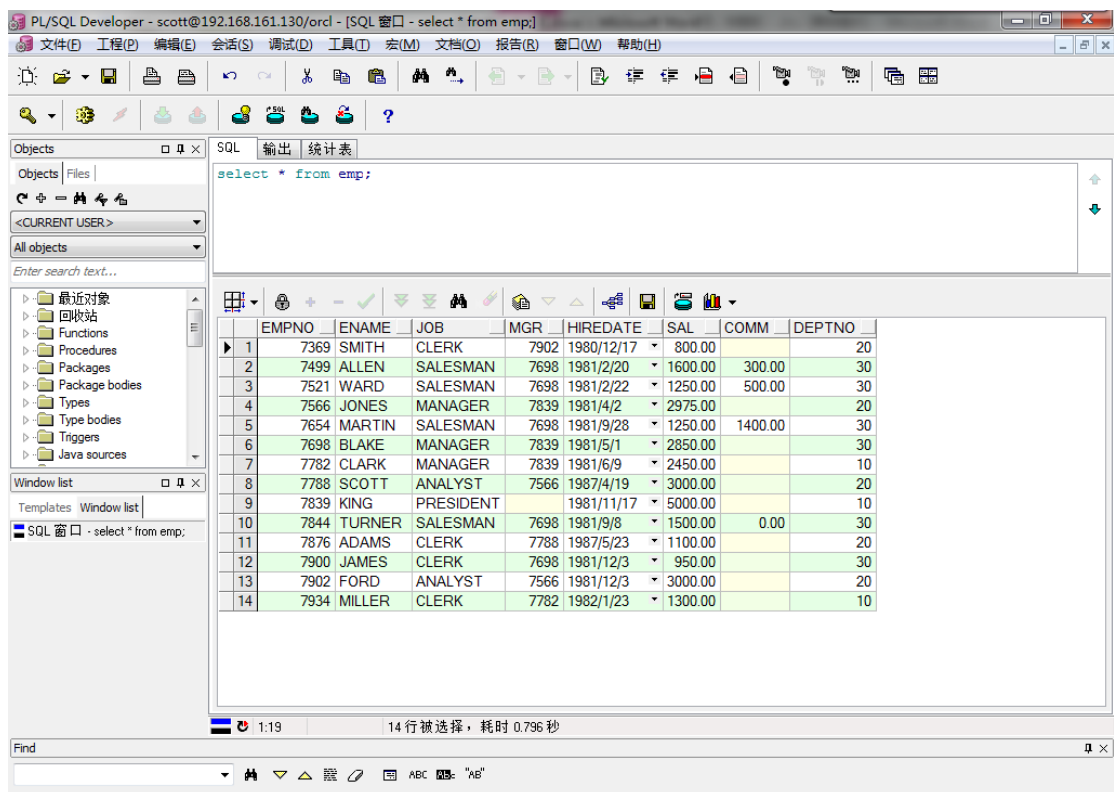
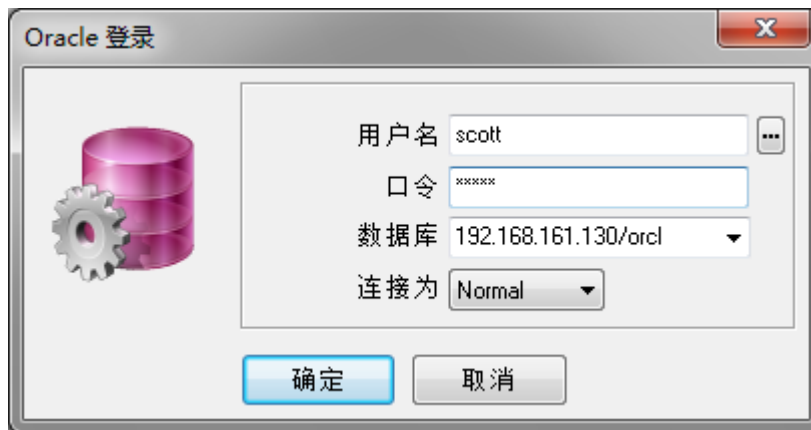
点击“完成”按钮，出现如下界面：



8、安装 PL/SQL Developer



9、测试连接 Linux 下的 Oracle 数据库

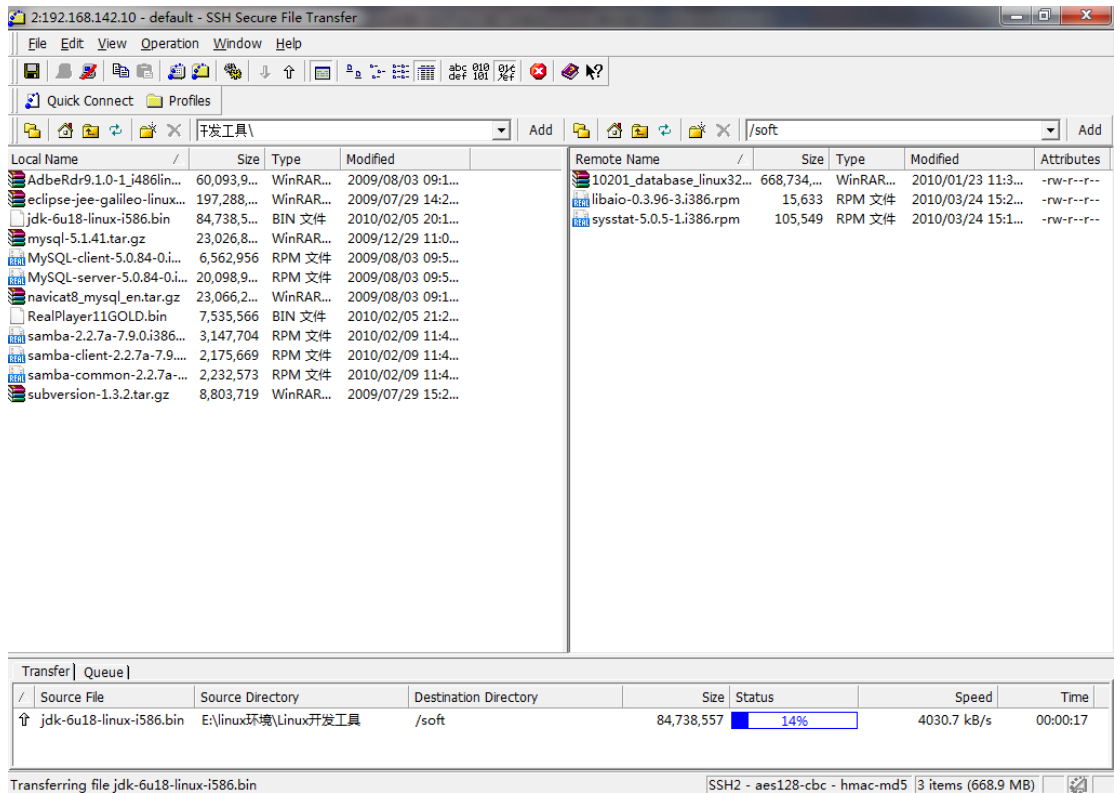


至此，Linux 搭建 Oracle 环境的过程已经全部完成了，可以看到，Linux 安装 Oracle 是非常复杂的一件事情，很容易出现各种各样的问题，我在安装的时候就出现了一些问题，这些问题大多都是监听的问题，只需要修改 listener.ora 文件即可。在下面我将讲解如何在 Linux 上搭建 Java 开发环境。

LINUX 搭建 JAVA 开发环境

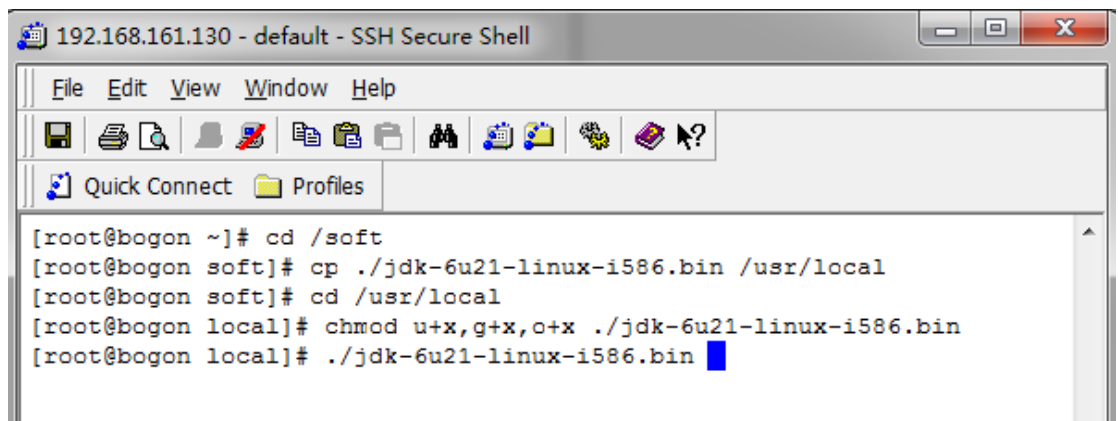
本部分主要讲解如何在 Linux 上安装 JDK，配置环境变量，这部分比较简单，闲话少说，直接看步骤。

1、切换到 root 用户将 JDK 上传到/soft 目录下



2、安装 JDK

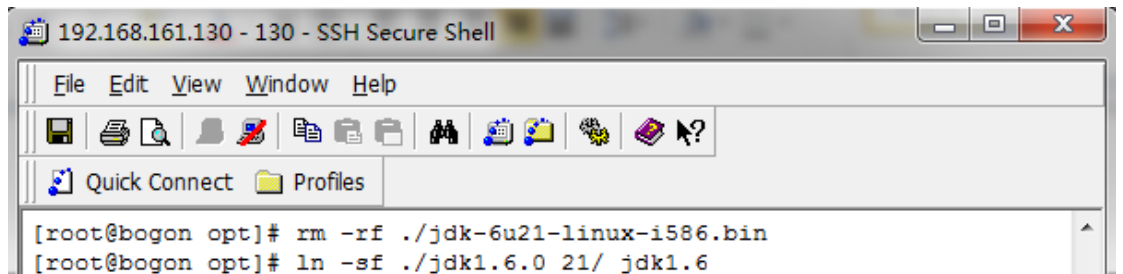
(1) 执行如下 Shell 脚本



弹出信息后直接敲空格即可，最后在输入协议的地方输入 yes 回车即可。

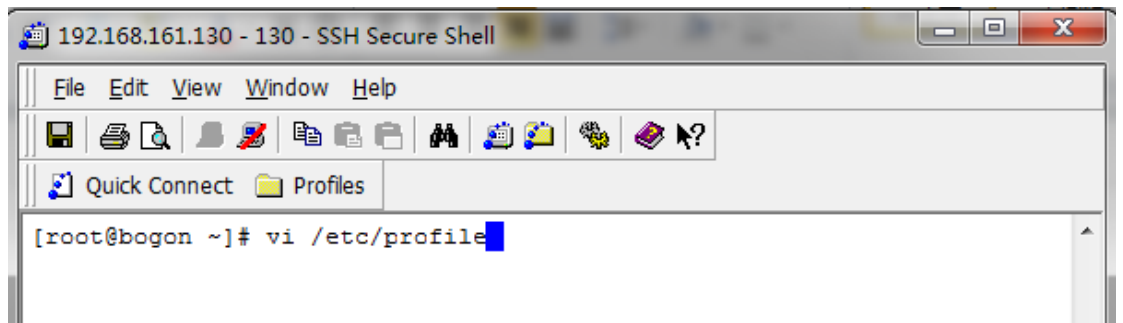
(2) 配置 JDK 环境变量

a) 执行如下 Shell 脚本，创建软链接



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon opt]# rm -rf ./jdk-6u21-linux-i586.bin
[root@bogon opt]# ln -sf ./jdk1.6.0_21/ jdk1.6
```

b) 执行如下 Shell，编辑环境变量

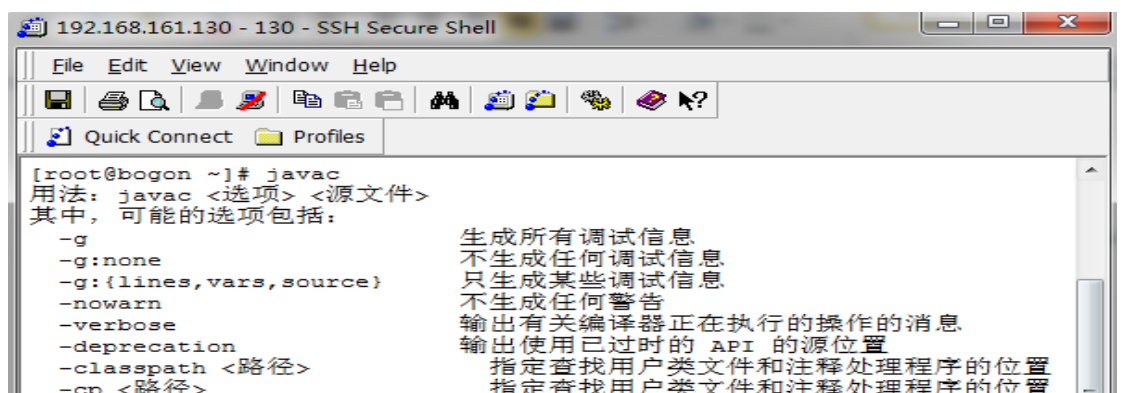


```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon ~]# vi /etc/profile
```

在该文件最后添加如下内容：

```
export JAVA_HOME=/opt/jdk1.6
export CLASSPATH=.:$JAVA_HOME/lib/tools.jar:$JAVA_HOME/lib/dt.jar
export PATH=$PATH:$JAVA_HOME/bin
```

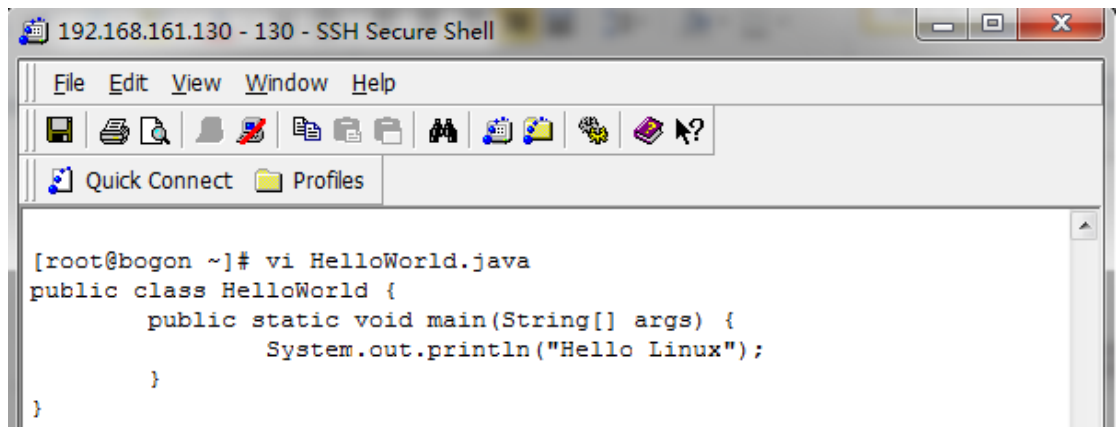
使用 logout 命令注销 Linux，重新进入 Linux 系统，输入 javac 查看 java 环境是否正确配置：



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon ~]# javac
用法: javac <选项> <源文件>
其中, 可能的选项包括:
-g          生成所有调试信息
-g:none     不生成任何调试信息
-g:{lines,vars,source} 只生成某些调试信息
-nowarn     不生成任何警告
-verbose    输出有关编译器正在执行的操作的消息
-deprecation 输出使用已过时的 API 的源位置
-classpath <路径> 指定查找用户类文件和注释处理程序的位置
-cp <路径> 指定查找用户类文件和注释处理程序的位置
```

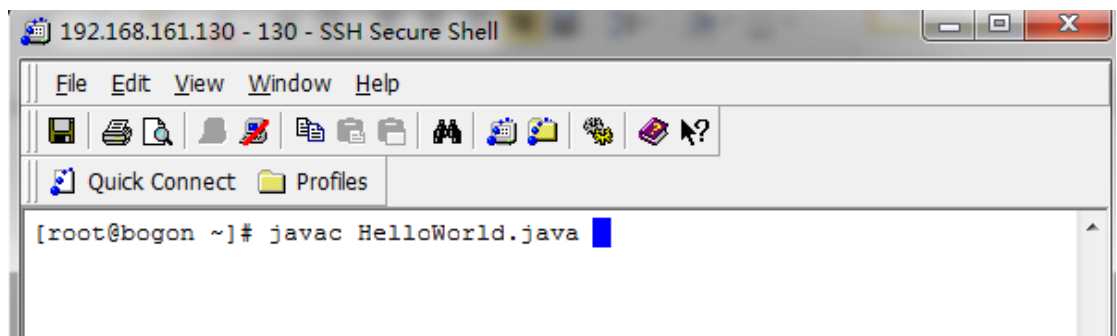
3、编写一个 Java 程序，测试 Java 开发环境是否配置正确。

- (1) 启动一个 vi 编辑器，输入如下 Java 代码



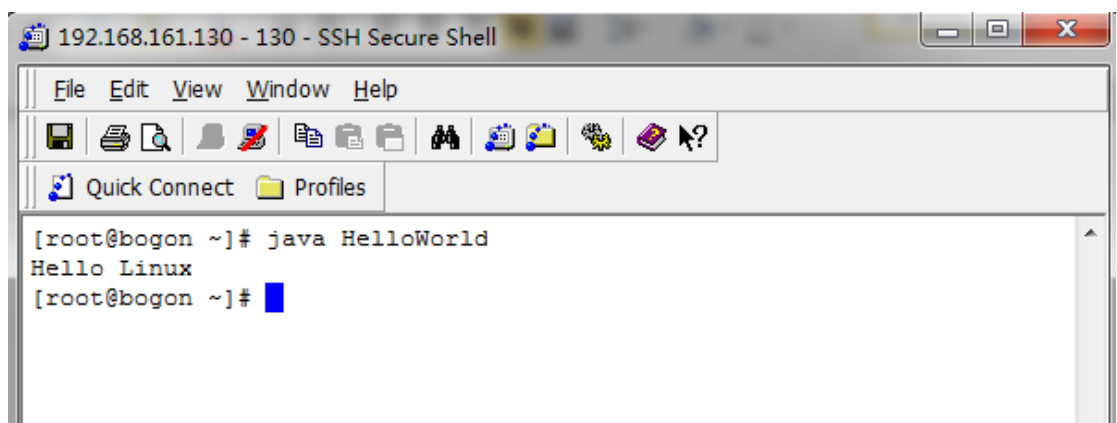
```
[root@bogon ~]# vi HelloWorld.java
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello Linux");
    }
}
```

- (2) 使用 javac 编译 Java 程序



```
[root@bogon ~]# javac HelloWorld.java
```

- (3) 执行 Java 程序



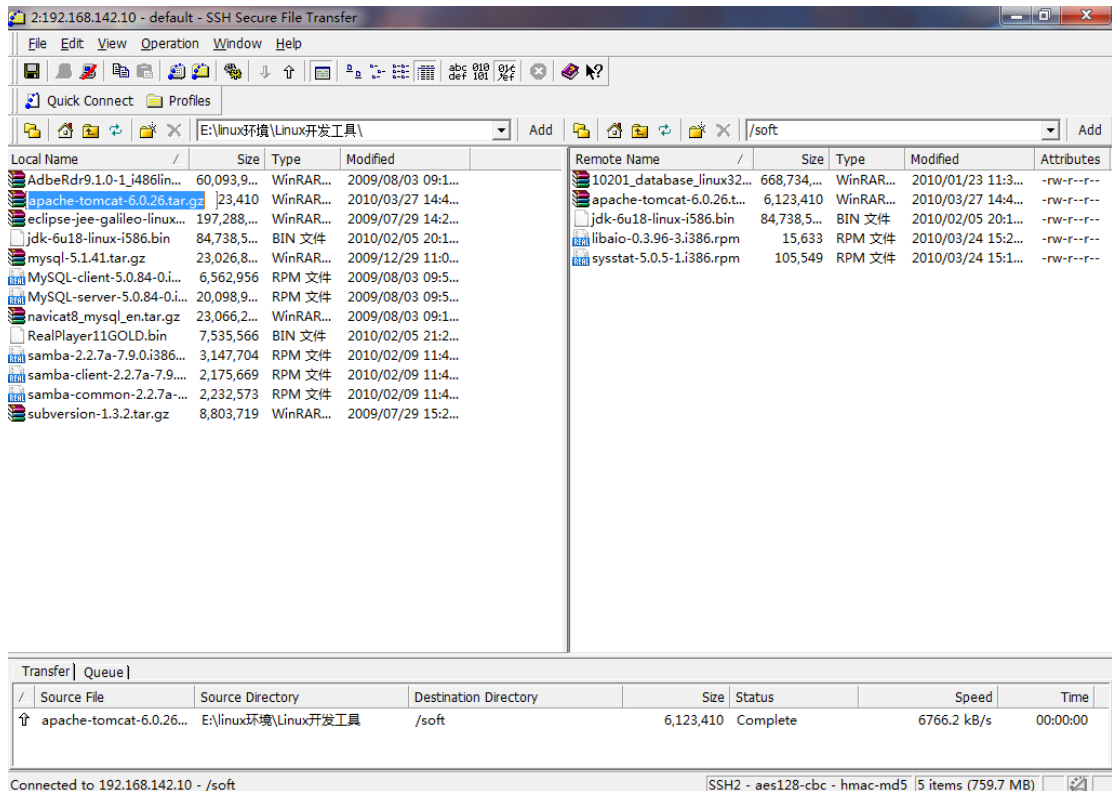
```
[root@bogon ~]# java HelloWorld
Hello Linux
[root@bogon ~]#
```

至此，Java 开发环境搭建完成，在下一章节我将讲解如何搭建 Tomcat 服务器。

TOMCAT 服务器安装

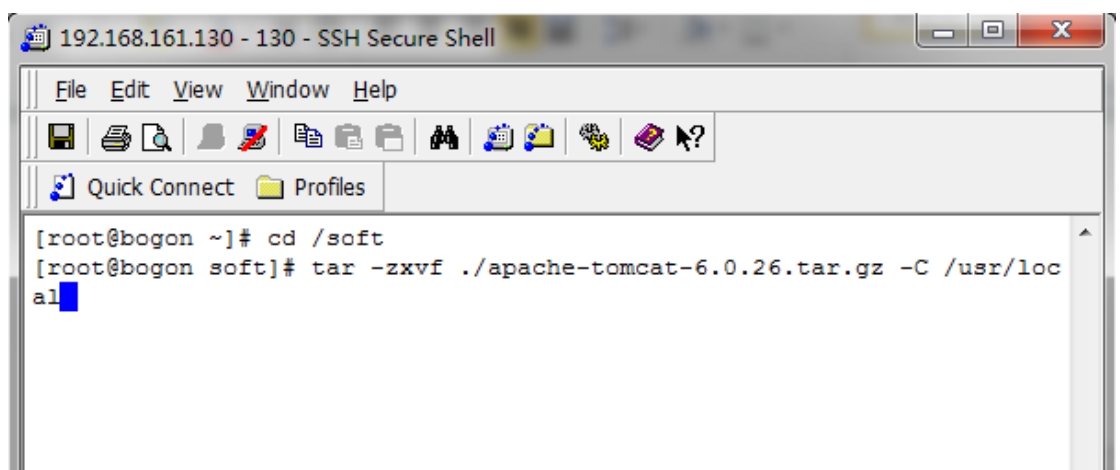
本节主要讲解如何在 Linux 上安装 Tomcat 服务器，以及环境变量的配置，并通过一个具体的实例来测试 Tomcat 服务器是否正常工作。

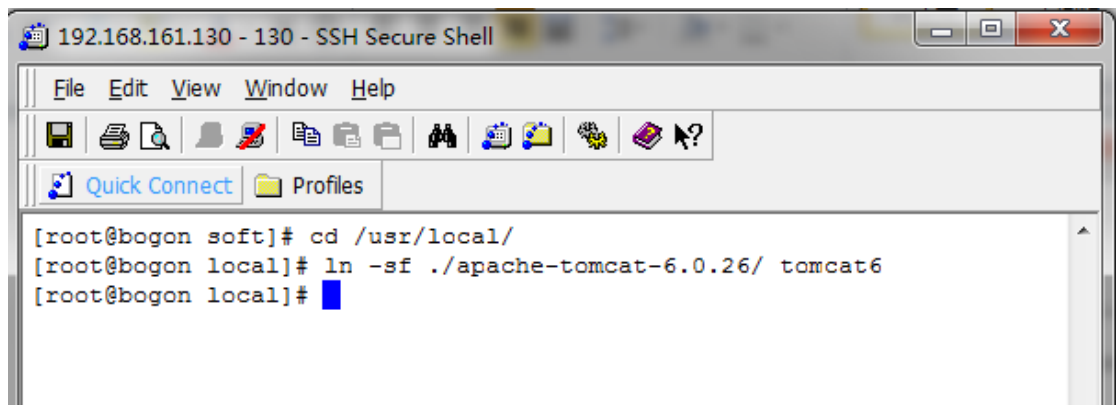
1、切换到 root 用户将 Tomcat 上传到/soft 目录下



2、安装 Tomcat，配置环境变量：

(1) 执行如下 Shell 脚本

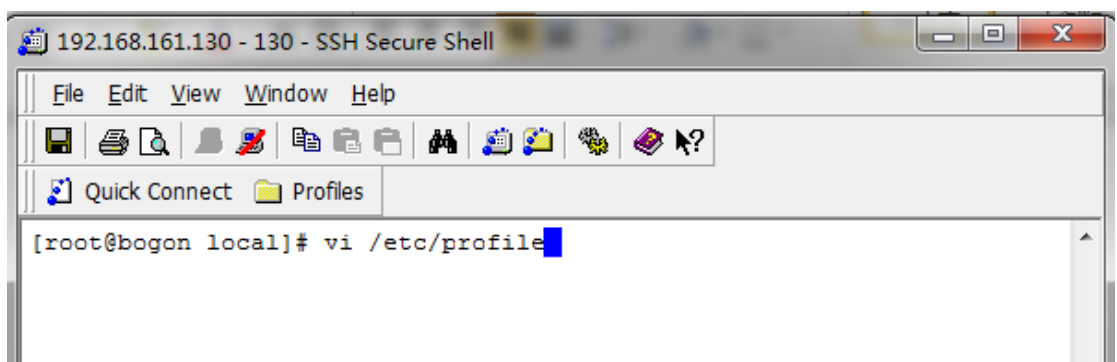




```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon soft]# cd /usr/local/
[root@bogon local]# ln -sf ./apache-tomcat-6.0.26/ tomcat6
[root@bogon local]#
```

(2) 配置 Tomcat 环境变量



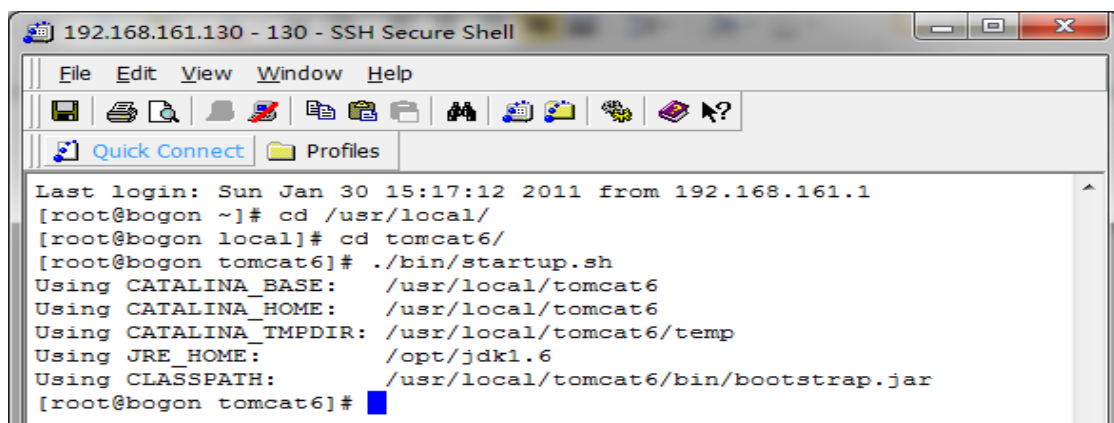
```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon local]# vi /etc/profile
```

修改该文件的 export 部分内容，修改后的内容如下：

```
export JAVA_HOME=/opt/jdk1.6
export CATALINA_HOME=/usr/local/tomcat6
export CLASSPATH=.:$JAVA_HOME/lib/tools.jar:$JAVA_HOME/lib/dt.jar:$
CATALINA_HOME/lib/servlet-api.jar
export PATH=$PATH:$JAVA_HOME/bin
```

3、测试 Tomcat 服务器，是否正常安装，输入如下 Shell 脚本以启动 Tomcat 服务器



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

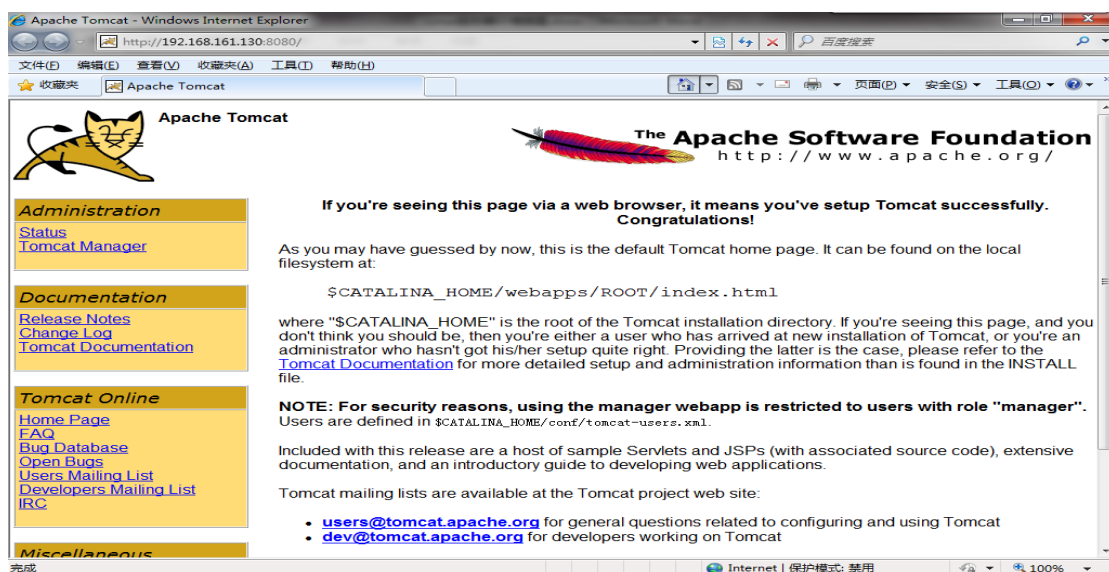
Last login: Sun Jan 30 15:17:12 2011 from 192.168.161.1
[root@bogon ~]# cd /usr/local/
[root@bogon local]# cd tomcat6/
[root@bogon tomcat6]# ./bin/startup.sh
Using CATALINA_BASE:   /usr/local/tomcat6
Using CATALINA_HOME:   /usr/local/tomcat6
Using CATALINA_TMPDIR: /usr/local/tomcat6/temp
Using JRE_HOME:        /opt/jdk1.6
Using CLASSPATH:       /usr/local/tomcat6/bin/bootstrap.jar
[root@bogon tomcat6]#
```

看到如上界面后，就说明 Tomcat 已经安装成功，并成功启动。

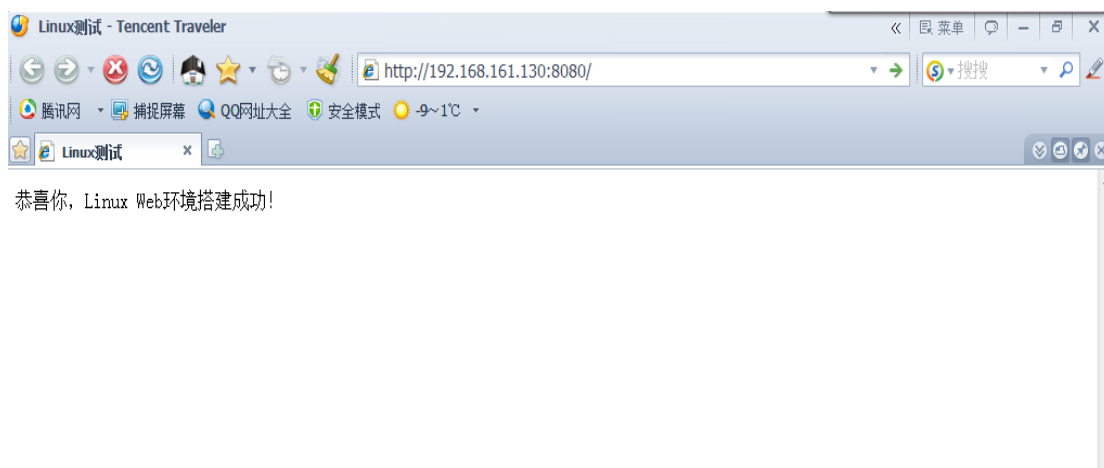
4、远程访问 Tomcat 服务器，并编写测试程序。

(1) 远程访问 Tomcat 服务器

在 Windows 系统中启动一个浏览器，输入远程 Tomcat 服务器地址，如果出现如下界面即说明访问成功：



(2) 编写一个测试 Web 应用，我们在 Windows 下编写一个 JavaWeb 程序并上传到远程服务器上，最终结果如下：

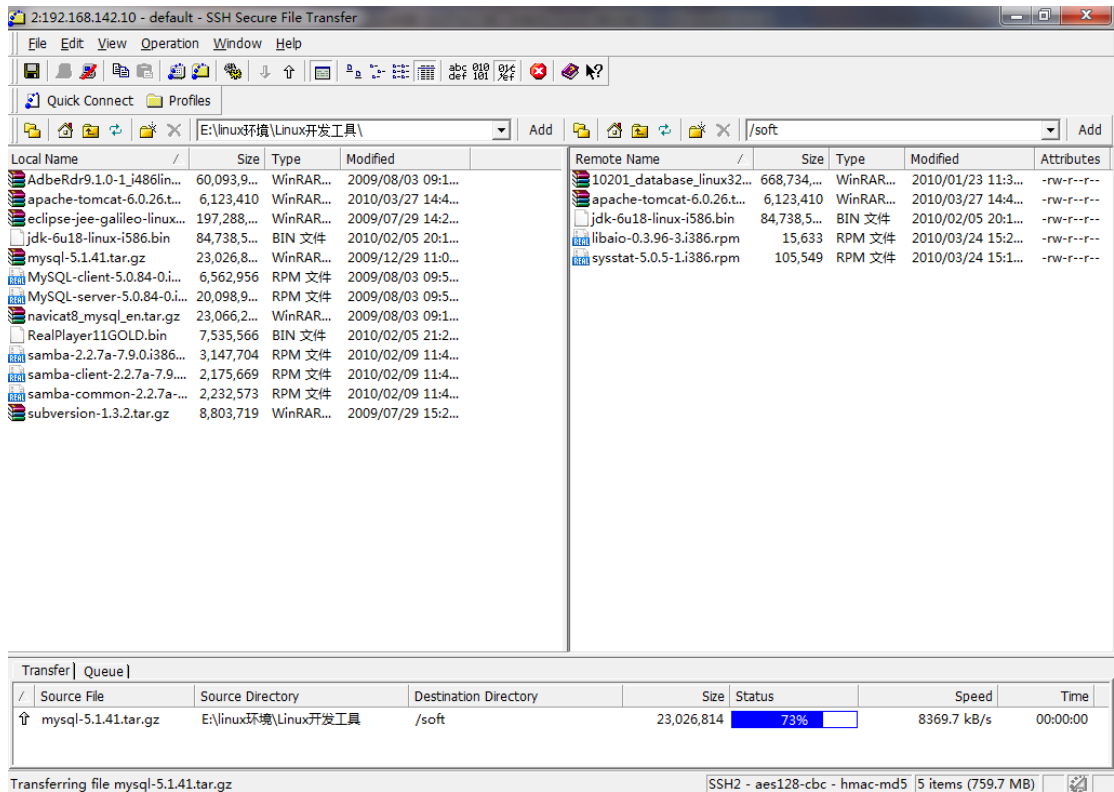


至此，Linux 系统下搭建 Tomcat 服务器就完成了，并且通过一个实例测试了服务器运行状态，在下面的章节中将讲解如何在 Linux 下基于源码编译方式安装 MySQL 数据库。

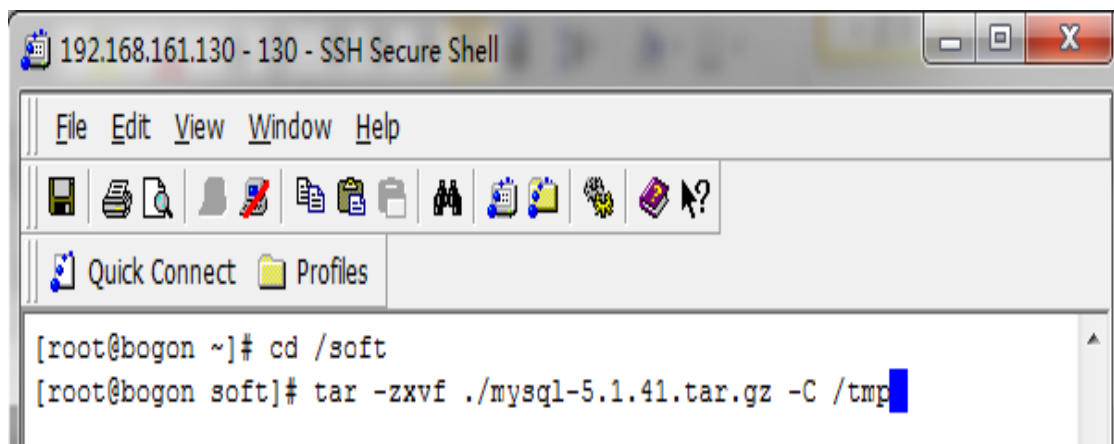
基于源码方式编译、安装 MySQL 数据库

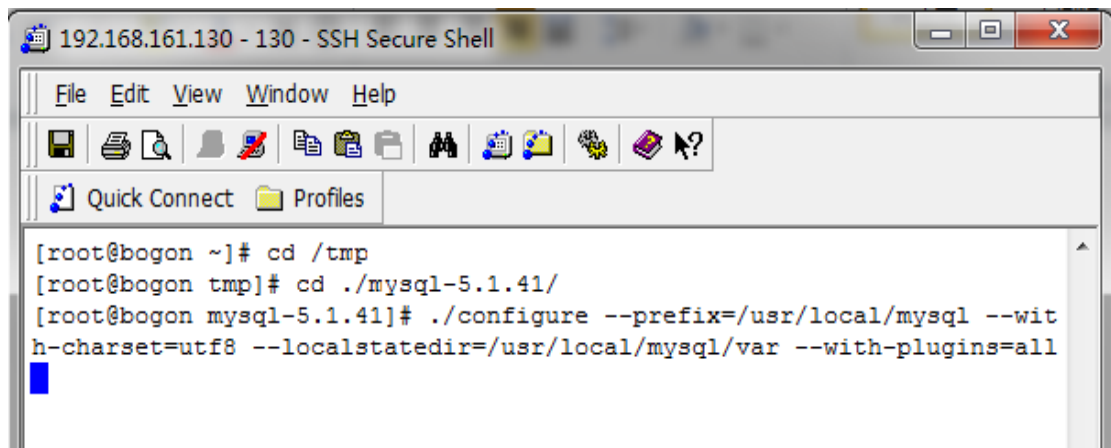
MySQL 数据库，是一个开源的、跨平台的高性能数据库，在 Windows 上安装 MySQL 数据库很简单，但是在 Linux 系统上安装是一件比较麻烦的事情，本节采用源码编译方式安装 MySQL 数据库，之所以采用源码方式编译，是为了后面搭建 PHP 环境做准备。

1、 切换到 root 用户将 MySQL 上传到/soft 目录下



2、 执行如下 Shell 脚本



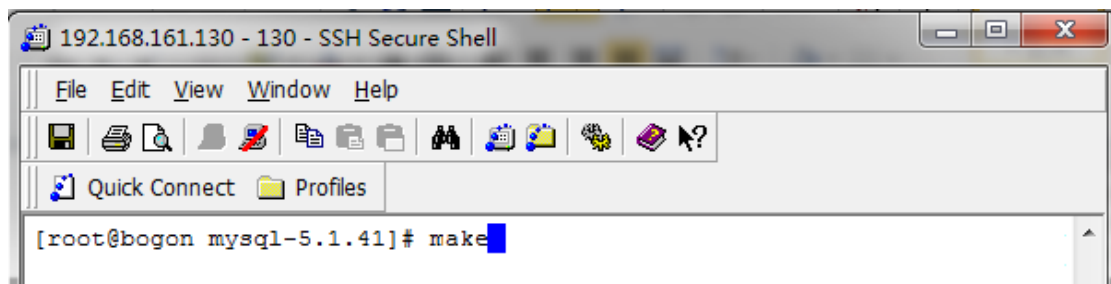


192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon ~]# cd /tmp
[root@bogon tmp]# cd ./mysql-5.1.41/
[root@bogon mysql-5.1.41]# ./configure --prefix=/usr/local/mysql --with-charset=utf8 --localstatedir=/usr/local/mysql/var --with-plugins=all
```

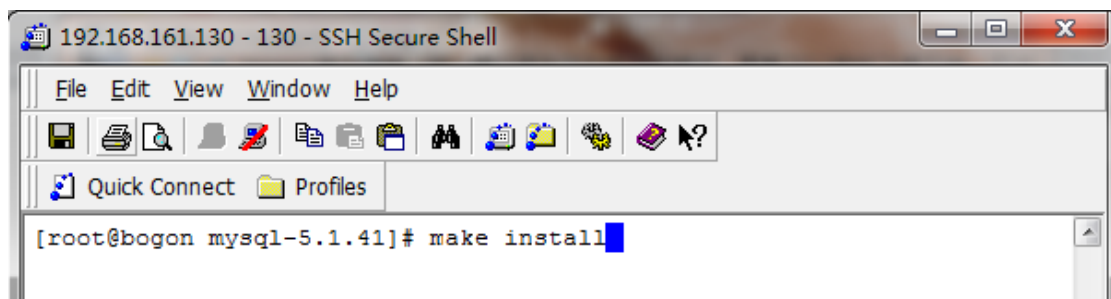


192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon mysql-5.1.41]# make
```

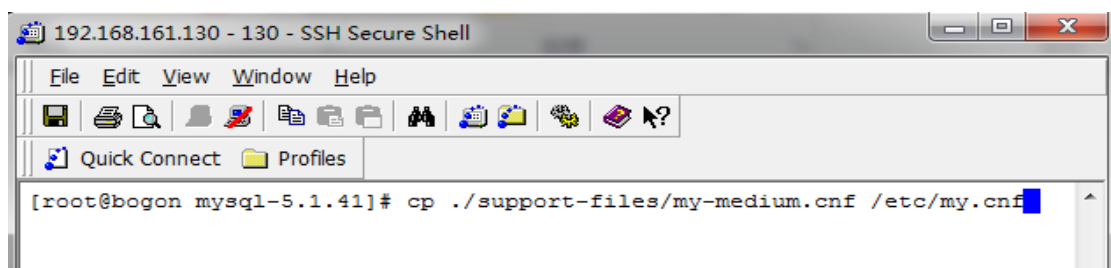


192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon mysql-5.1.41]# make install
```

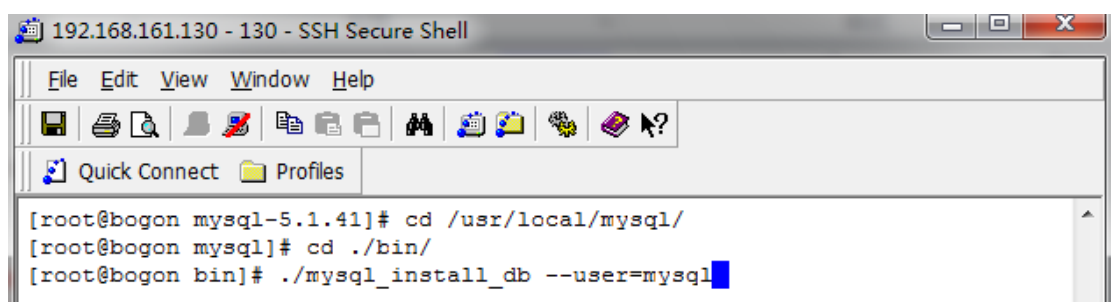


192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon mysql-5.1.41]# cp ./support-files/my-medium.cnf /etc/my.cnf
```

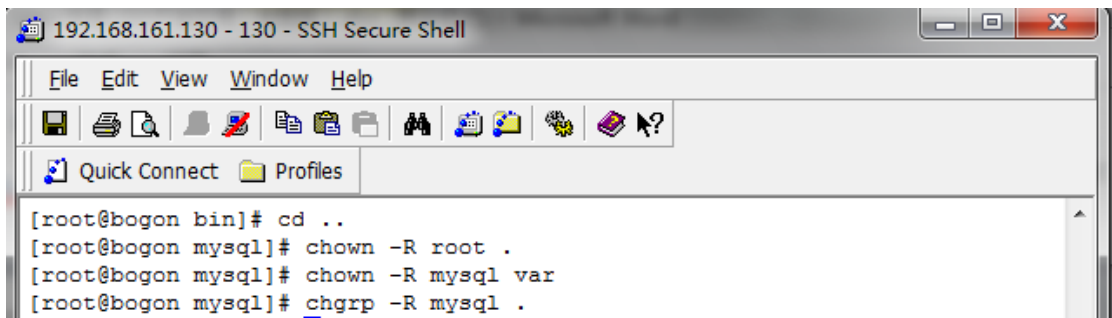


192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

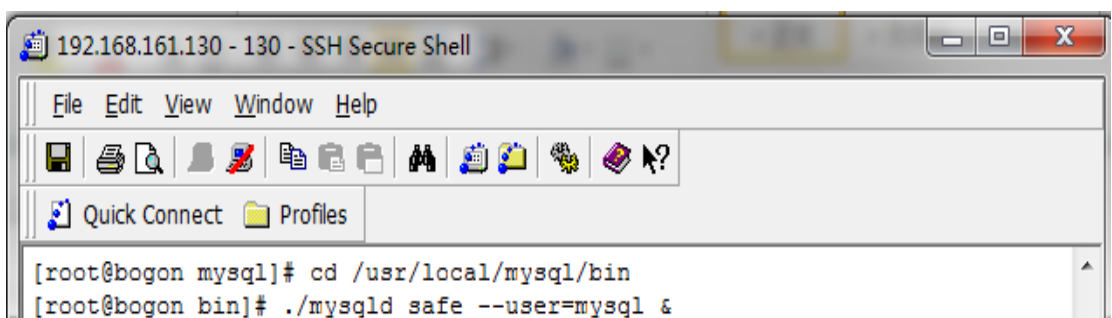
```
[root@bogon mysql-5.1.41]# cd /usr/local/mysql/
[root@bogon mysql]# cd ./bin/
[root@bogon bin]# ./mysql_install_db --user=mysql
```



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon bin]# cd ..
[root@bogon mysql]# chown -R root .
[root@bogon mysql]# chown -R mysql var
[root@bogon mysql]# chgrp -R mysql .
```

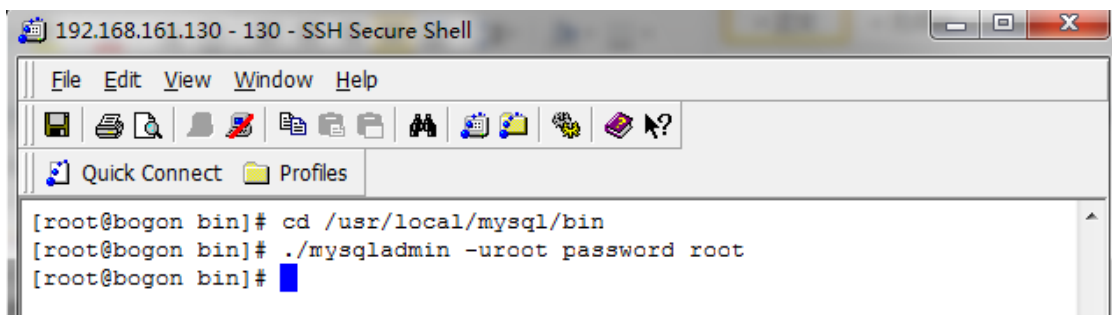
3、配置 MySQL 数据库

✚ 以后台进程方式启动 MySQL 数据库，输入如下 Shell 脚本：



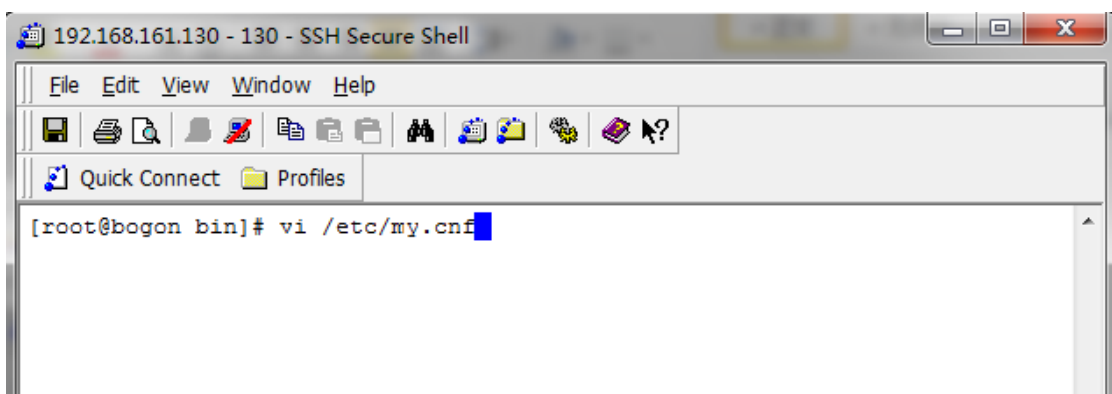
```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon mysql]# cd /usr/local/mysql/bin
[root@bogon bin]# ./mysqld_safe --user=mysql &
```

✚ 修改 MySQL 数据库 root 用户密码



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon bin]# cd /usr/local/mysql/bin
[root@bogon bin]# ./mysqladmin -uroot password root
[root@bogon bin]#
```

✚ 修改 MySQL 配置文件 my.cnf，使其忽略表名大小写

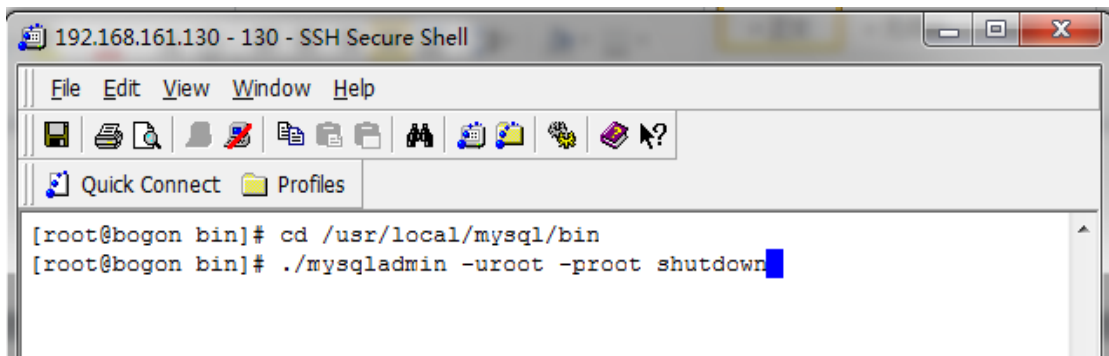


```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon bin]# vi /etc/my.cnf
```

找到[mysqld]部分，在下面添加如下内容：

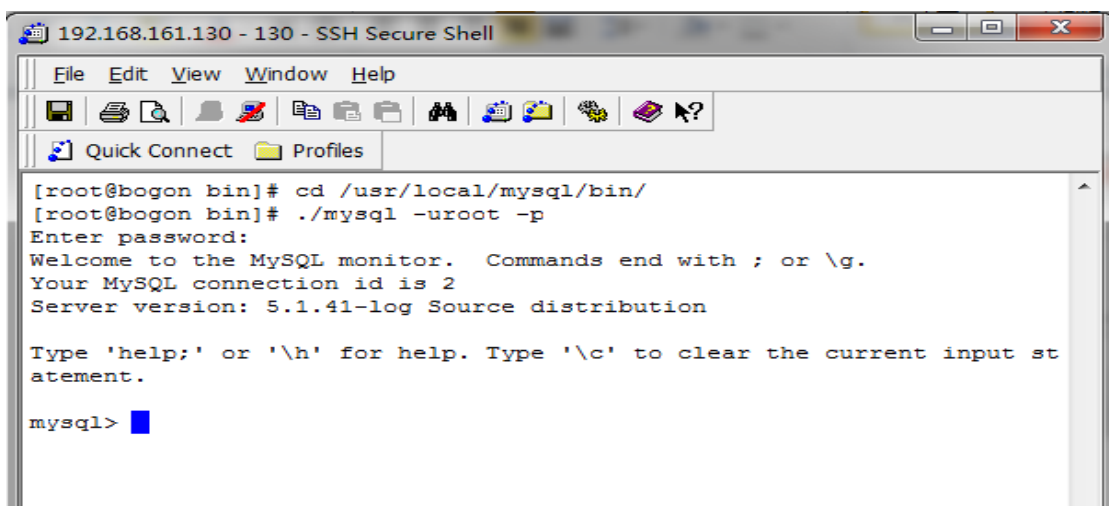
```
lower_case_table_names = 1
```

✚ 停止 MySQL 数据库进程，输入如下 Shell 脚本



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon bin]# cd /usr/local/mysql/bin
[root@bogon bin]# ./mysqladmin -uroot -proot shutdown
```

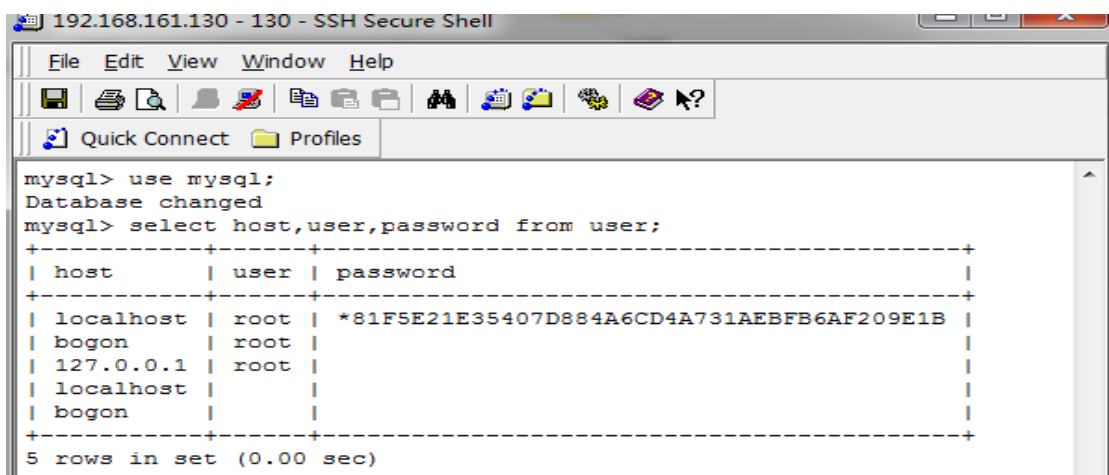
✚ 配置远程连接 MySQL 数据库（使 Windows 能够连接到 Linux 上的 MySQL 数据库）



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon bin]# cd /usr/local/mysql/bin/
[root@bogon bin]# ./mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.1.41-log Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
mysql> use mysql;
Database changed
mysql> select host,user,password from user;
+-----+-----+-----+
| host      | user  | password                                     |
+-----+-----+-----+
| localhost | root  | *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B |
| bogon     | root  |                                             |
| 127.0.0.1 | root  |                                             |
| localhost |       |                                             |
| bogon     |       |                                             |
+-----+-----+-----+
5 rows in set (0.00 sec)
```

在 SQL 窗口中输入如下 SQL 语句:

```
mysql> delete from user where host='bogon';
Query OK, 2 rows affected (0.01 sec)

mysql> delete from user where host='127.0.0.1';
Query OK, 1 row affected (0.00 sec)

mysql> delete from user where user='' and password='';
Query OK, 1 row affected (0.00 sec)
```

执行如上几条 delete 语句后, user 表中应该只有如下一条记录:

```
mysql> select host,user,password from user;
+-----+-----+-----+
| host      | user | password                                     |
+-----+-----+-----+
| localhost | root | *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

修改这条记录的 host 字段值为"%", 具体 SQL 语句如下:

```
mysql> update user set host='%' where user='root';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql>
```

修改后的最终效果如下:

```
mysql> select host,user,password from user;
+-----+-----+-----+
| host | user | password                                     |
+-----+-----+-----+
| %     | root | *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

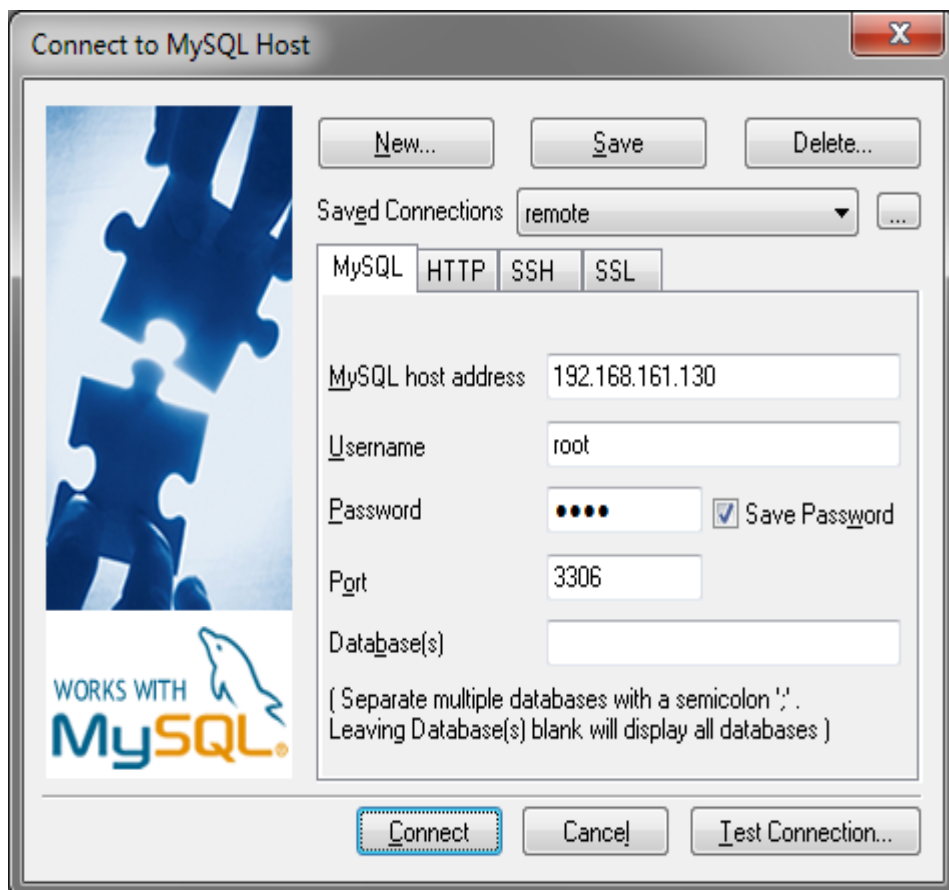

重启 MySQL 数据库，并用 Windows 连接远程 MySQL 数据库。

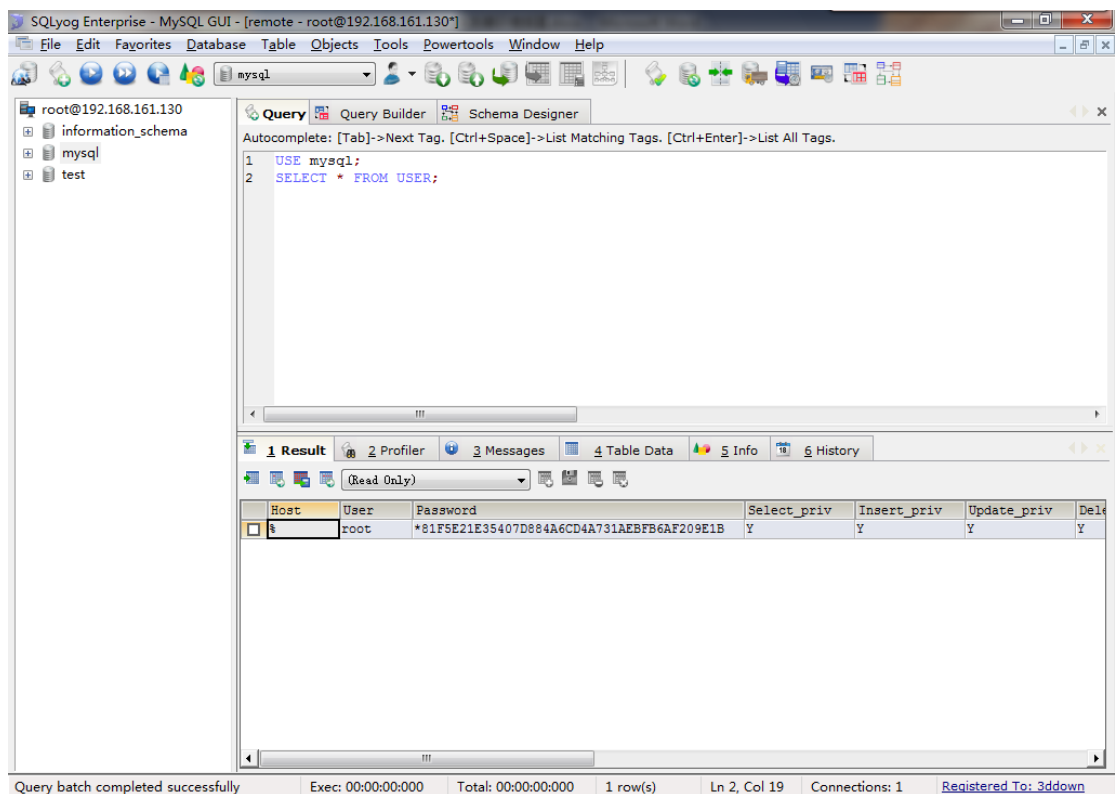
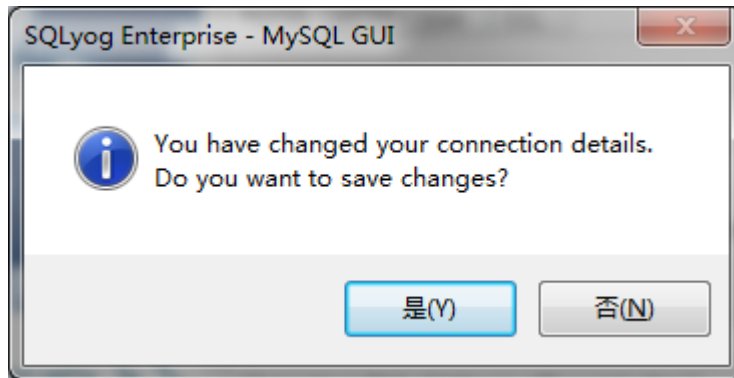
```
[root@bogon bin]# cd /usr/local/mysql/bin/
[root@bogon bin]# ./mysqladmin -uroot -proot shutdown
110130 16:52:36 mysqld_safe mysqld from pid file /usr/local/mysql/var/
bogon.pid ended
[1]+  Done                  ./mysqld_safe --user=mysql
[root@bogon bin]# ./mysqld_safe --user=mysql &
[1] 24809
[root@bogon bin]# 110130 16:52:44 mysqld_safe Logging to '/usr/local/m
ysql/var/bogon.err'.
110130 16:52:44 mysqld_safe Starting mysqld daemon with databases from
/usr/local/mysql/var

[root@bogon bin]#
```

Connected to 192.168.161.130 SSH2 - aes128-cbc - hmac-md5 70x22

Windows 远程连接：



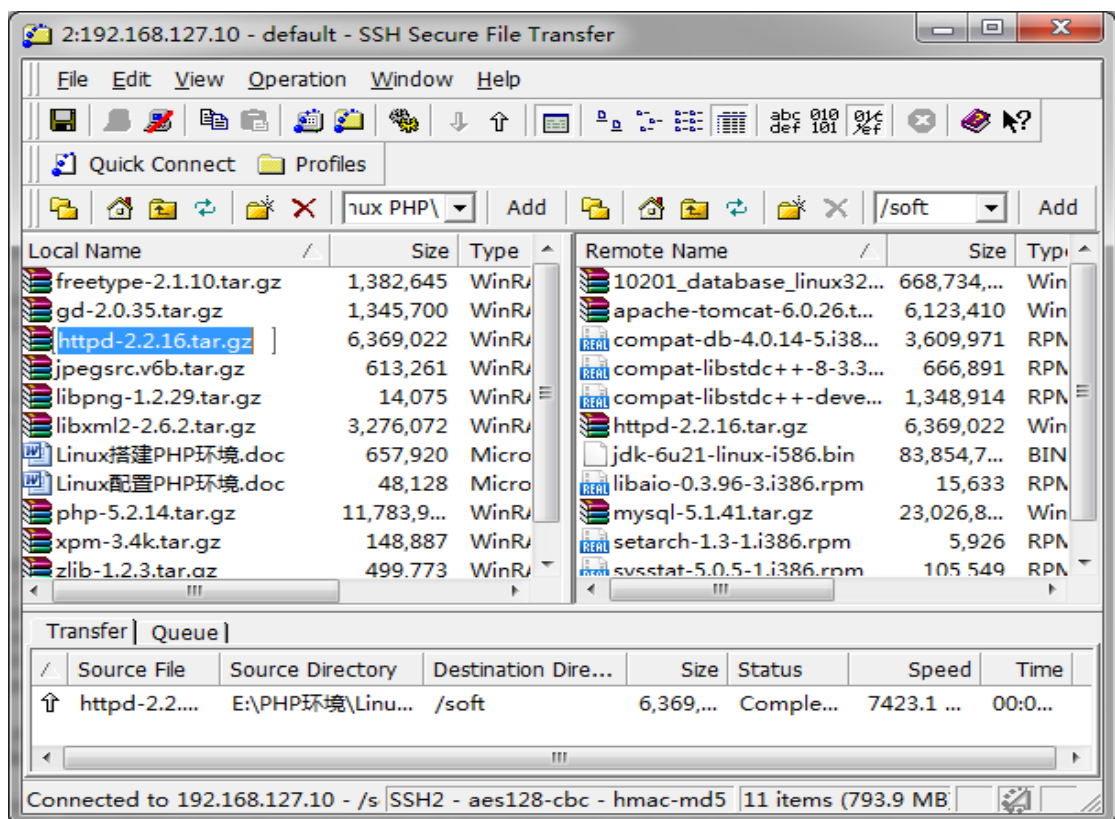


至此，MySQL 数据库就在 Linux 操作系统上安装成功了，我们也通过 Windows 操作系统成功的连接到了远程数据库，在下面的章节中，将讲解如何在 Linux 操作系统下安装 Apache 服务器。

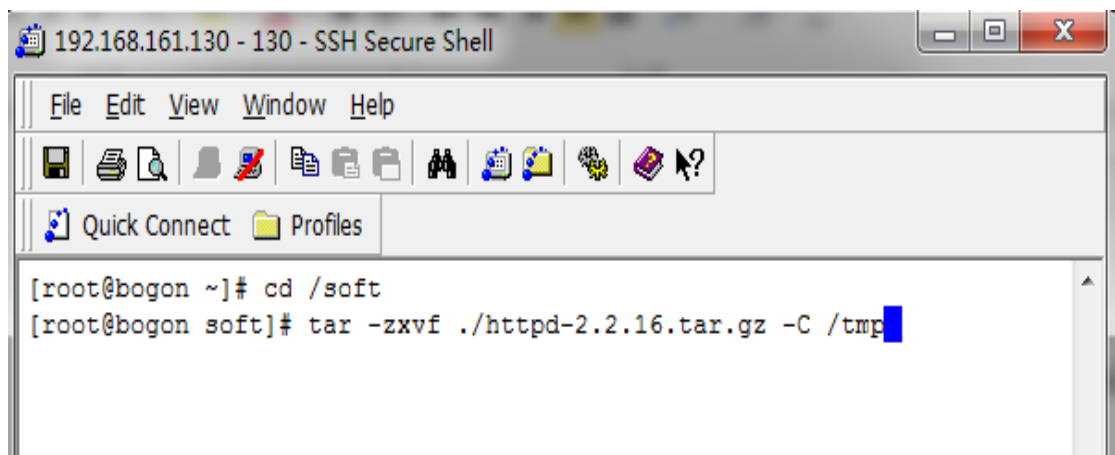
APACHE 服务器安装

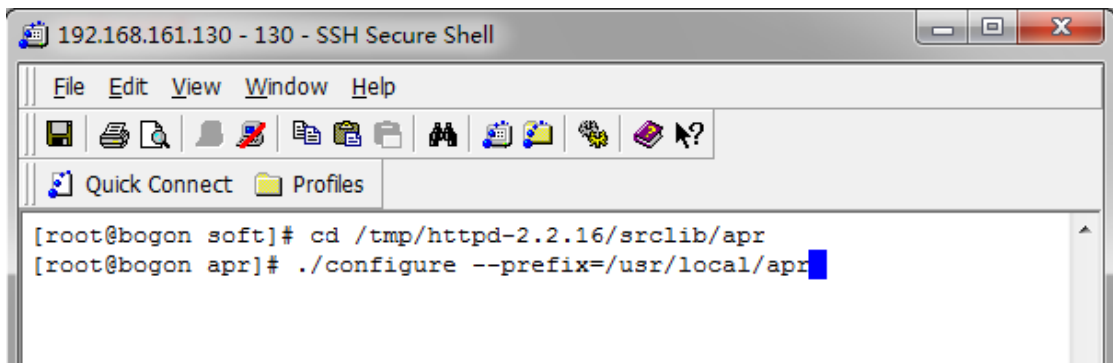
本部分将重点讲解如何在 Linux 操作系统下安装 Apache 服务器，Apache 服务器也是基于源码编译方式安装，分为三个部分：编译 apr 部分、编译 apr-util 部分和编译 Apache 服务器部分。下面分别进行讲解。

- 1、切换到 root 用户，使用 SSH 将 Apache 上传到 /soft 目录



- 2、编译 apr 部分，输入如下 Shell 脚本



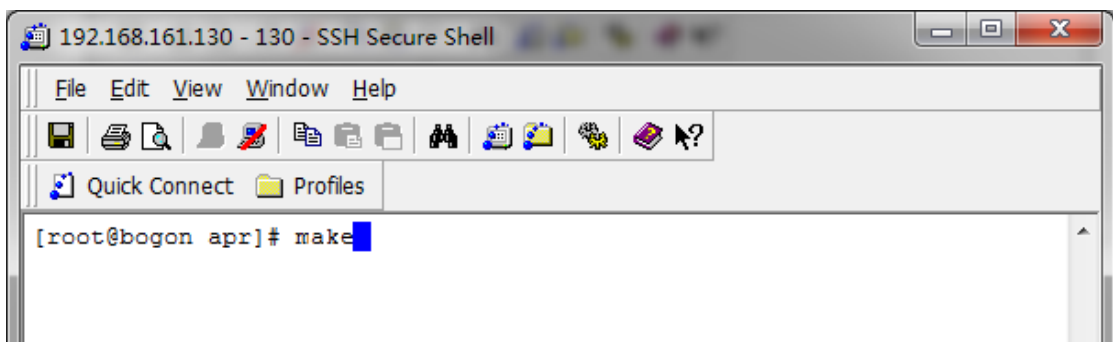


192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon soft]# cd /tmp/httpd-2.2.16/src/lib/apr
[root@bogon apr]# ./configure --prefix=/usr/local/apr
```

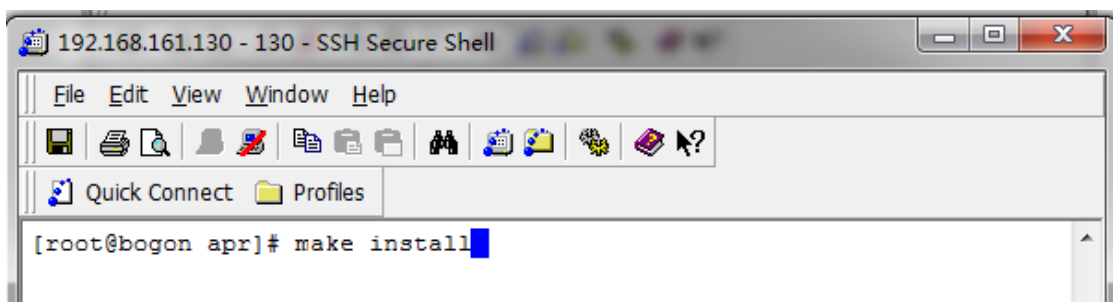


192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon apr]# make
```



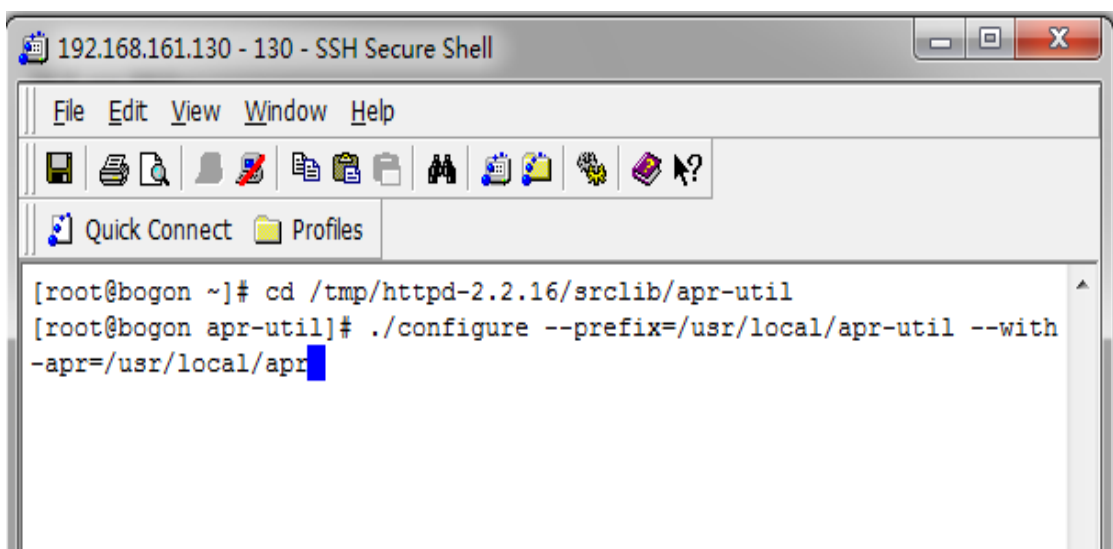
192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon apr]# make install
```

3、编译 apr-util 部分，输入如下 Shell 脚本

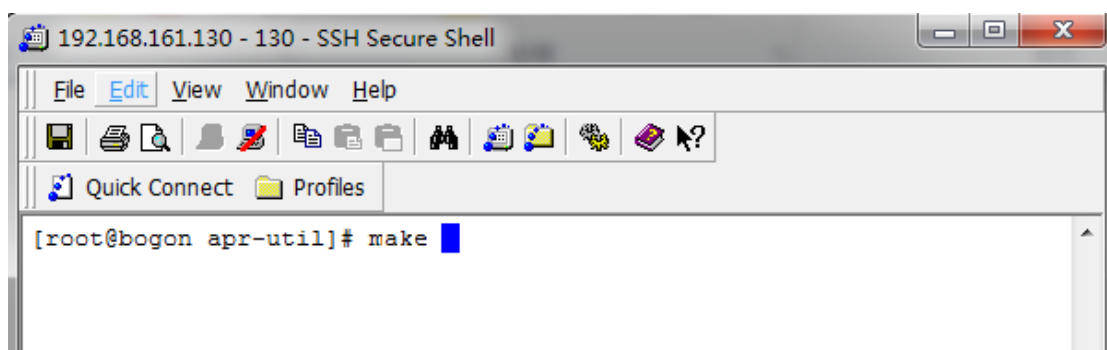


192.168.161.130 - 130 - SSH Secure Shell

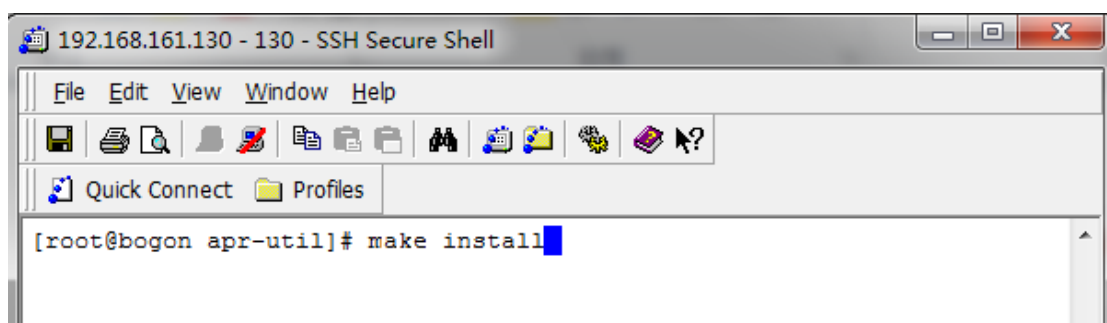
File Edit View Window Help

Quick Connect Profiles

```
[root@bogon ~]# cd /tmp/httpd-2.2.16/src/lib/apr-util
[root@bogon apr-util]# ./configure --prefix=/usr/local/apr-util --with-
-apr=/usr/local/apr
```

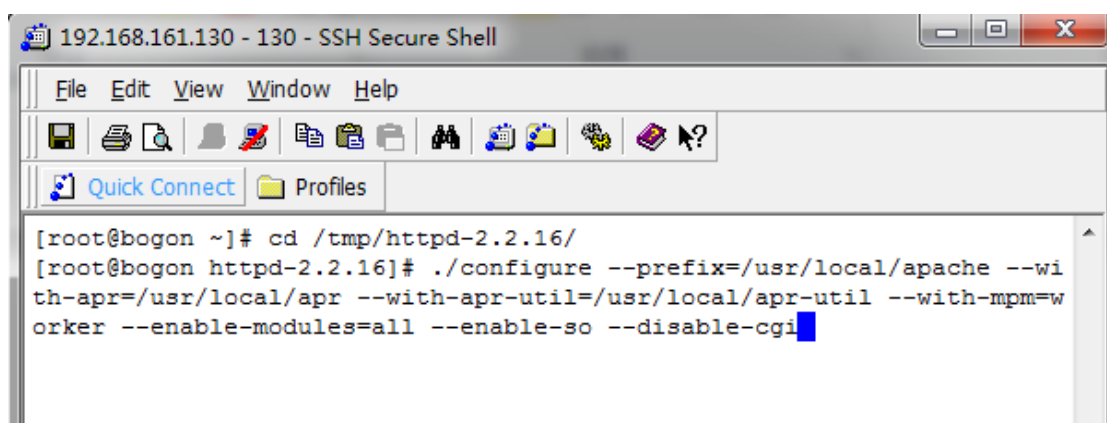


```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon apr-util]# make
```

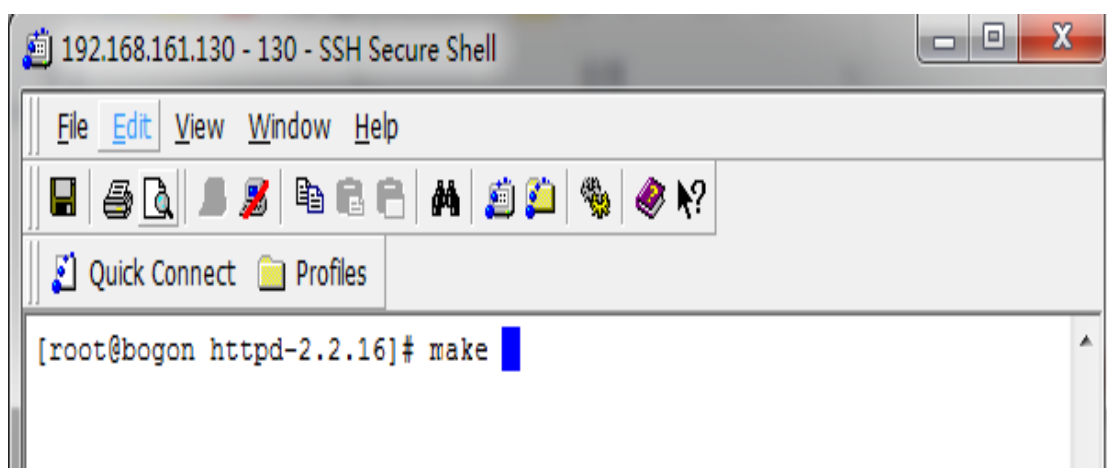


```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon apr-util]# make install
```

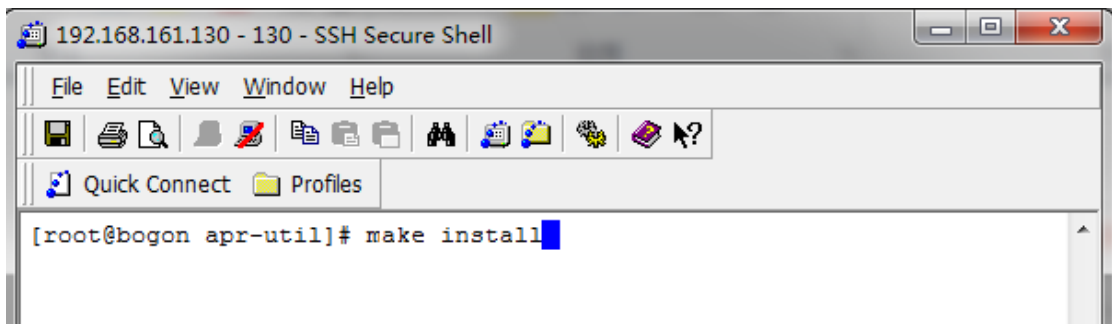
4、编译 Apache 服务器部分，输入如下 Shell 脚本



```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon ~]# cd /tmp/httpd-2.2.16/
[root@bogon httpd-2.2.16]# ./configure --prefix=/usr/local/apache --with-apr=/usr/local/apr --with-apr-util=/usr/local/apr-util --with-mpm=worker --enable-modules=all --enable-so --disable-cgi
```



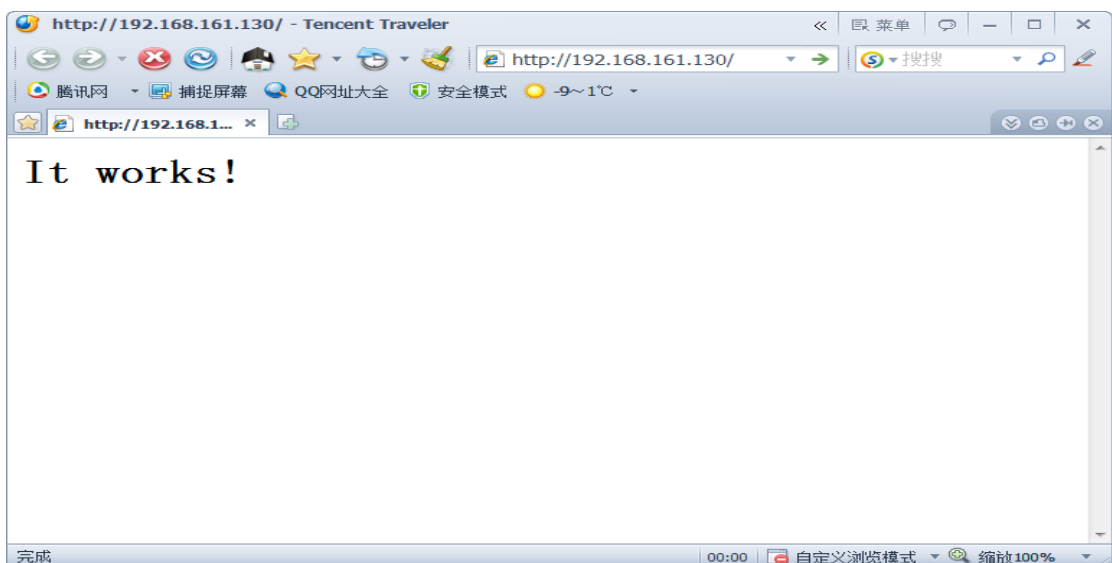
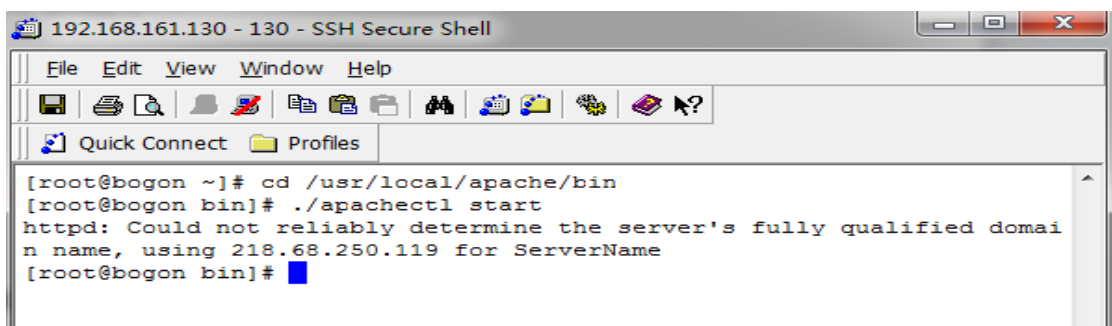
```
192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon httpd-2.2.16]# make
```



Linux 管理 Apache 常用命令：

apachectl start
apachectl stop
apachectl restart

5、回到 Windows 操作系统验证 Apache 服务器是否安装成功

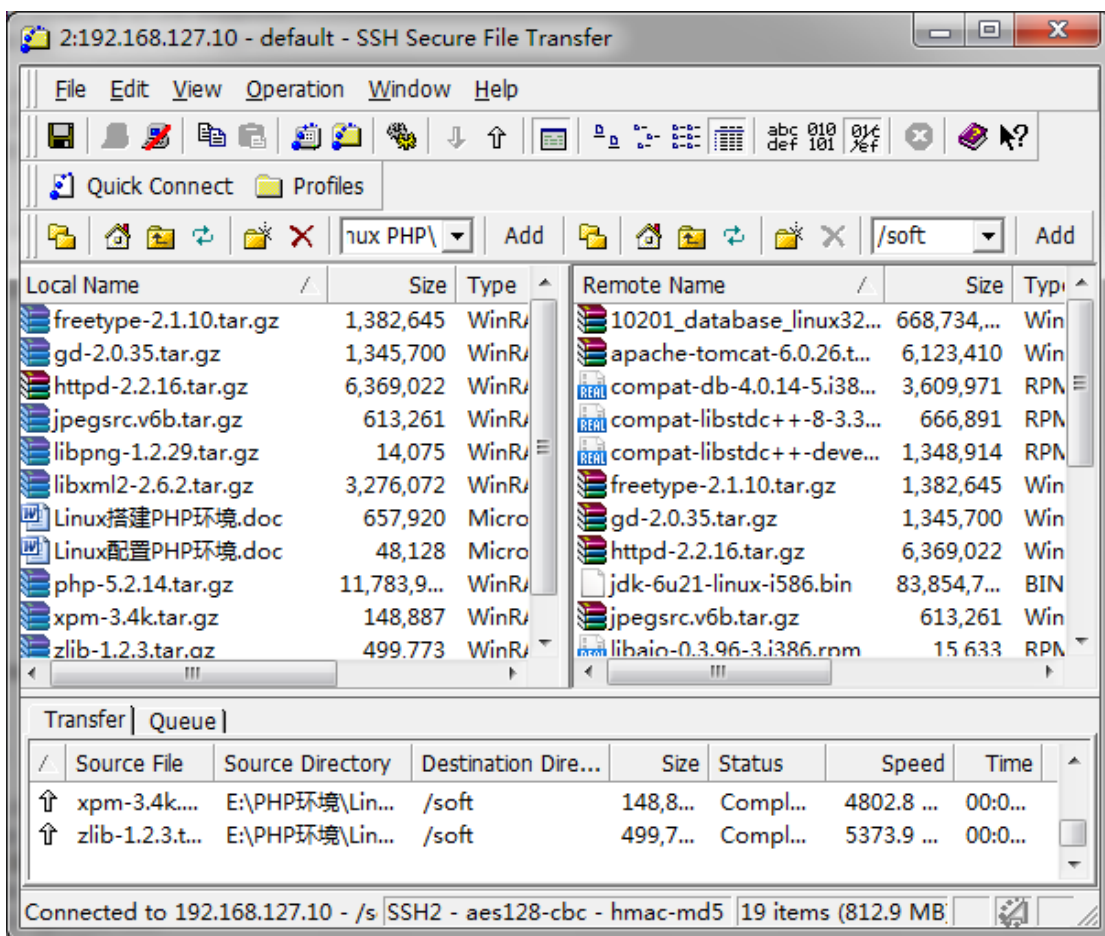


至此，我们已经成功的在 Linux 操作系统上安装了 Apache 服务器，下一章节我将重点讲解如何在 Linux 操作系统上搭建 PHP 开发环境。

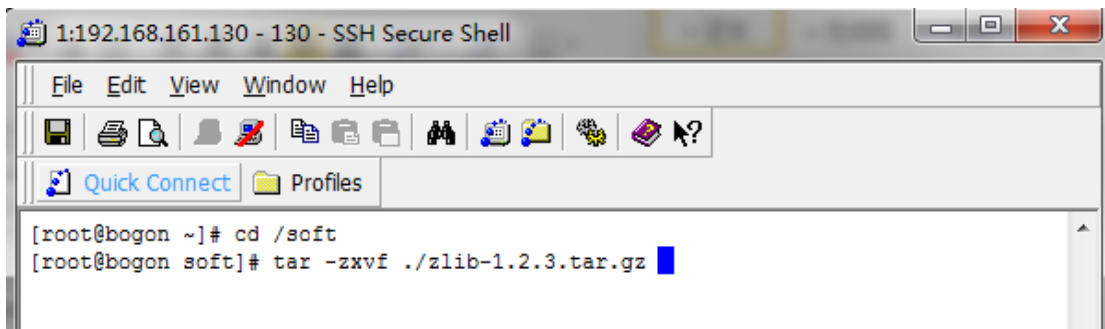
PHP 环境搭建

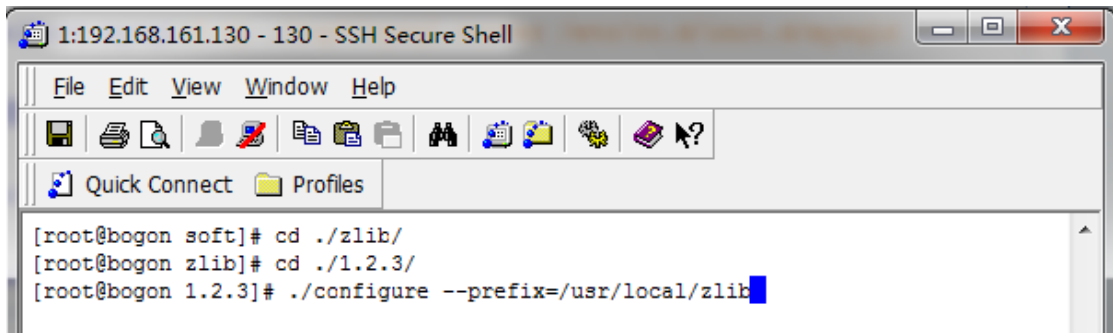
本部分将重点讲解如何在 Linux 操作系统上安装 PHP 环境, PHP 在 Windows 上直接安装就可以了, 但是在 Linux 上安装就比较麻烦了, 我们需要自己安装 GD 等一些插件。下面我讲解具体步骤。

- ✚ 将 PHP 所需环境上传到/soft 目录



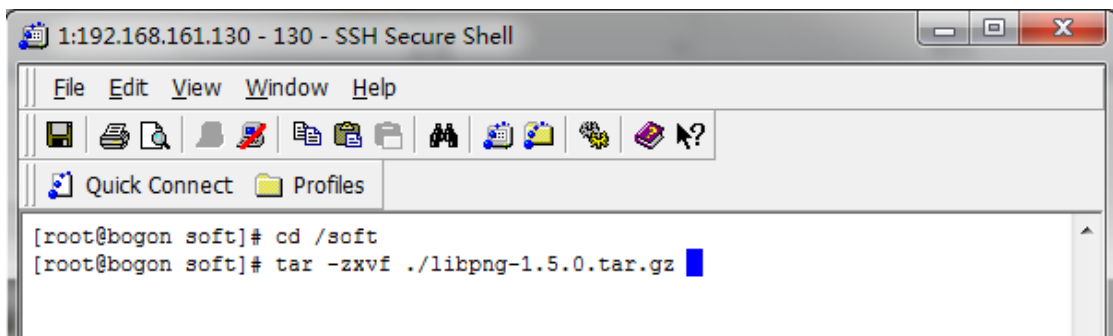
- ✚ 安装 zlib 库, 步骤如下:



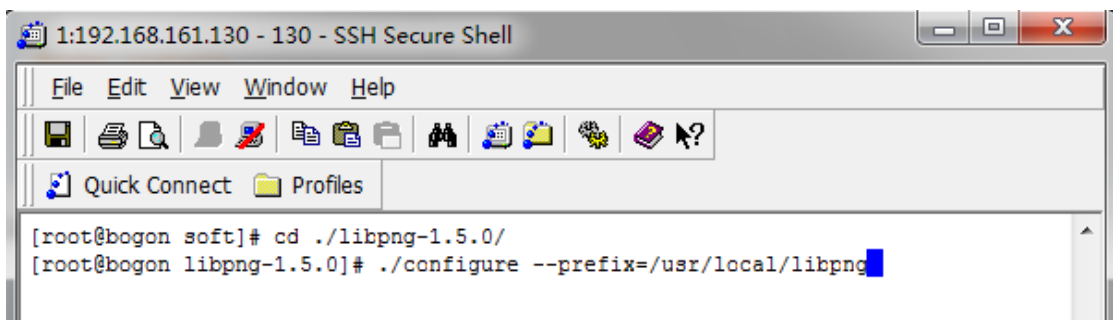


```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon soft]# cd ./zlib/
[root@bogon zlib]# cd ./1.2.3/
[root@bogon 1.2.3]# ./configure --prefix=/usr/local/zlib
```

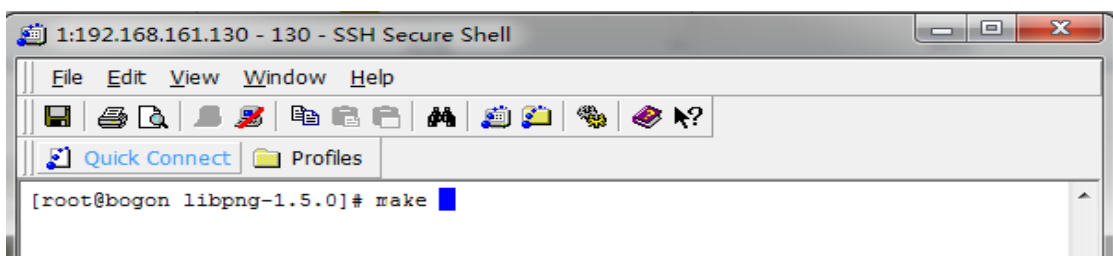
✚ 安装 libpng 库，步骤如下



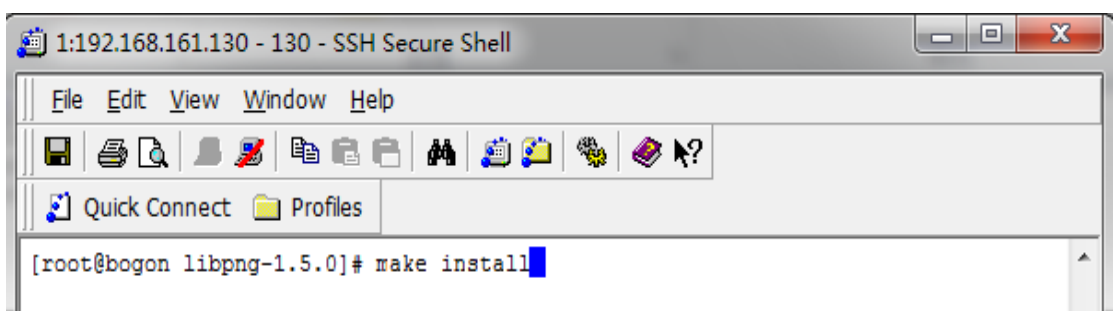
```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon soft]# cd /soft
[root@bogon soft]# tar -zxvf ./libpng-1.5.0.tar.gz
```



```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon soft]# cd ./libpng-1.5.0/
[root@bogon libpng-1.5.0]# ./configure --prefix=/usr/local/libpng
```

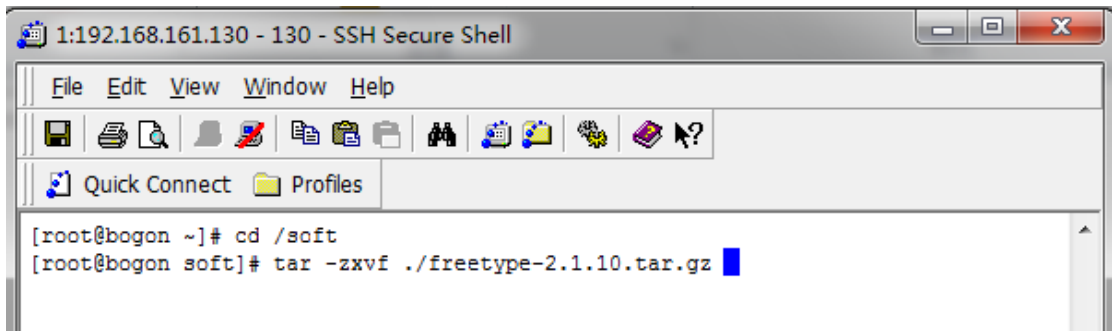


```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon libpng-1.5.0]# make
```

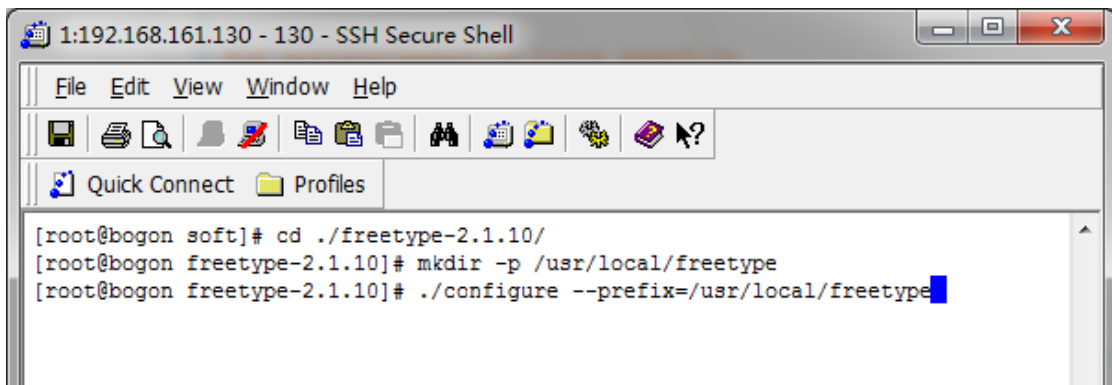


```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon libpng-1.5.0]# make install
```

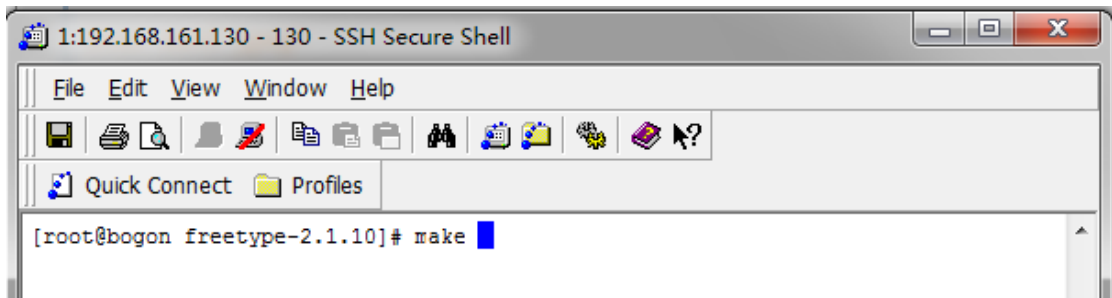
✚ 安装 freetype 库，步骤如下



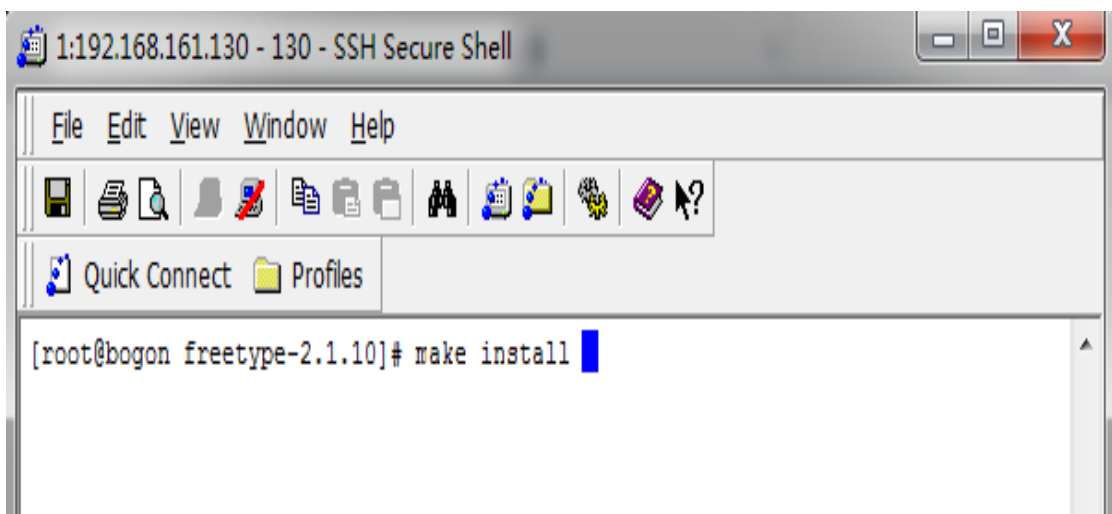
```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon ~]# cd /soft
[root@bogon soft]# tar -zxvf ./freetype-2.1.10.tar.gz
```



```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon soft]# cd ./freetype-2.1.10/
[root@bogon freetype-2.1.10]# mkdir -p /usr/local/freetype
[root@bogon freetype-2.1.10]# ./configure --prefix=/usr/local/freetype
```

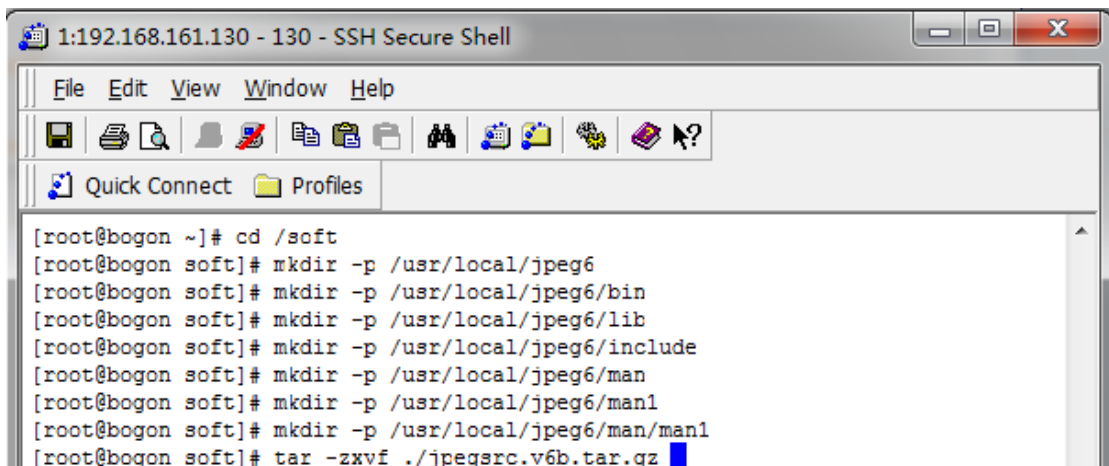


```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon freetype-2.1.10]# make
```



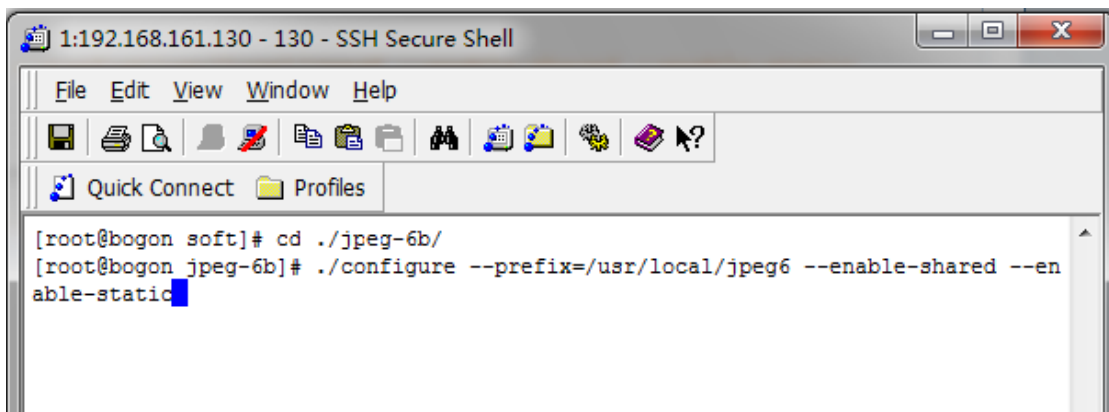
```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
[root@bogon freetype-2.1.10]# make install
```

✚ 安装 jpegsrc 库，步骤如下：



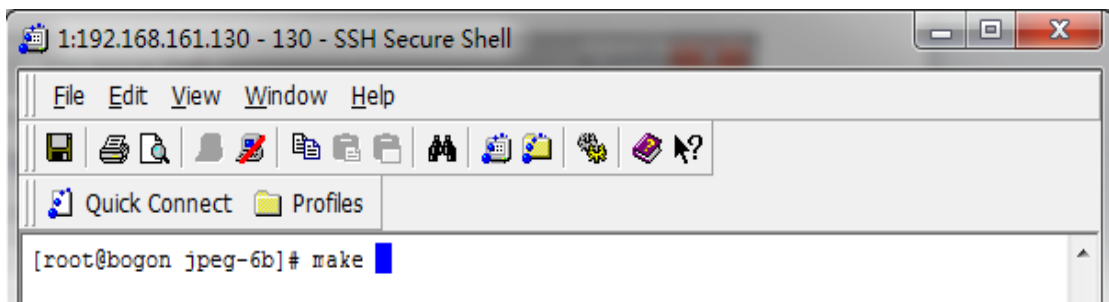
```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon ~]# cd /soft
[root@bogon soft]# mkdir -p /usr/local/jpeg6
[root@bogon soft]# mkdir -p /usr/local/jpeg6/bin
[root@bogon soft]# mkdir -p /usr/local/jpeg6/lib
[root@bogon soft]# mkdir -p /usr/local/jpeg6/include
[root@bogon soft]# mkdir -p /usr/local/jpeg6/man
[root@bogon soft]# mkdir -p /usr/local/jpeg6/man1
[root@bogon soft]# mkdir -p /usr/local/jpeg6/man/man1
[root@bogon soft]# tar -zxvf ./jpegsrc.v6b.tar.gz
```



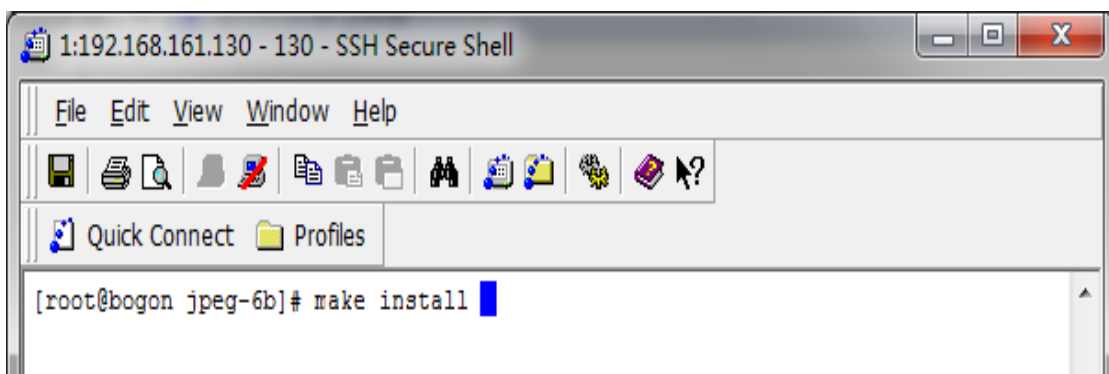
```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon soft]# cd ./jpeg-6b/
[root@bogon jpeg-6b]# ./configure --prefix=/usr/local/jpeg6 --enable-shared --enable-static
```



```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon jpeg-6b]# make
```



```
1:192.168.161.130 - 130 - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles

[root@bogon jpeg-6b]# make install
```

```
[root@bogon jpeg-6b]# make install-lib
```

Connected to 192.168.161.130 SSH2 - aes128-cbc - hmac-md5 80x24

✚ 安装 libxml 库，步骤如下：

```
1:192.168.161.130 - 130 - SSH Secure Shell
```

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon soft]# cd /soft
[root@bogon soft]# mkdir -p /usr/local/libxml2
[root@bogon soft]#
```

```
1:192.168.161.130 - 130 - SSH Secure Shell
```

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon soft]# tar -zxvf ./libxml2-2.6.2.tar.gz
```

```
1:192.168.161.130 - 130 - SSH Secure Shell
```

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon soft]# cd ./libxml2-2.6.2/
[root@bogon libxml2-2.6.2]# ./configure --prefix=/usr/local/libxml2
```

```
1:192.168.161.130 - 130 - SSH Secure Shell
```

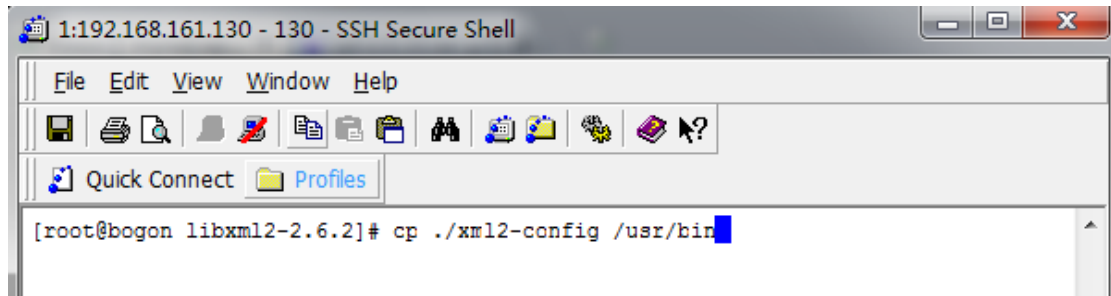
File Edit View Window Help

Quick Connect Profiles

```
[root@bogon libxml2-2.6.2]# make
```

```
[root@bogon libxml2-2.6.2]# make install
```

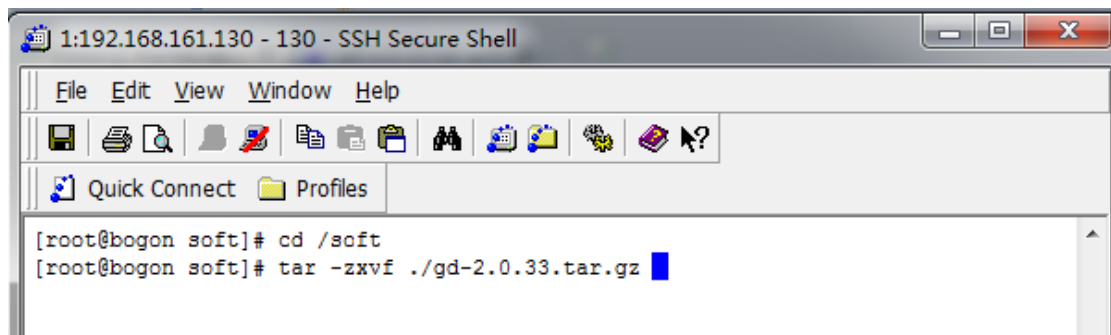
Connected to 192.168.161.130 SSH2 - aes128-cbc - hmac-md5 80x24



A terminal window titled "1:192.168.161.130 - 130 - SSH Secure Shell". The window has a menu bar with "File", "Edit", "View", "Window", and "Help". Below the menu bar is a toolbar with various icons. The terminal shows the command `cp ./xml2-config /usr/bin` being executed in a shell where the prompt is `[root@bogon libxml2-2.6.2]`.

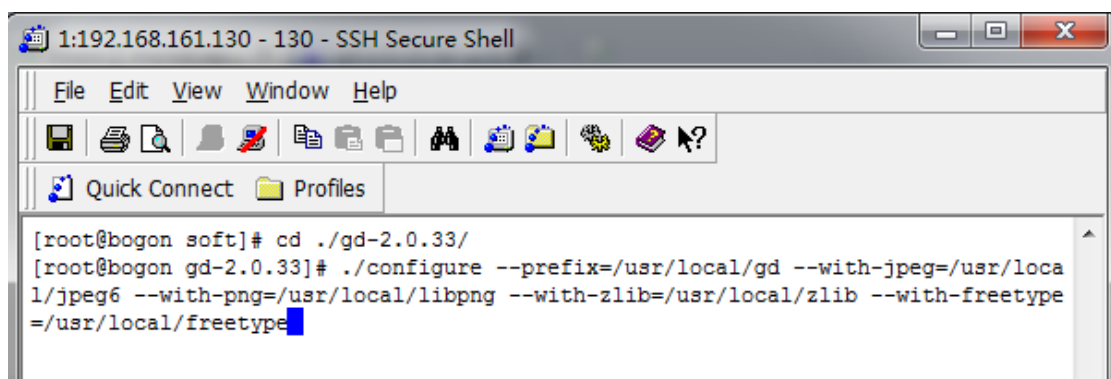
```
[root@bogon libxml2-2.6.2]# cp ./xml2-config /usr/bin
```

✚ 安装 gd 库，具体步骤如下：



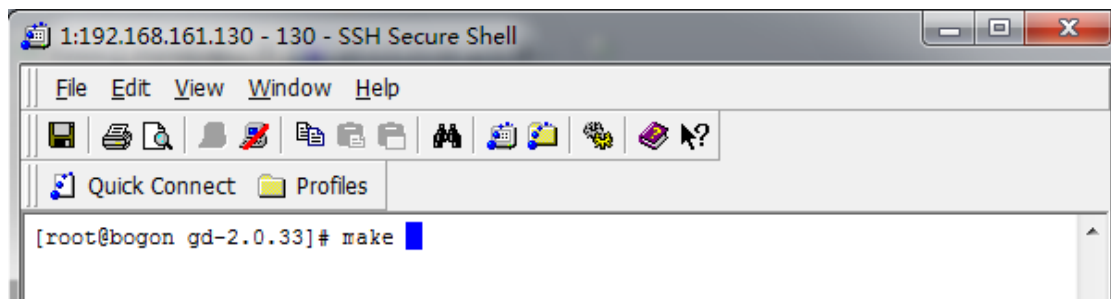
A terminal window titled "1:192.168.161.130 - 130 - SSH Secure Shell". The window has a menu bar with "File", "Edit", "View", "Window", and "Help". Below the menu bar is a toolbar with various icons. The terminal shows the commands `cd /soft` and `tar -zxvf ./gd-2.0.33.tar.gz` being executed in a shell where the prompt is `[root@bogon soft]`.

```
[root@bogon soft]# cd /soft
[root@bogon soft]# tar -zxvf ./gd-2.0.33.tar.gz
```



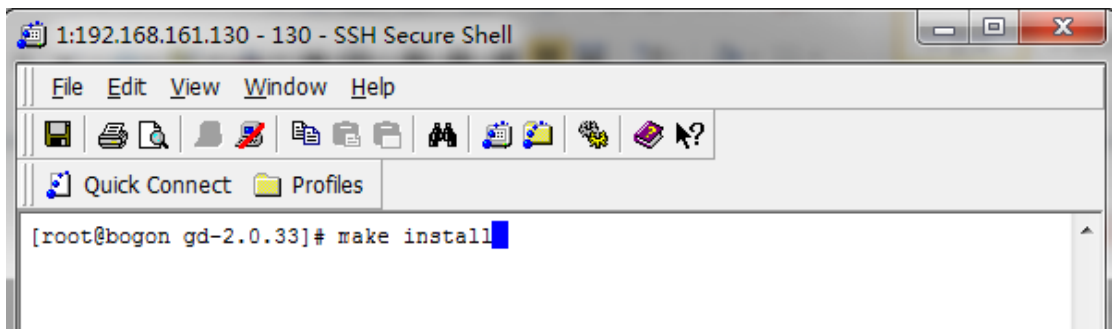
A terminal window titled "1:192.168.161.130 - 130 - SSH Secure Shell". The window has a menu bar with "File", "Edit", "View", "Window", and "Help". Below the menu bar is a toolbar with various icons. The terminal shows the commands `cd ./gd-2.0.33/` and `./configure --prefix=/usr/local/gd --with-jpeg=/usr/local/jpeg6 --with-png=/usr/local/libpng --with-zlib=/usr/local/zlib --with-freetype=/usr/local/freetype` being executed in a shell where the prompt is `[root@bogon gd-2.0.33]`.

```
[root@bogon soft]# cd ./gd-2.0.33/
[root@bogon gd-2.0.33]# ./configure --prefix=/usr/local/gd --with-jpeg=/usr/local/jpeg6 --with-png=/usr/local/libpng --with-zlib=/usr/local/zlib --with-freetype=/usr/local/freetype
```



A terminal window titled "1:192.168.161.130 - 130 - SSH Secure Shell". The window has a menu bar with "File", "Edit", "View", "Window", and "Help". Below the menu bar is a toolbar with various icons. The terminal shows the command `make` being executed in a shell where the prompt is `[root@bogon gd-2.0.33]`.

```
[root@bogon gd-2.0.33]# make
```



注意：如果在安装 **gd** 库时报错，请按如下步骤解决：

(1) 安装 **gettext** 库

```
cp /usr/lib/libattr.* /lib/  
tar xzf gettext-0.17.tar.gz  
cd gettext-0.17  
./configure  
make  
make install
```

先安装 **gettext** 库后，再编译 **gd** 库。

(2) 在 **gd** 的 **configure** 后，修改 **Makefile** 文件

将

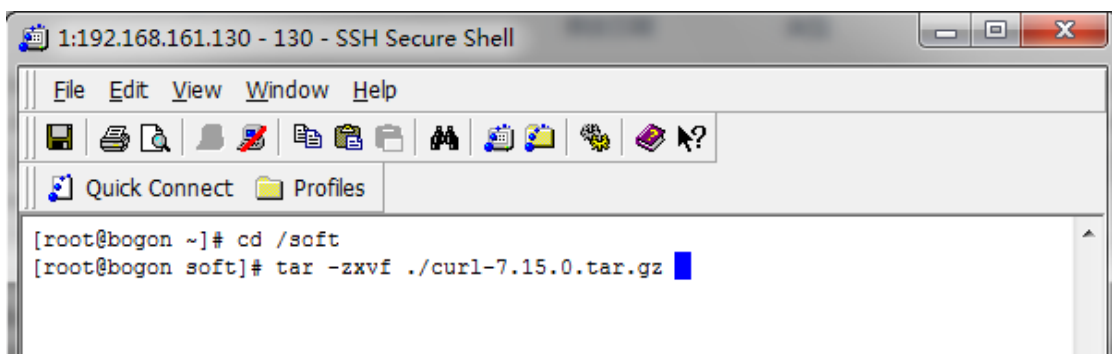
```
CPPFLAGS    =    -I/usr/local/freetype/include/freetype2    -I/usr/local/freetype/include  
-I/usr/local/freetype//include    -I/usr/local/jpeg6//include
```

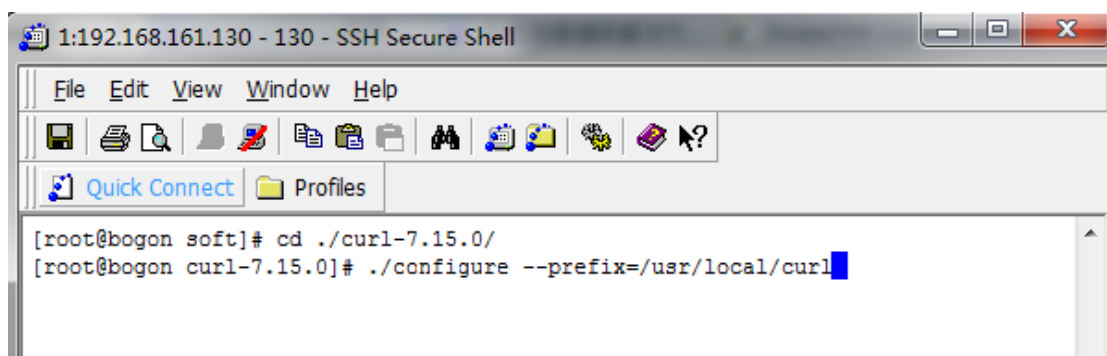
修改为

```
CPPFLAGS    =    -I/usr/local/freetype/include/freetype2    -I/usr/local/freetype/include  
-I/usr/local/freetype//include    -I/usr/local/jpeg6//include -I/usr/local/libpng//include
```

修改后在 **make**、**make install**

✚ 安装 **curl** 库，具体步骤如下：



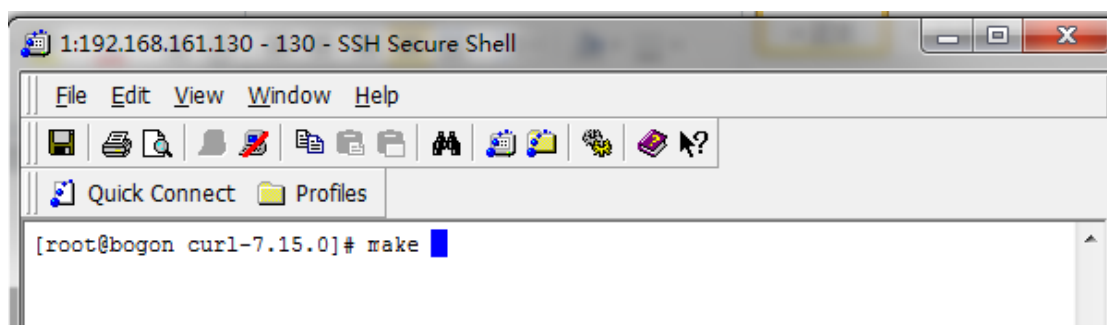


1:192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon soft]# cd ./curl-7.15.0/  
[root@bogon curl-7.15.0]# ./configure --prefix=/usr/local/curl
```

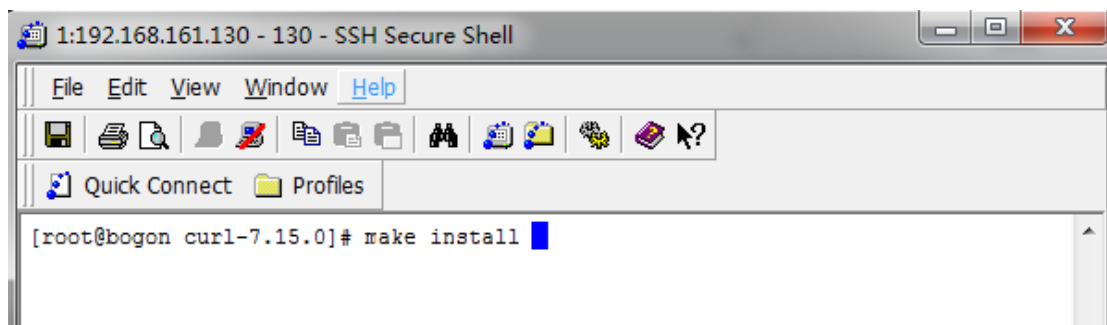


1:192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon curl-7.15.0]# make
```



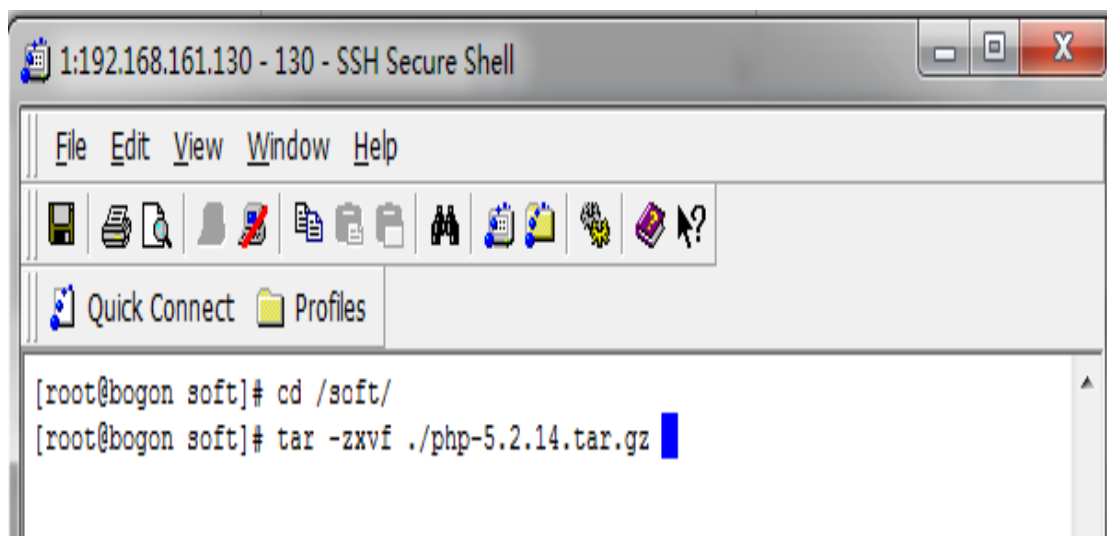
1:192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon curl-7.15.0]# make install
```

✚ 安装 PHP5，具体步骤如下：



1:192.168.161.130 - 130 - SSH Secure Shell

File Edit View Window Help

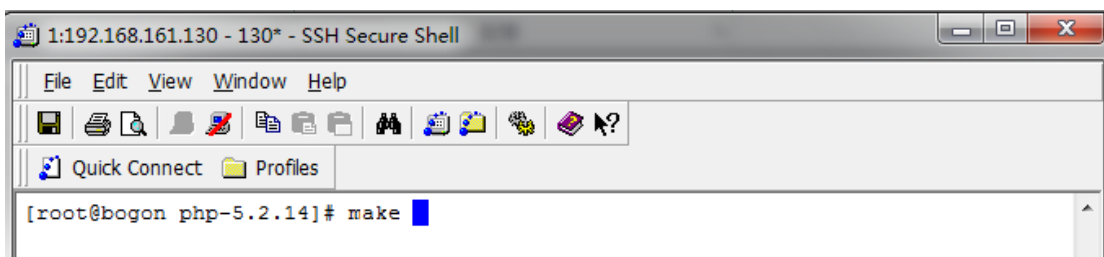
Quick Connect Profiles

```
[root@bogon soft]# cd /soft/  
[root@bogon soft]# tar -zxvf ./php-5.2.14.tar.gz
```



```
[root@bogon php-5.2.14]# ./configure --prefix=/usr/local/php5 --with-apxs2=/usr/local/apache/bin/apxs --with-jpeg-dir=/usr/local/jpeg6/ --with-png-dir=/usr/local/libpng/ --with-gd=/usr/local/gd/ --with-freetype-dir=/usr/local/freetype/ --with-zlib-dir=/usr/local/zlib/ --with-mysql=/usr/local/mysql --with-libxml-dir=/usr/local/libxml2/ --enable-track-vars --enable-mbstring=all
```

Connected to 192.168.161.130 SSH2 - aes128-cbc - hmac-md5 80x24 NUM

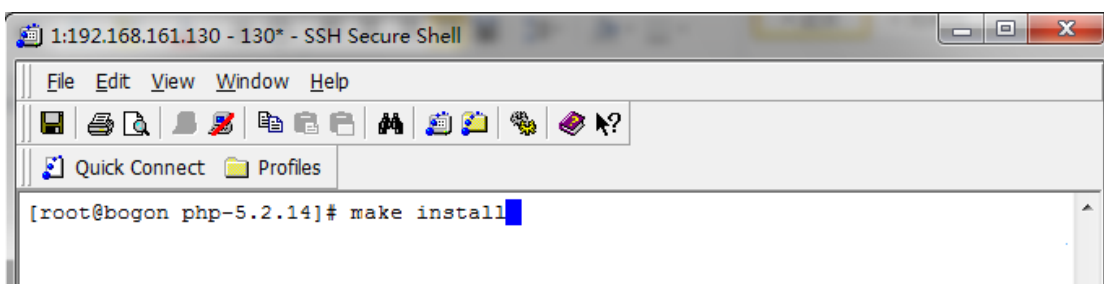


1:192.168.161.130 - 130* - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon php-5.2.14]# make
```

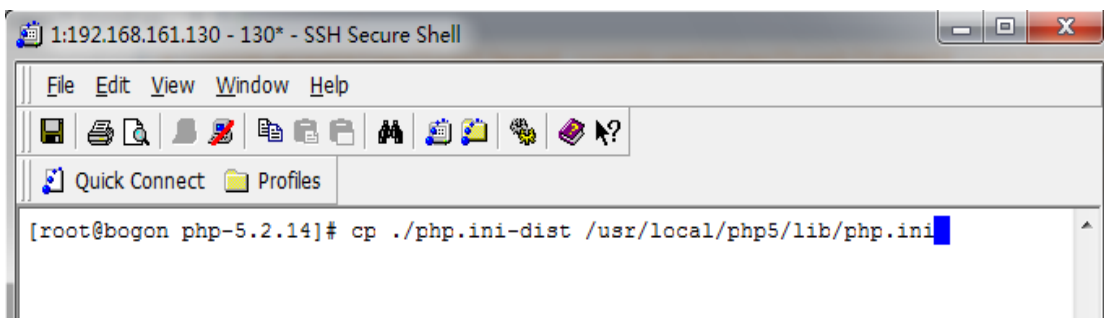


1:192.168.161.130 - 130* - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon php-5.2.14]# make install
```



1:192.168.161.130 - 130* - SSH Secure Shell

File Edit View Window Help

Quick Connect Profiles

```
[root@bogon php-5.2.14]# cp ./php.ini-dist /usr/local/php5/lib/php.ini
```

使用 vi 编辑器编辑 Apache 服务器端 httpd.conf 文件

在 297 #AddType application/x-gzip .tgz 下添加如下内容:

```
298 AddType application/x-httpd-php .php
```

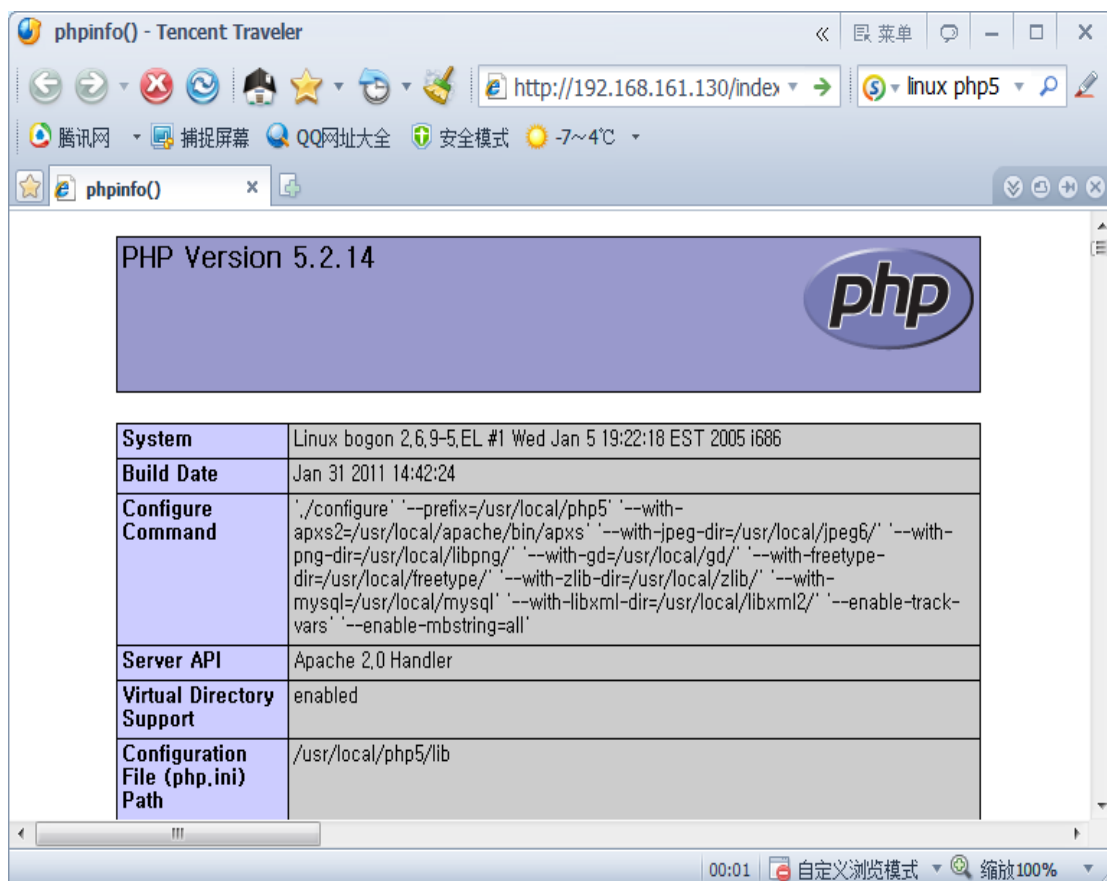
确认 53 LoadModule php5_module modules/libphp5.so 此

句前面没有#，如果有#，将其去掉。

```
166 <IfModule dir_module>
167     DirectoryIndex index.html
找到 168 </IfModule>，修改为
```

```
166 <IfModule dir_module>
167     DirectoryIndex index.html index.php
168 </IfModule>
```

编写一个 php 页面测试 PHP 环境是否成功搭建。

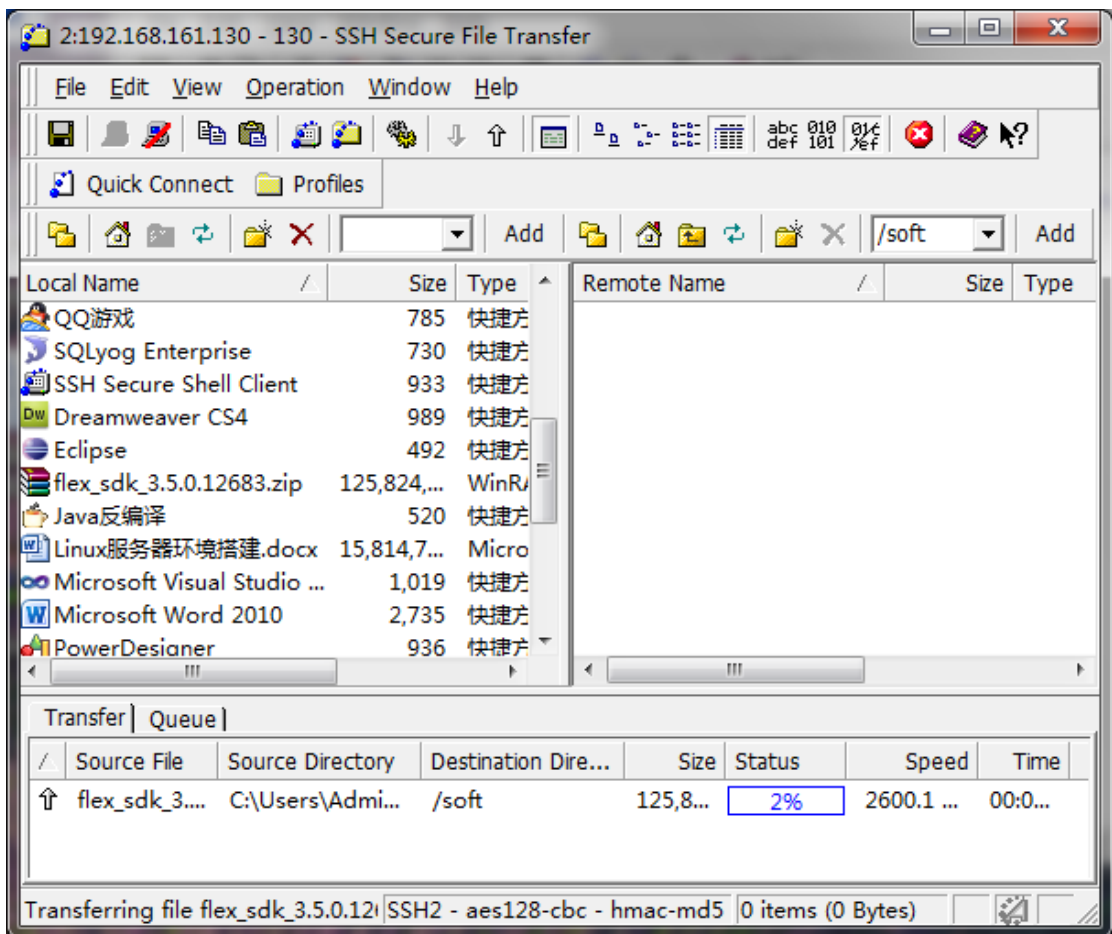


至此，我们的 PHP 环境就算真正的搭建完成了，大家仔细看后，就是在安装 gd 的时候会有一些问题出现，其他的没什么，在最后一章节中我将讲解如何在 Linux 下搭建 Flex 开发环境。

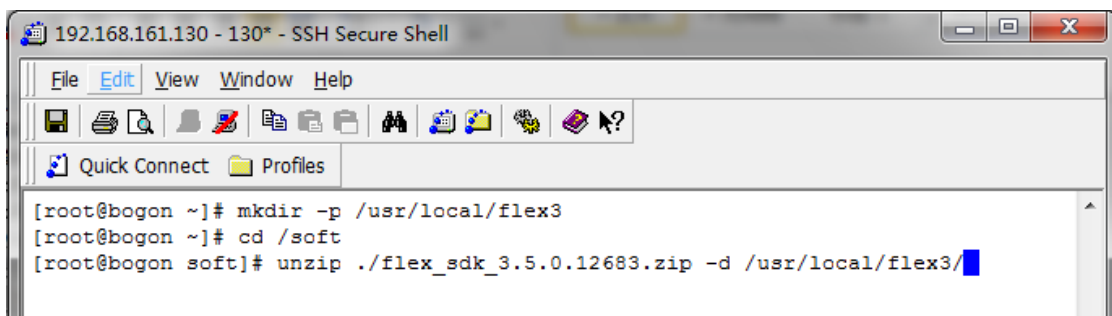
FLEX 开发环境搭建

本部分是整个文档的最后一部分了,这部分主要讲解如何在 Linux 下搭建 Flex 开发环境, Flex 是 Adobe 公司的产品,是一个全新的 RIA 技术,下面我就来讲解一下如何在 Linux 下搭建 Flex 开发环境。

✚ 将 flex-sdk 上传至 Linux 服务器



✚ 键入如下 Shell 脚本:

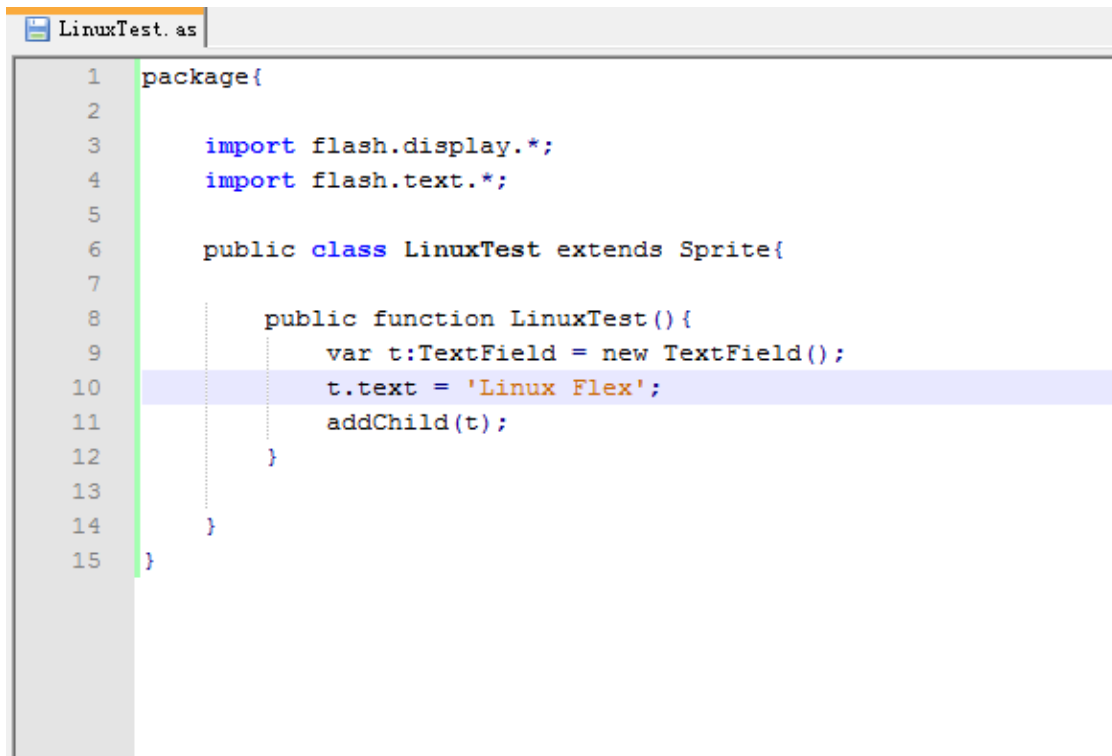


✚ 配置 Flex 环境变量

用 vi 编辑器编辑 ~/.bashrc 文件，添加如下内容：

```
export PATH=$PATH:/usr/local/flex3/bin
```

✚ 编写一个 ActionScript 文件测试，Flex 环境是否搭建成功

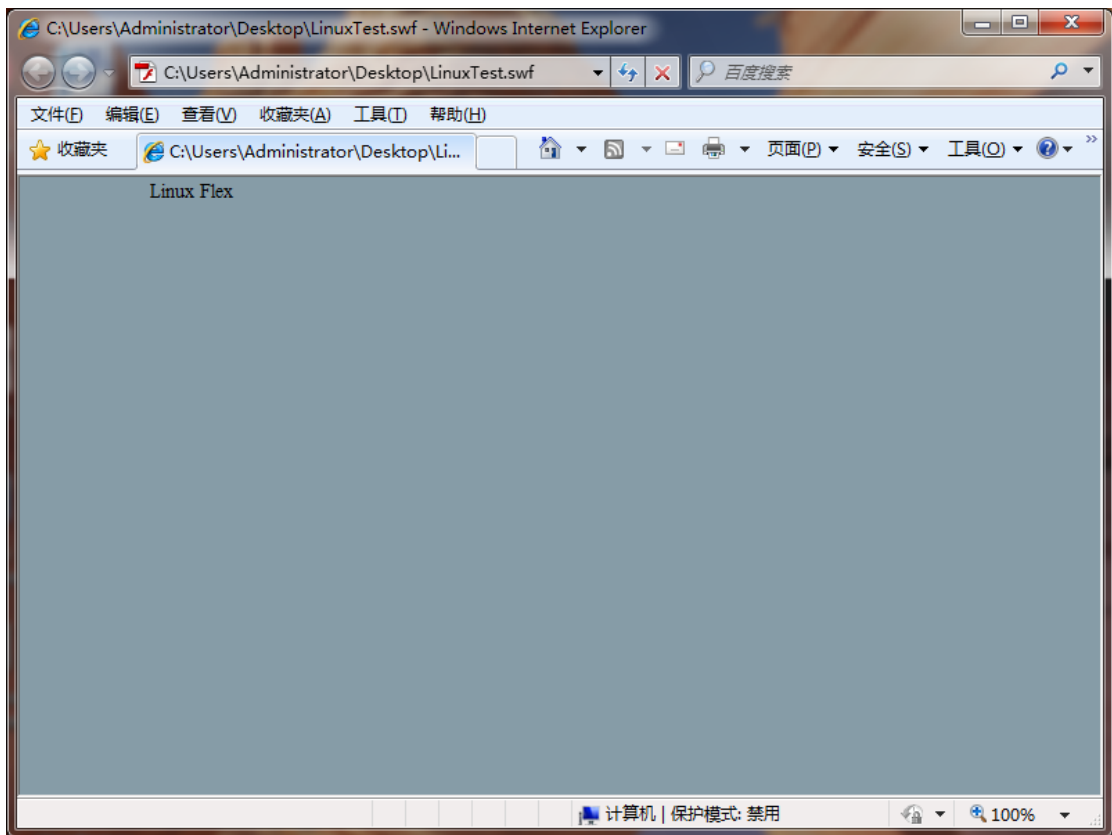
A screenshot of a text editor window titled 'LinuxTest.as'. The editor shows the following ActionScript code:

```
1 package{
2
3     import flash.display.*;
4     import flash.text.*;
5
6     public class LinuxTest extends Sprite{
7
8         public function LinuxTest(){
9             var t:TextField = new TextField();
10            t.text = 'Linux Flex';
11            addChild(t);
12        }
13    }
14 }
15 }
```

编译 ActionScript 程序：

```
[root@bogon /]# mxmclc LinuxTest.as
Loading configuration file /usr/local/flex3/frameworks/flex-config.xml
/LinuxTest.swf (616 bytes)
[root@bogon /]#
```

将编译好后的 Flex 程序放入 Windows 下，将 swf 放入浏览器测试。



至此，我们已经大功告成，完成了所有的配置工作，已经成功的在 Linux 上搭建了各种常用的开发环境。在写这份文档的过程中自己也遇到了很多困难，也经过了很多次失败，希望这份文档对大家有所帮助，我的联系方式：QQ (176125727)，Email: realliuchu@126.com。欢迎各位同仁拍砖。