# 下载可编译的linux0.11的源代码

# 在vmware中安装redhat9.0，并制作docker镜像（该步骤可以上略，直接使用docker pull yy20190811/rh9:1.00）

## 下载redhat9.0的安装盘

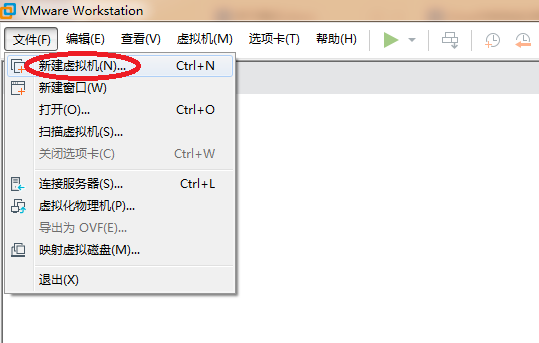
到百度搜索下载redhat9.0（个人版， 是一个很古老的版本，早就没有维护了。现在新的都是linux企业版rhel）

下载后，有下面3个iso文件

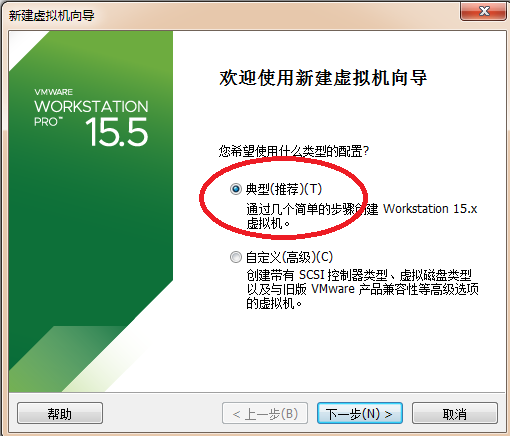


## 新建虚拟机

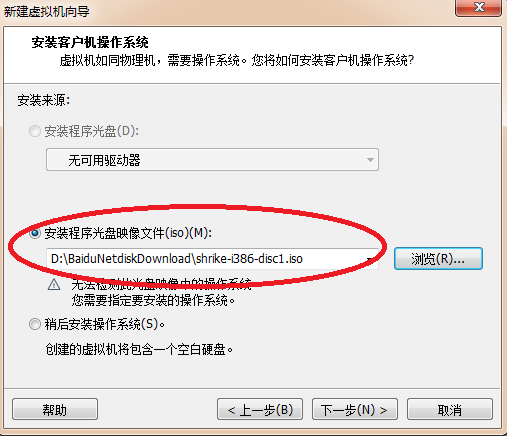
### 点击vmware的[文件][新建虚拟机]菜单



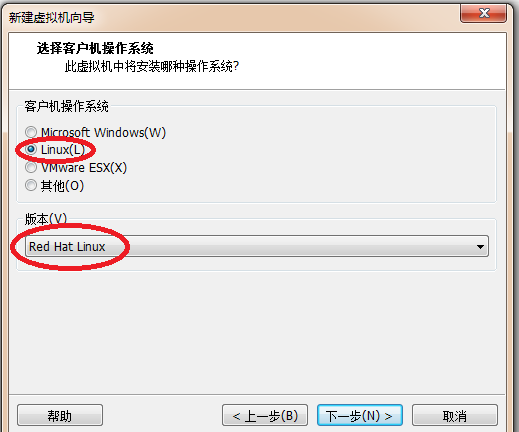
### 选择“典型”配置



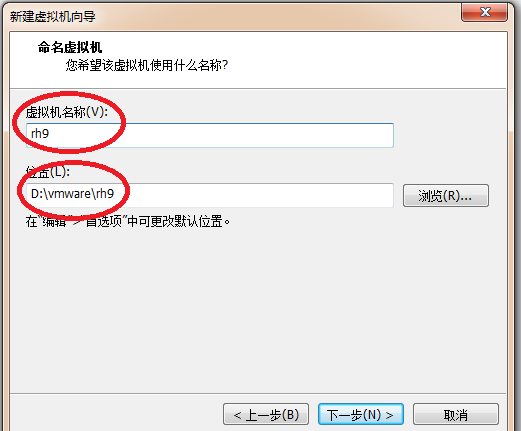
### 设置安装光盘：shrike-i386-disc1.iso（选择第一张光盘）



### 设置操作系统

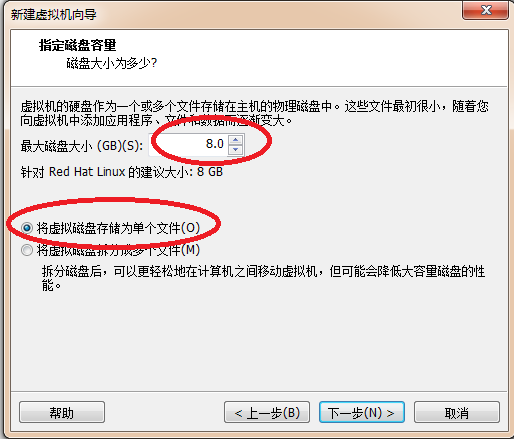


### 设置虚拟机名称和存储位置

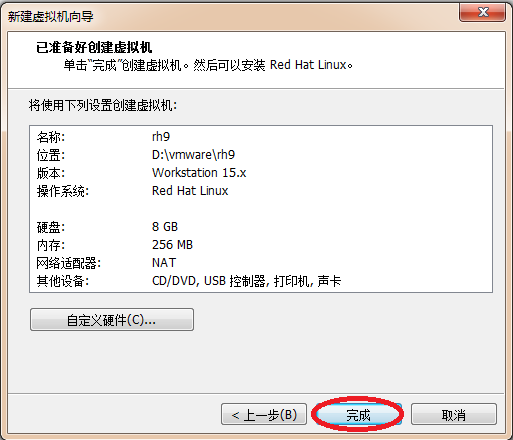


### 设置系统的硬盘

这里只做linux0.11的编译环境，可以设置一个比较小的硬盘， 设置单个的硬盘文件



### 完成虚拟机的新建

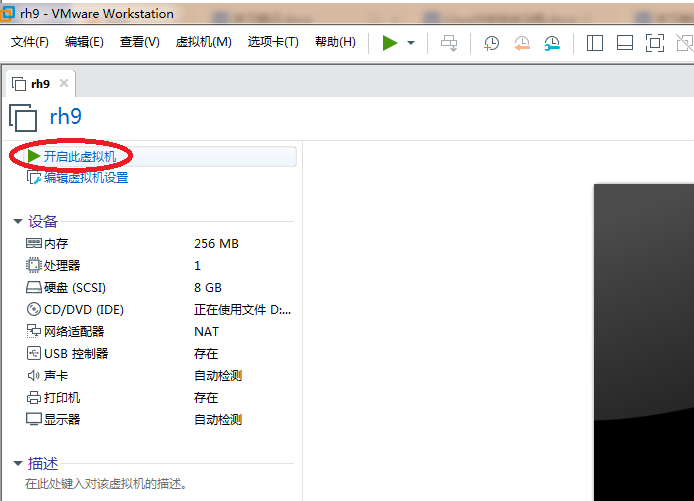


### 创建的虚拟机如下

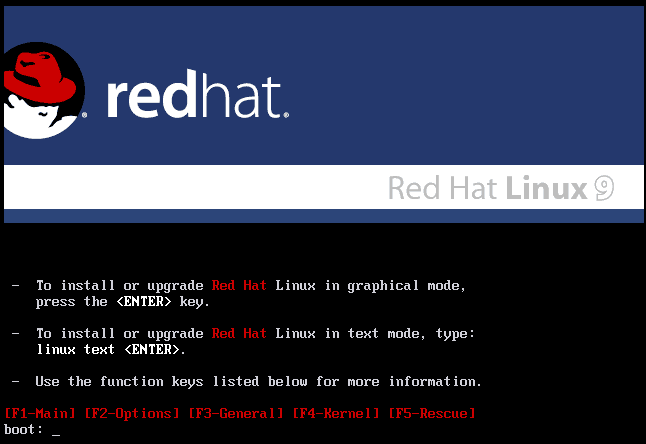


## 在新建的虚拟机中安装redhat9.0

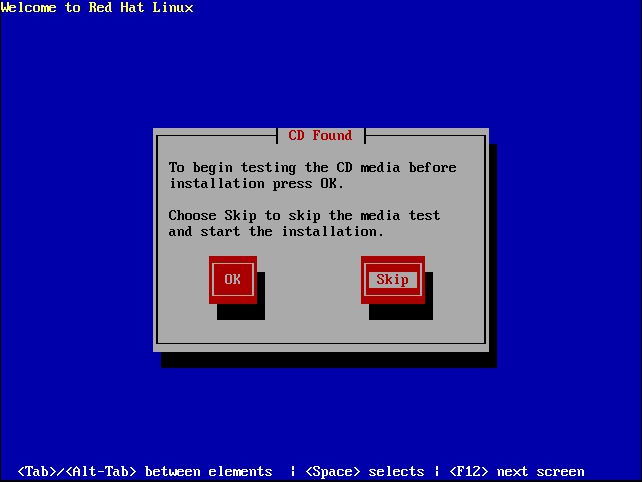
### 启动虚拟机



### 等会出现下面的样子，直接按回车， 选择图形化安装



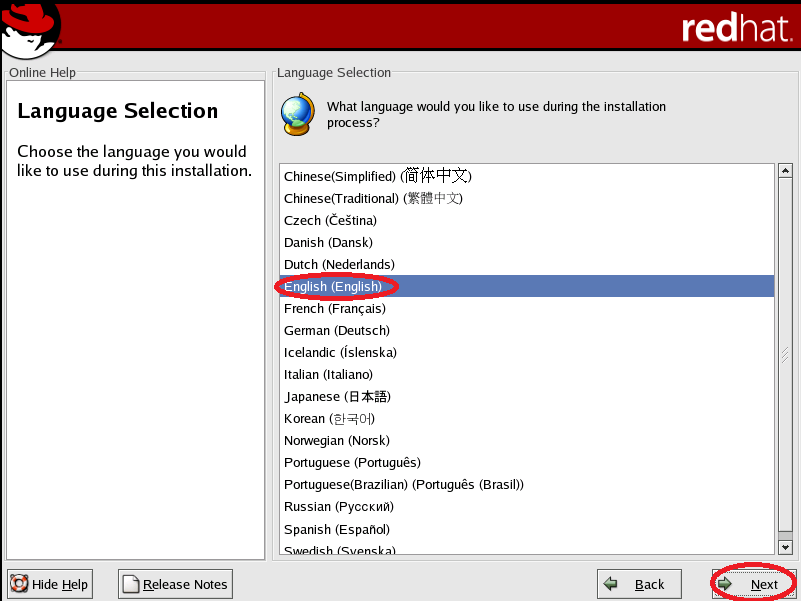
### 等会出现下面的样子， 通过键盘的左右方向键，直接按“skip”按钮，跳过对安装盘的检查



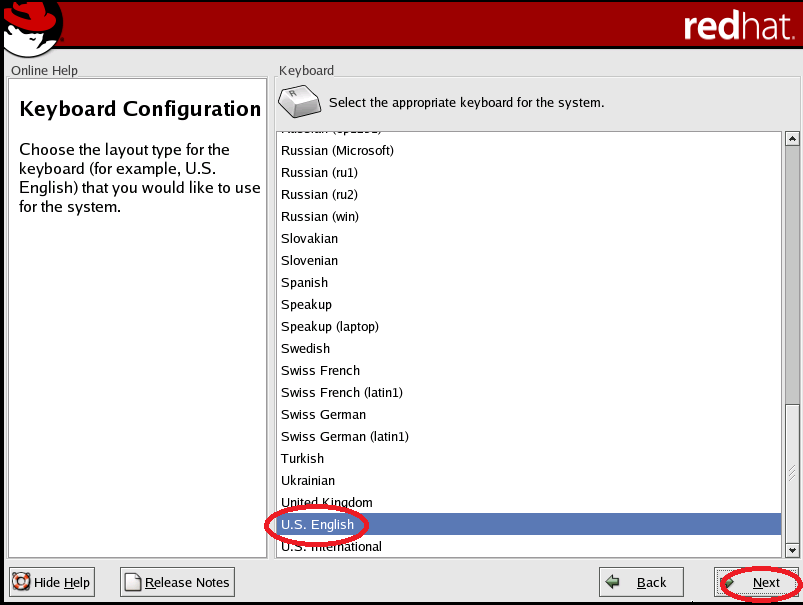
### 等会知道出现欢迎窗口，用鼠标点击“next”



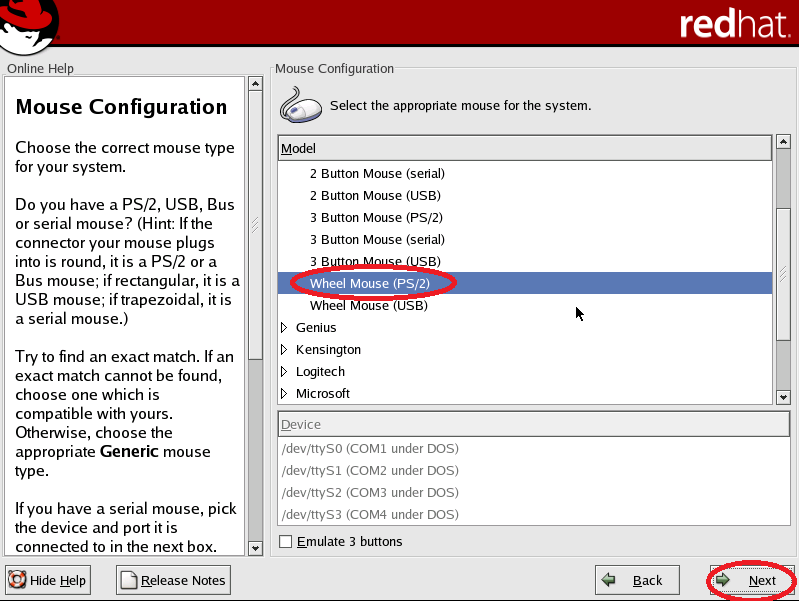
### 设置安装时使用的语言“英语”



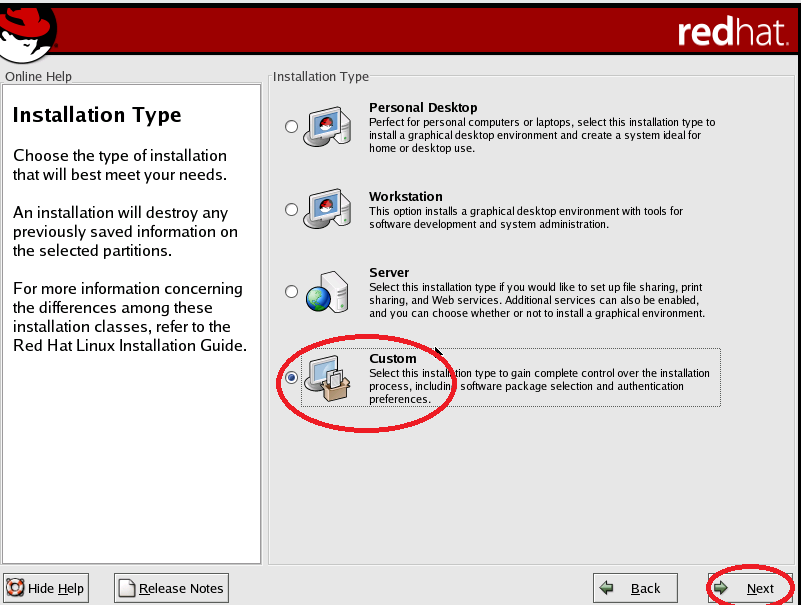
### 设置键盘布局为：英语



### 设置鼠标为：2键无滚轮



### 设置安装类型为：自定义



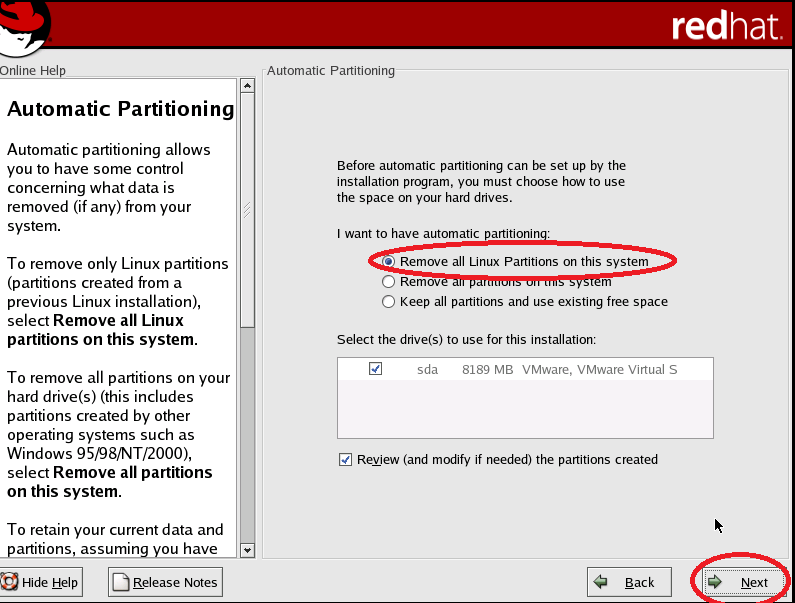
### 设置磁盘划分为自动



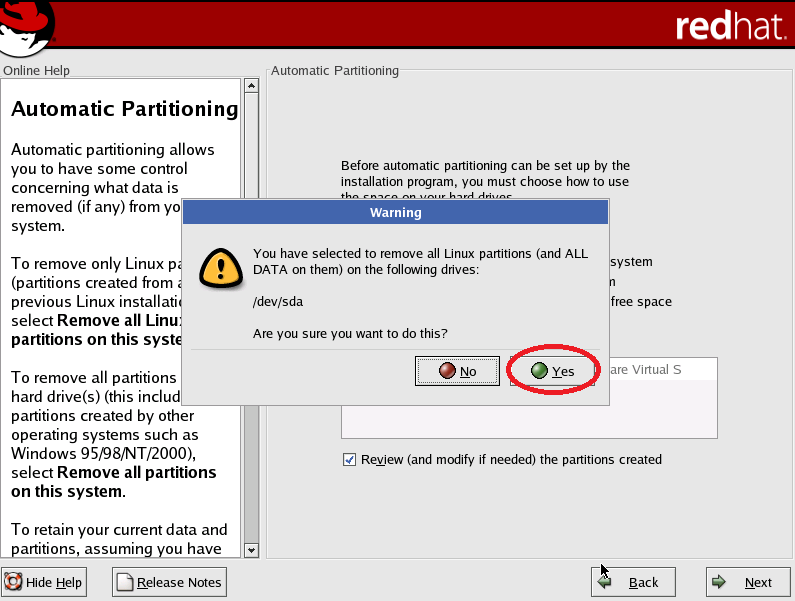
程序会弹框， 提示你安装会重新划分磁盘， 会覆盖磁盘上的数据



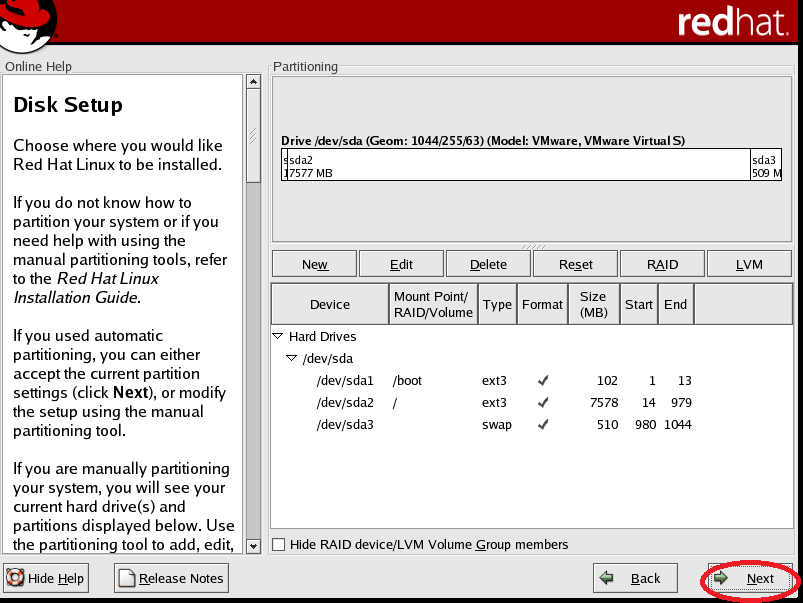
### 设置：删除磁盘上的所以分区， 重新做磁盘划分



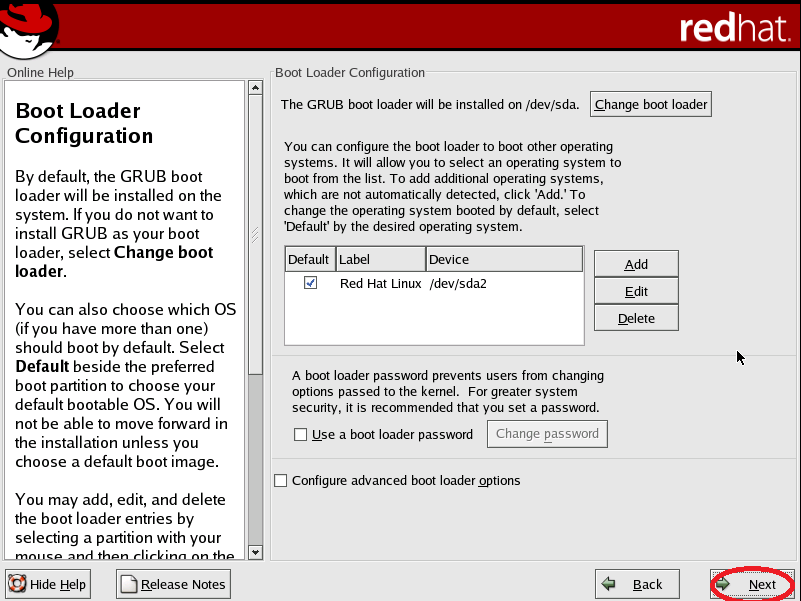
提示下面的磁盘的所以数据将会被删除



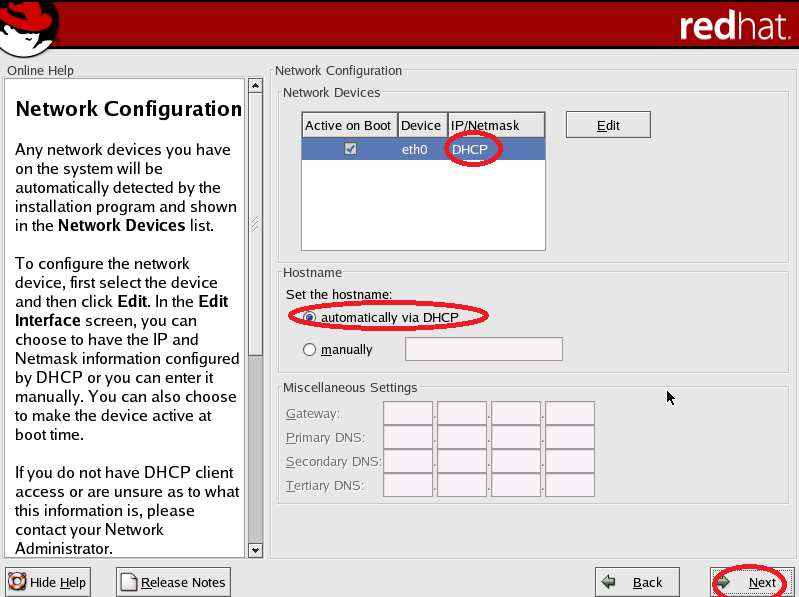
### 同意程序自动划分的磁盘分区



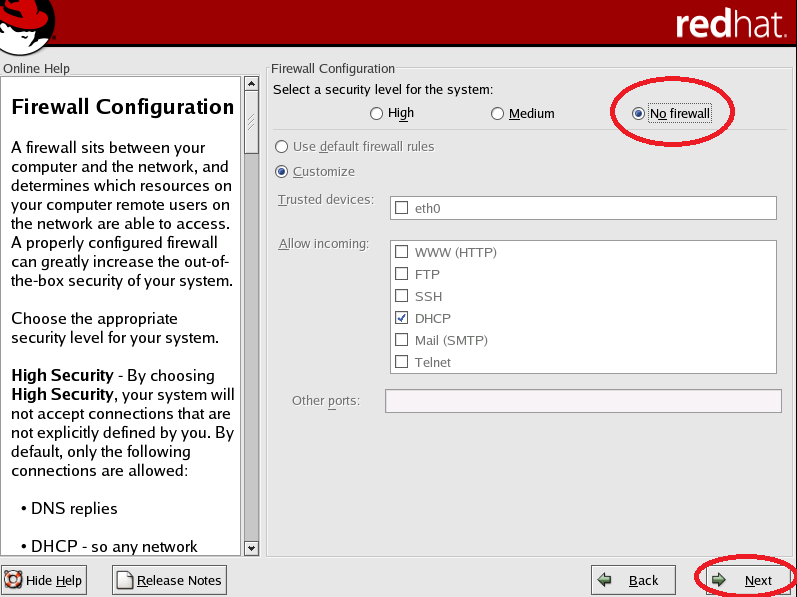
### 设置grub的根启动磁盘



### 设置虚拟机的网络使用DHCP方式

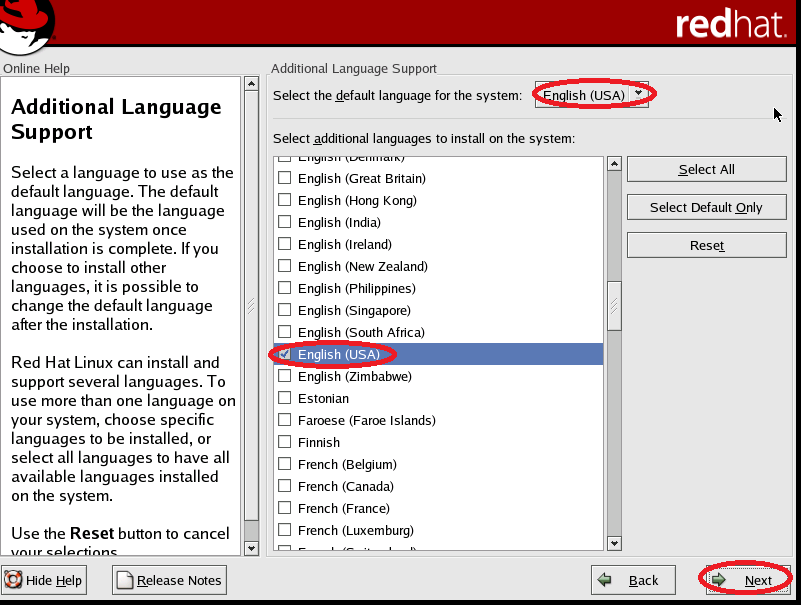


### 关闭虚拟机的防火墙

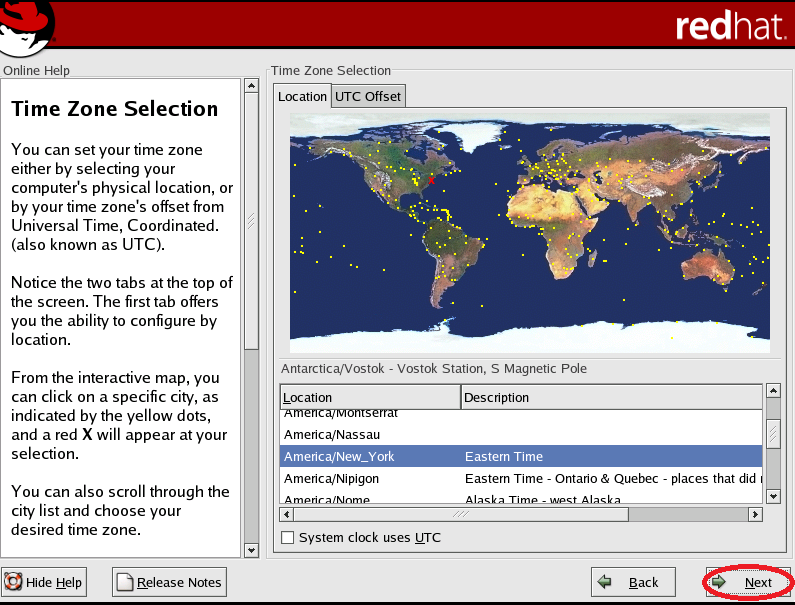


### 设置虚拟机使用的语言：英语

注意前面设置的安装时的语言， 这个是虚拟机安装好后，虚拟机新安装的操作系统的语言



### 时区，随便乱设置

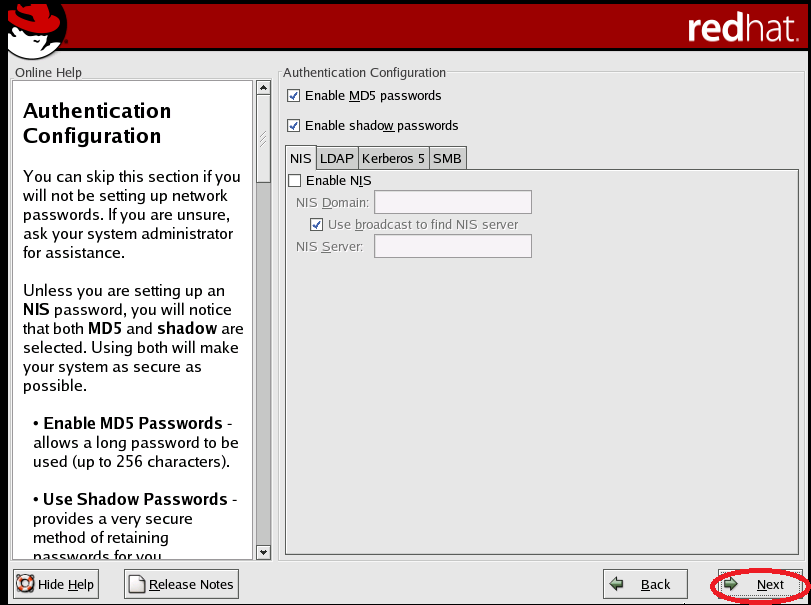


### 输入新系统的root用户的密码

这里输入密码yY123456

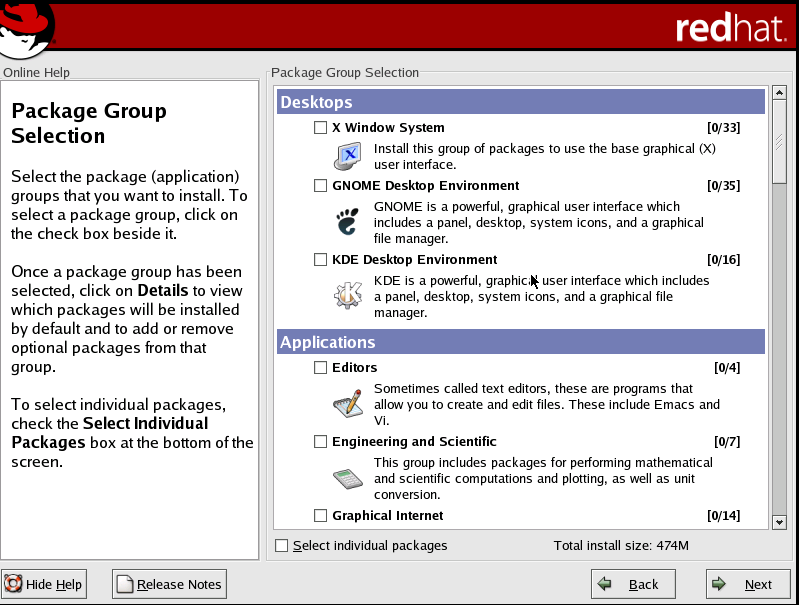


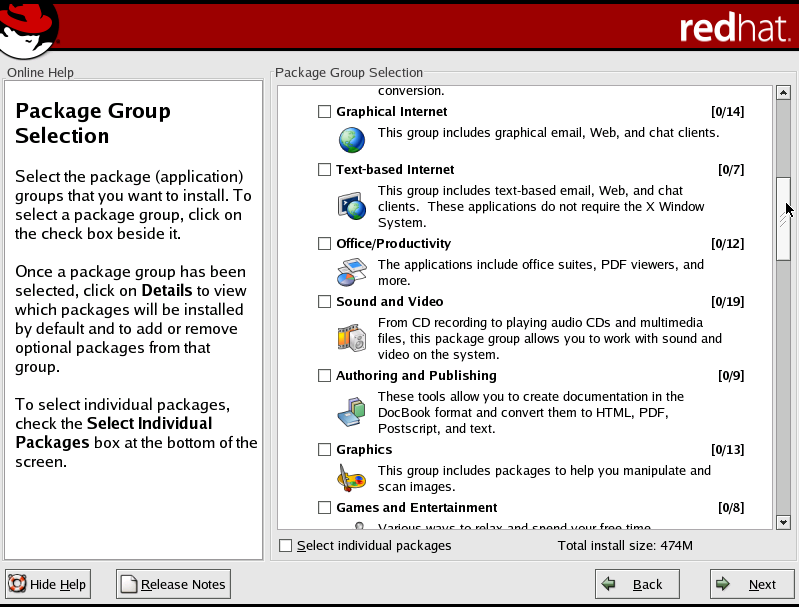
### 认证，直接“next”

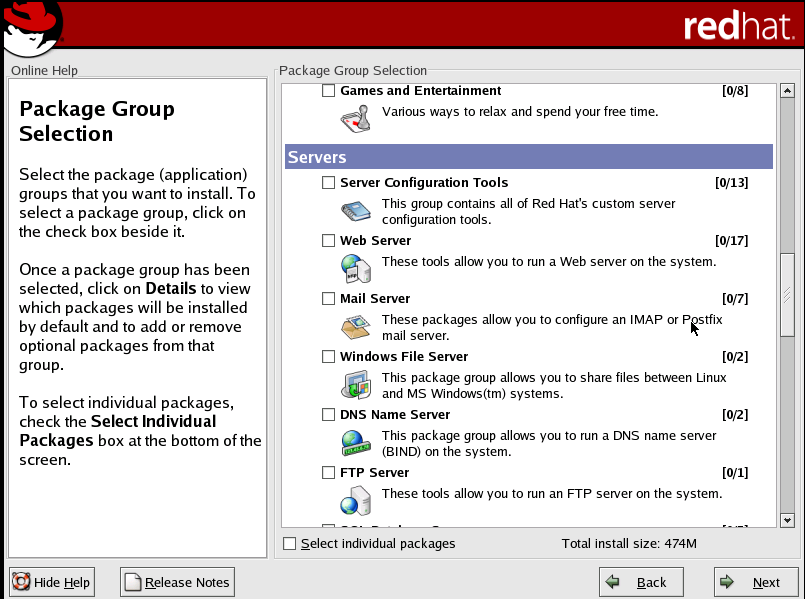


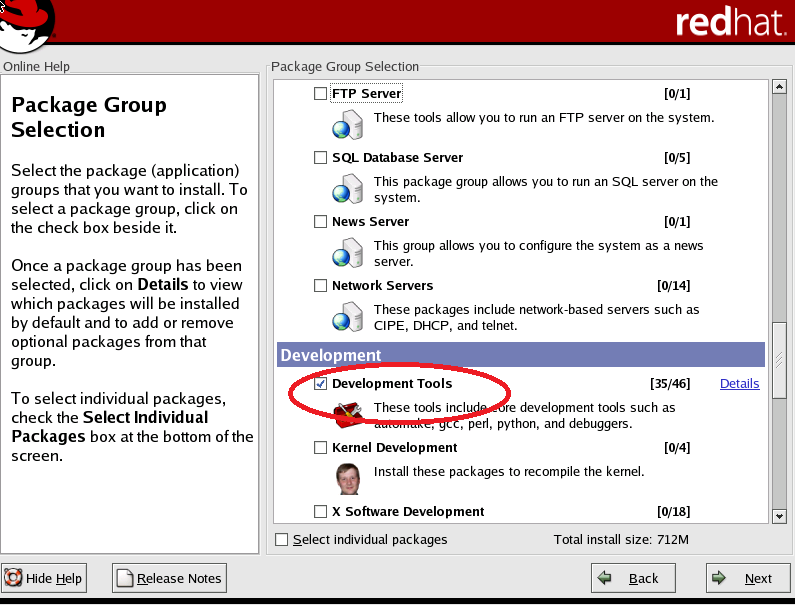
### 所以的包，都不安装，只安装“Devlopment Tools”

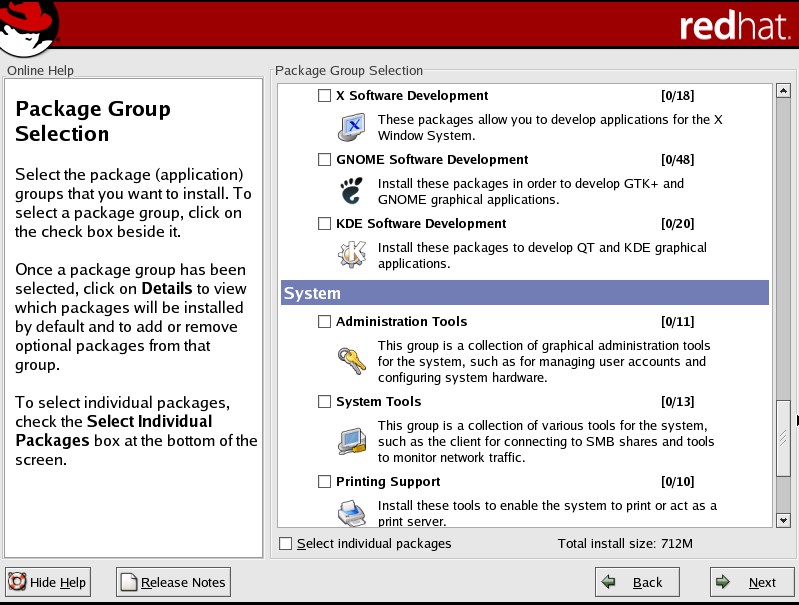
注意看下面的倒数第3副图中的红色圆圈圈起来的类容

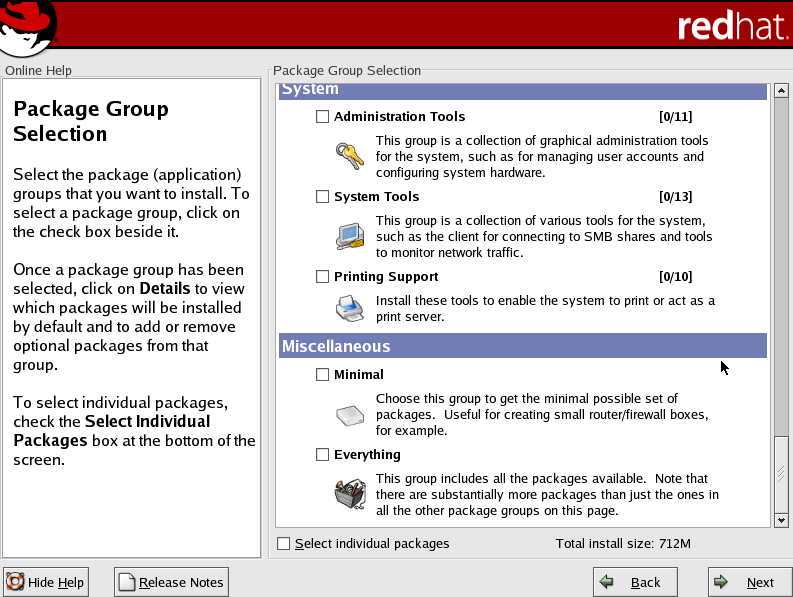




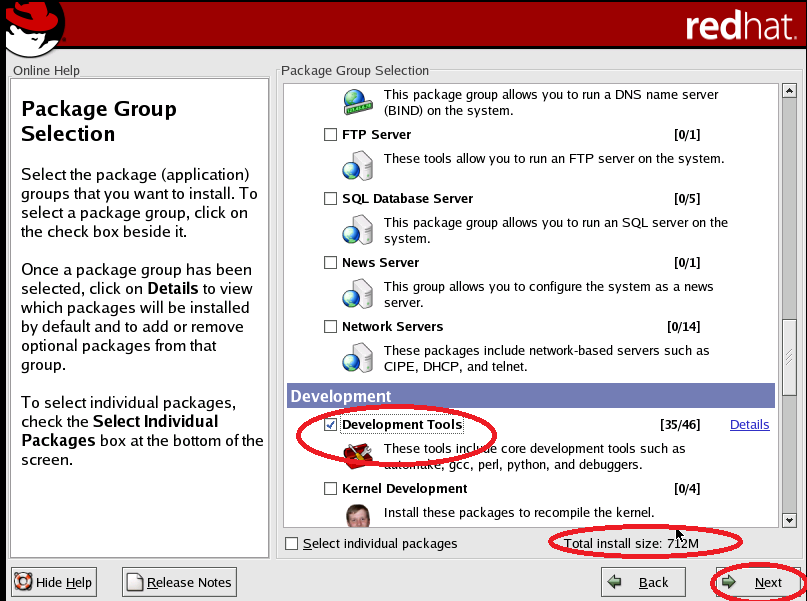




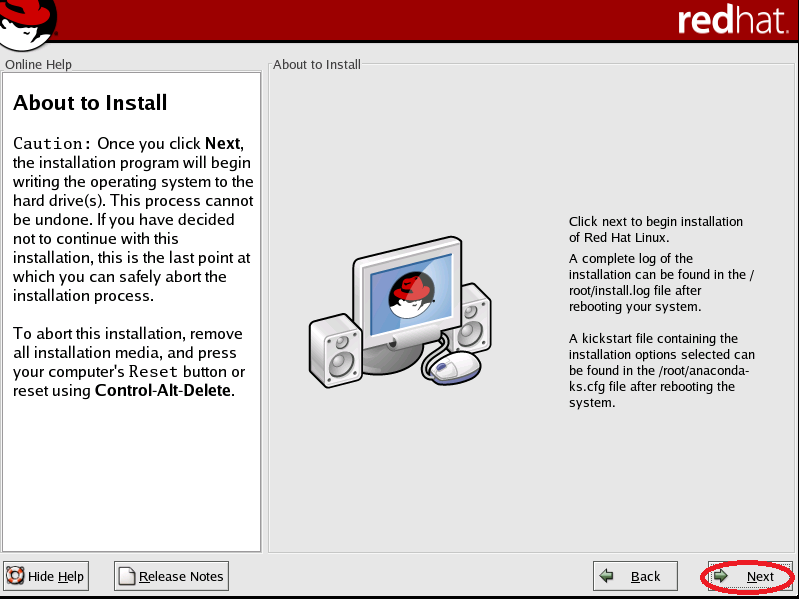




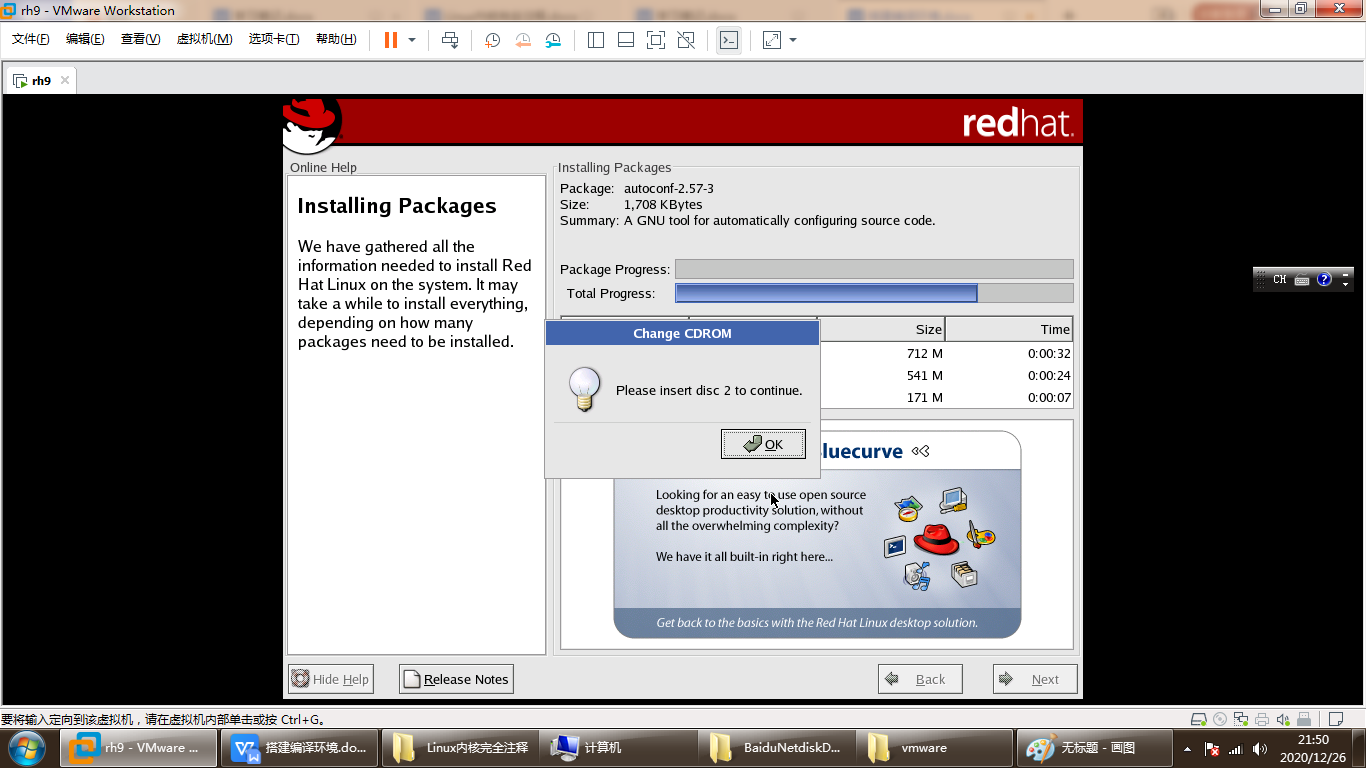
### 选择好安装包后，系统的大小为712M



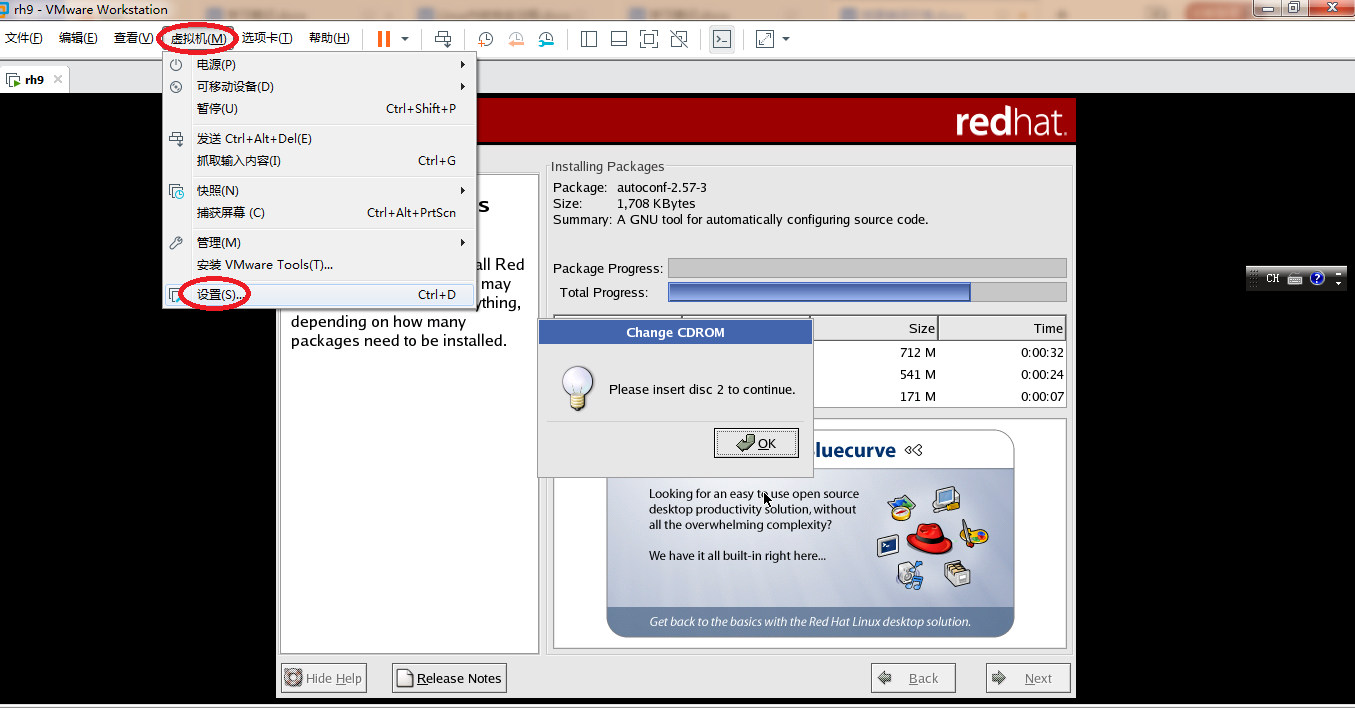
### 直接“next”，开始拷贝文件开始安装



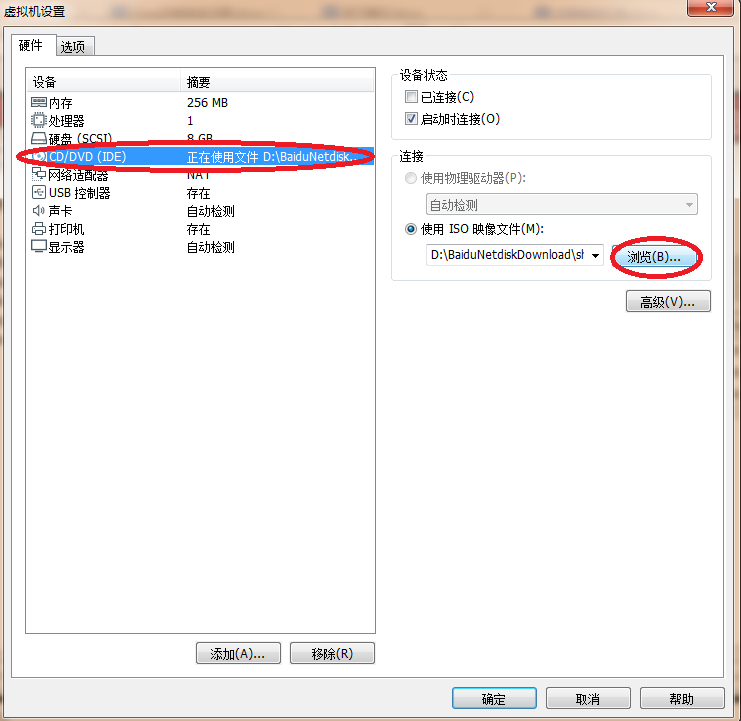
### 当出现下面窗口， 就是“第一张光盘安装完成”，需要插入第二张光盘

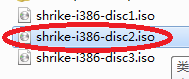


### 点击vmware的菜单

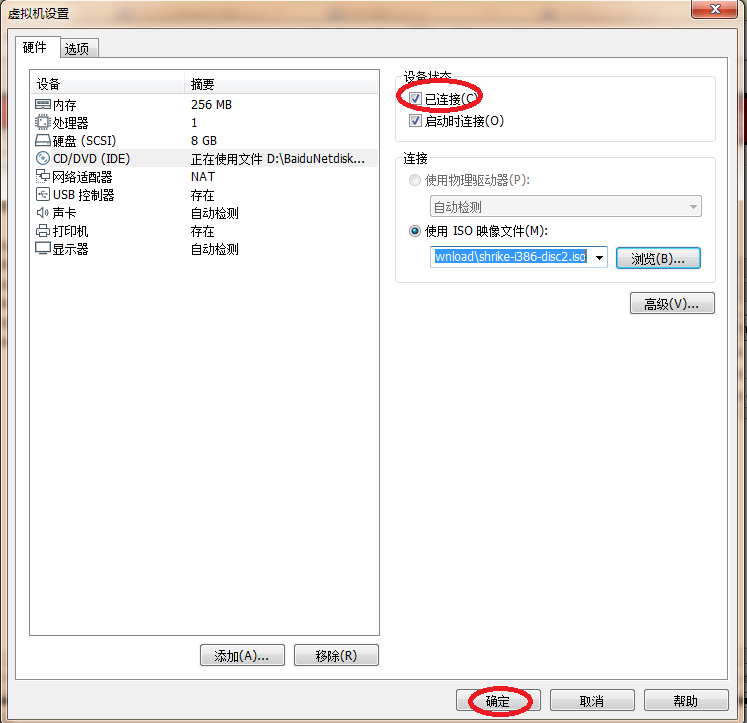


### 选择第2张的安装盘

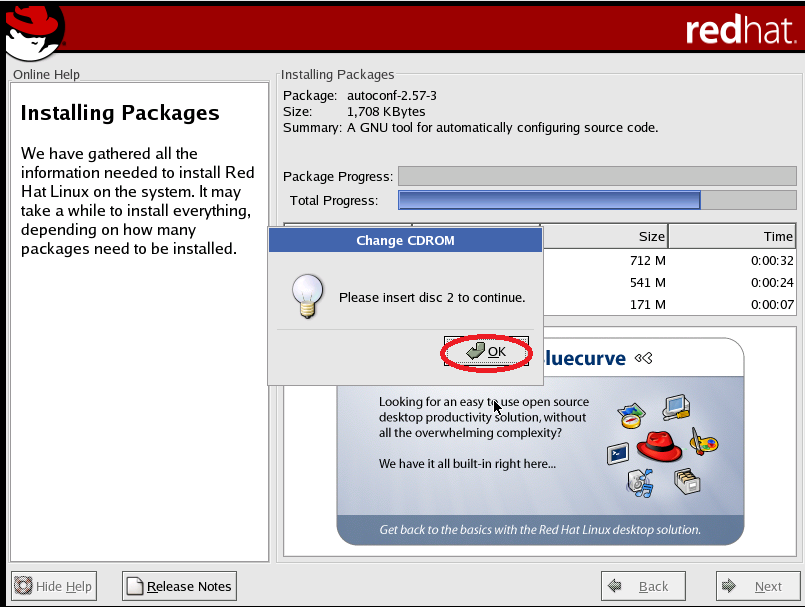




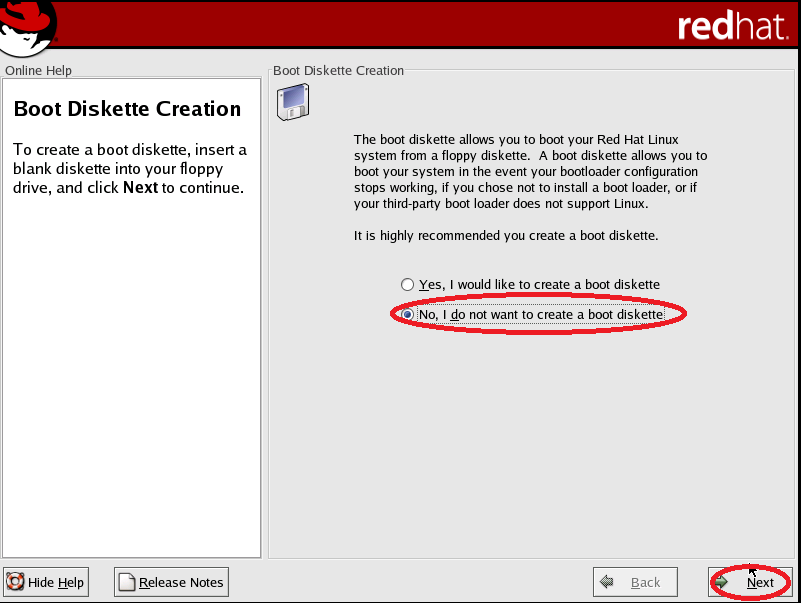
### 选择第2张光盘后，将光盘设置为“已连接”，“点击确定”按钮



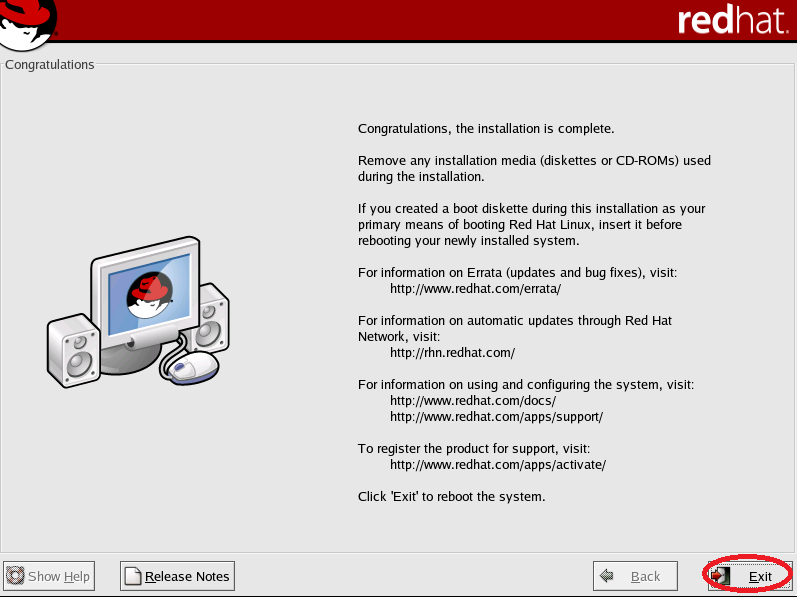
### 进入虚拟机， 点击确定按钮



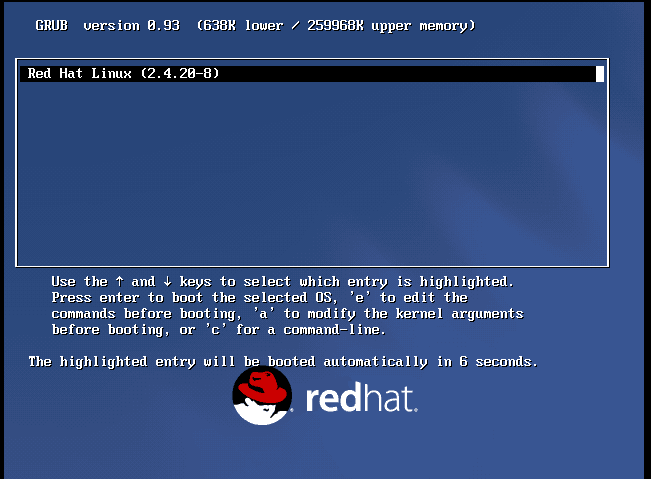
### 不需要制作启动软盘



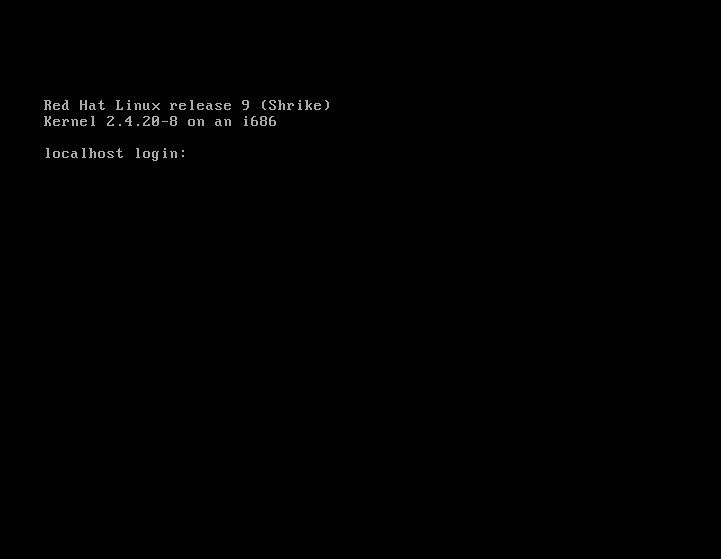
### 安装完成



### 退出后， 虚拟机会自动重启

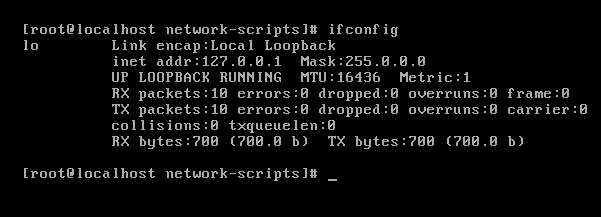


### 重启完成后，linux启动到控制台，输入root，密码（yY123456）登录



### 查看虚拟机的IP地址

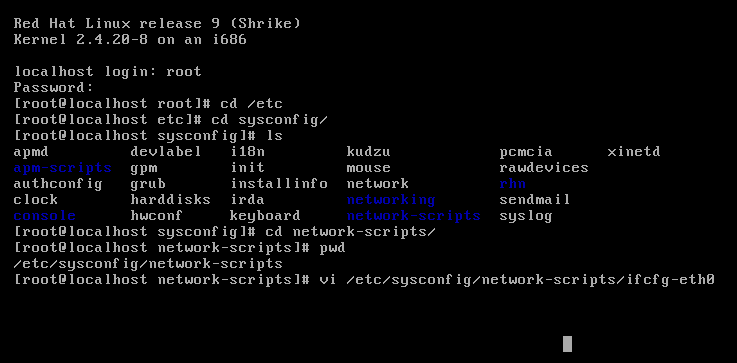
输入命令ifconfig



由上图可以看到，没有有效的ip地址（只有一个127.0.0.1的自环回的地址）

### 编辑

vi /etc/sysconfig/network-scripts/ifcfg-eth0





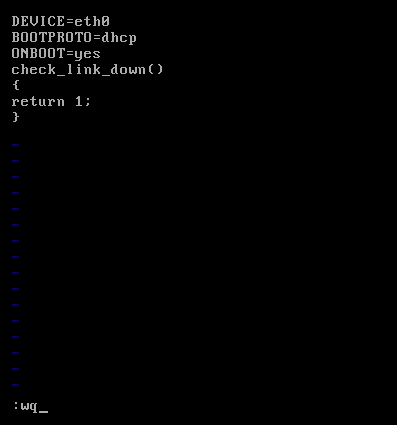
### 在文件末尾添加下面的文字

check\_link\_down()

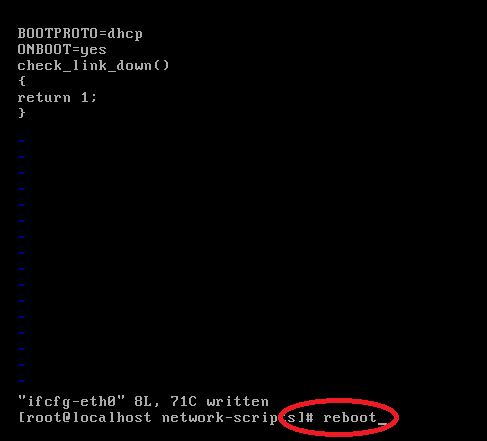
{

return 1;

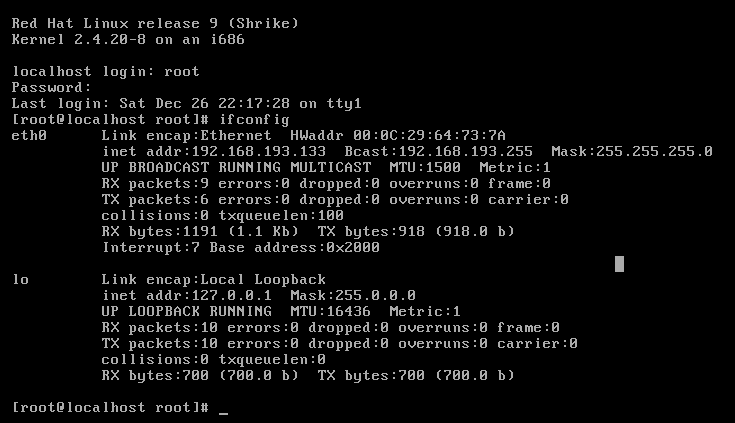
}



### 重启系统



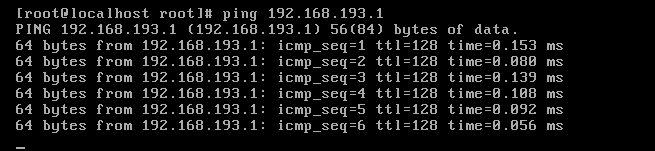
### 启动后重新登录， 使用ifconfig查看虚拟机的ip地址



IP地址已经ok， 是192.168.193.133（由此可以猜测出，虚拟机外的主机的IP地址是：192.168.193.1）

### ping 主机

ping 192.168.193.1

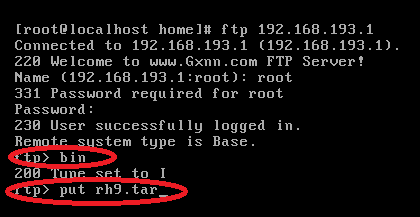


### 完

## 将redhat9.0系统化打包

tar -cvf /home/rh9.tar / --exclude=proc --exclude=dev --exclude=run --exclude=boot --exclude=/home --exclude=/root --exclude=/opt --exclude=/mnt --exclude=misc

## 通过ftp将rh9.tar放到主机上



## 在主机将rh9.tar引入到docker中形成镜像

cat rh9.tar|docker import -rh9

## 上传docker镜像

### 将镜像改名

docker tag 7bf7f696af12 yy20190811/rh9:1.00

### 登录docker服务器

docker login

### 上传镜像

docker push yy20190811/rh9:1.00

# 编译linux0.11

## 下载linux0.11源代码

Linux的源代码都已经很老了，已经很难找到合适的编译环境了。幸好赵炯老师已经准备了一份

<http://www.oldlinux.org/Linux.old/bochs/linux-0.11-gdb-rh9-050619.tar.gz>

注意一定要下载rh9这个版本

解压

tar -zxvf linux-0.11-gdb-rh9-050619.tar.gz



## 运行docker容器

Docker run -itv /home/linux-gdb-rh9/linux:/linux

## 编译

1. Docker进入到linux目录

cd /linux

1. make clean
2. make
3. dd bs=8192 if=Image of=./a.img conv=notrunc