# 分区

## 必须的分区

### boot分区：

作用：引导分区，包含了系统启动的必要内核文件，即使根分区顺坏也能正常引导启动 一般这些文件所占空间在200M以下，

分区格式：建议ext4，按需求更改

### /分区（根分区）:

作用：所有的文件都从这里开始，你可以比喻为Windows的C盘，但其实有区别。如果你有大量的数据在根目录下（比如FTP等）可以划分大一点的空间。

分区格式：建议ext4，按需求更改

### swap分区：

作用：类似于Windows的虚拟内存，在内存不够用时占用硬盘的虚拟内存来进行临时数据的存放，而对于linux就是swap分区

 分区格式：swap格式

## 可选的分区

### home分区：

　作用：存放用户数据，HOME的结构一般是 HOME/userName/userFile，如果不分则默认在/目录下

  分区格式：建议ext4，按需求更改

### var分区

作用：用于log日志的文件的存放，如果不分则默认在/目录

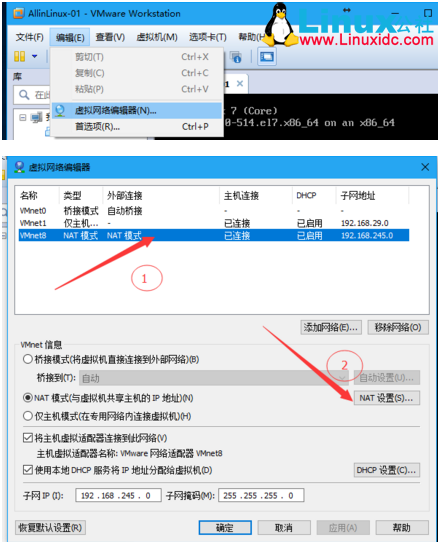
 分区格式：建议ext4，按需求更改

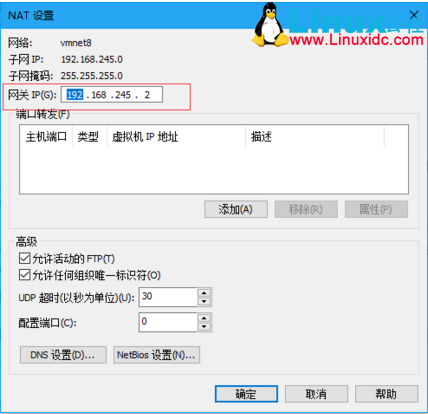
# Ping

NAT和桥接的比较:

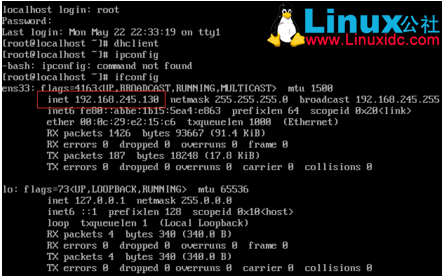
## 1.虚拟机使用NAT模式

NAT模式下的网关地址



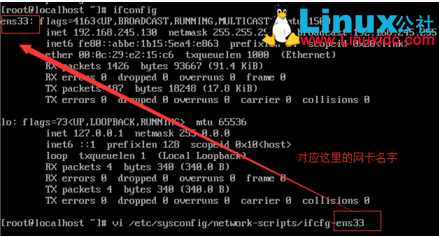


进入虚拟机：ip addr

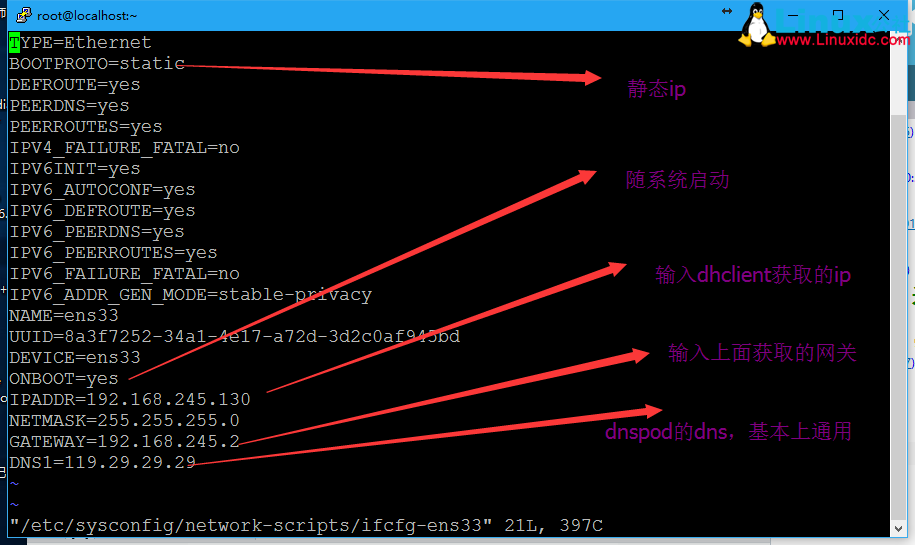


记录下dh服务分配的ip地址

为系统设置一个静态ip，方便日后使用远程连接工具（putty&Xshell）



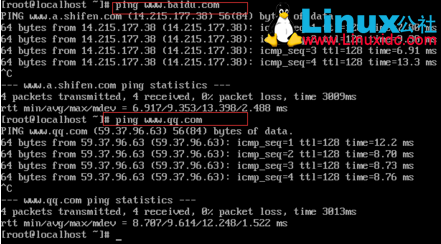
vi /etc/sysconf/network-scripts/ifcfg-en



重启网络服务 CentOS7中使用如下命令 systemctl restart network.service

IMG_262

验证网络配置是否成功 ping 一下 www.baidu.com 或者 www.qq.com



ping通了，就说明NAT模式下的网络配置成功了！

# 远程连接工具（putty&Xshell）

主机号由dh服务分配的ip地址 ip addr 显示的ip

# 4.安装oracle

## 前的准备工作

### **2.0 设置IP地址为静态（固定IP地址）**

### **2.1 修改stsctl.conf文件**

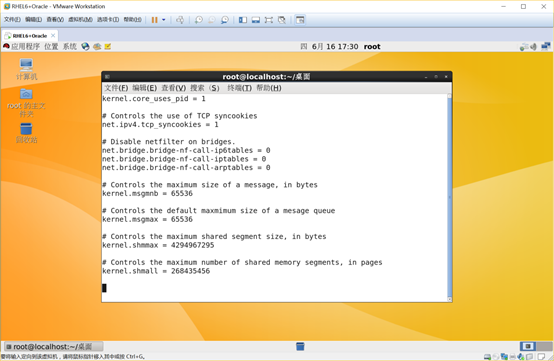
    Linux是为小文件设计的，Oracle数据库安装需要占用较多资源，要把各项参数调大。

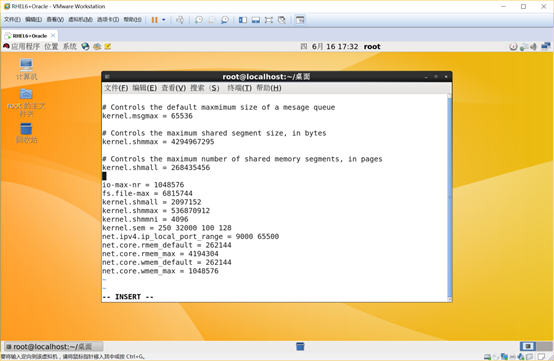
    使用vi编辑/etc/sysctl.conf文件。命令如下：

|  |
| --- |
| vi /etc/sysctl.conf |

    按i进入插入模式，把光标移到文档最后，将以下一段文字（复制自Oracle 11g官方文档）插入到文档最后。

|  |
| --- |
| fs.aio-max-nr = 1048576  fs.file-max = 6815744  kernel.shmall = 2097152  kernel.shmmax = 536870912  kernel.shmmni = 4096  kernel.sem = 250 32000 100 128  net.ipv4.ip\_local\_port\_range = 9000 65500  net.core.rmem\_default = 262144  net.core.rmem\_max = 4194304  net.core.wmem\_default = 262144  net.core.wmem\_max = 1048576  net.ipv4.tcp\_wmem = 262144 262144 262144   net.ipv4.tcp\_rmem = 4194304 4194304 4194304 |





 注意注释掉

 # Disable netfilter on bridges.

 #net.bridge.bridge-nf-call-ip6tables = 0

 #net.bridge.bridge-nf-call-iptables = 0

 #net.bridge.bridge-nf-call-arptables = 0

（重新加载这些参数，一定要重启系统，其他不使用）

    复制完成后，按【ESC】键输入:wq保存退出。

各参数详解：

kernel.shmmax：是核心参数中最重要的参数之一，用于定义单个共享内存段的最大值。设置应该足够大，能在一个共享内存段下容纳下整个的SGA ,设置的过低可能会导致需要创建多个共享内存段，这样可能导致系统性能的下降。至于导致系统下降的主要原因为在实例启动以及ServerProcess创建的时候，多个小的共享内存段可能会导致当时轻微的系统性能的降低(在启动的时候需要去创建多个虚拟地址段，在进程创建的时候要让进程对多个段进行“识别”，会有一些影响)，但是其他时候都不会有影响。

官方建议值：

32位linux系统：可取最大值为4GB（4294967296bytes）-1byte，即4294967295。建议值为多于内存的一半，所以如果是32为系统，一般可取值为4294967295。32位系统对SGA大小有限制，所以SGA肯定可以包含在单个共享内存段中。

64位linux系统：可取的最大值为物理内存值-1byte，建议值为多于物理内存的一半，一般取值大于SGA\_MAX\_SIZE即可，可以取物理内存-1byte。例如，如果为12GB物理内存，可取12\*1024\*1024\*1024-1=12884901887，SGA肯定会包含在单个共享内存段中。

kernel.shmall： 该参数控制可以使用的共享内存的总页数。Linux共享内存页大小为4KB,共享内存段的大小都是共享内存页大小的整数倍。一个共享内存段的最大大小是16G，那么需要共享内存页数是16GB/4KB=16777216KB /4KB=4194304（页），也就是64Bit系统下16GB物理内存，设置kernel.shmall = 4194304才符合要求(几乎是原来设置2097152的两倍)。这时可以将shmmax参数调整到16G了，同时可以修改SGA\_MAX\_SIZE和SGA\_TARGET为12G（您想设置的SGA最大大小，当然也可以是2G~14G等，还要协调PGA参数及OS等其他内存使用，不能设置太满，比如16G）

kernel.shmmni：该参数是共享内存段的最大数量。shmmni缺省值4096，一般肯定是够用了。

fs.file-max：该参数决定了系统中所允许的文件句柄最大数目，文件句柄设置代表linux系统中可以打开的文件的数量。

fs.aio-max-nr：此参数限制并发未完成的请求，应该设置避免I/O子系统故障。推荐值是：1048576 其实它等于 1024\*1024 也就是 1024K 个。

kernel.sem：以kernel.sem = 250 32000 100 128为例：

       250是参数semmsl的值，表示一个信号量集合中能够包含的信号量最大数目。

       32000是参数semmns的值，表示系统内可允许的信号量最大数目。

       100是参数semopm的值，表示单个semopm()调用在一个信号量集合上可以执行的操作数量。

       128是参数semmni的值，表示系统信号量集合总数。

net.ipv4.ip\_local\_port\_range：表示应用程序可使用的IPv4端口范围。

net.core.rmem\_default：表示套接字接收缓冲区大小的缺省值。

net.core.rmem\_max：表示套接字接收缓冲区大小的最大值。

net.core.wmem\_default：表示套接字发送缓冲区大小的缺省值。

net.core.wmem\_max：表示套接字发送缓冲区大小的最大值。

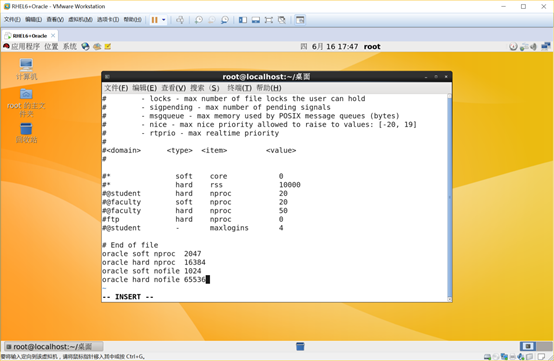
### **2.2 修改limits.conf**

    修改：/etc/security/limits.conf

|  |
| --- |
| vi /etc/security/limits.conf |

    在文件末尾加入以下内容：

|  |
| --- |
| oracle soft nproc  2047  oracle hard nproc  16384  oracle soft nofile 1024  oracle hard nofile 65536   oracle hard stack 10240 |

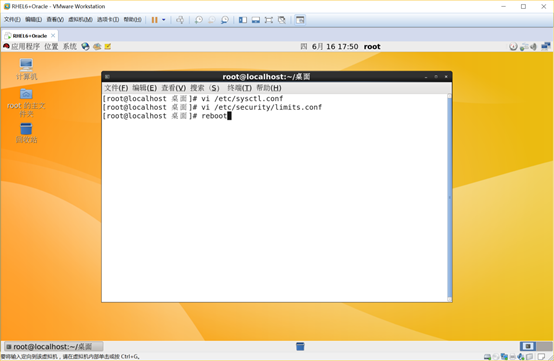


保存并退出。

### **2.3 重启系统**

    修改完参数之后就要重启了，这样做是为了系统能认识我们之前配置的参数。输入以下命令：

reboot



## 2.4 执行一系列操作

    以普通用户登录**（千万不要用root！！！！）**，然后切换到root用户。（密码就是root的密码）

|  |
| --- |
| su root |

执行以下命令：

#### **2.4.1 创建组dba**

|  |
| --- |
| groupadd oracle |

tail /etc/group

**2.4.2 创建oracle用户并添加到dba组**

    继续输入以下命令：

|  |
| --- |
| useradd oracle |

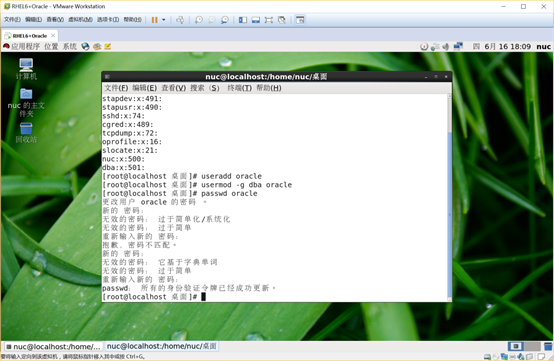
    将oracle用户添加到dba组

|  |
| --- |
| usermod -g oracle oracle |



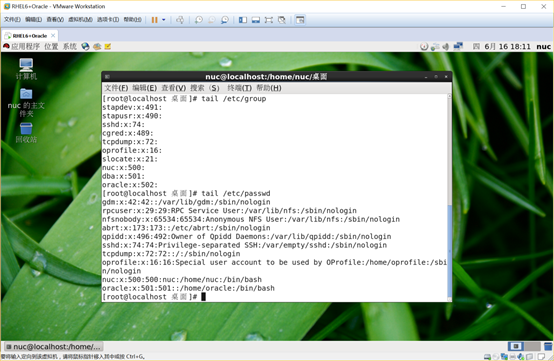
给这个用户加一个密码（密码一定要好记，别管他的提示），命令如下：

|  |
| --- |
| passwd oracle |



测试一下刚才的操作是否成功：

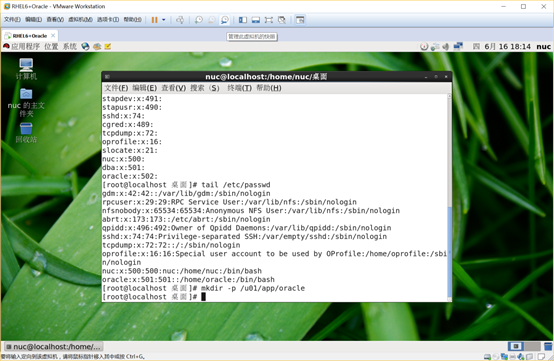
|  |
| --- |
| tail /etc/group  tail /etc/passwd |



#### **2.4.3 为Oracle安装创建目录**

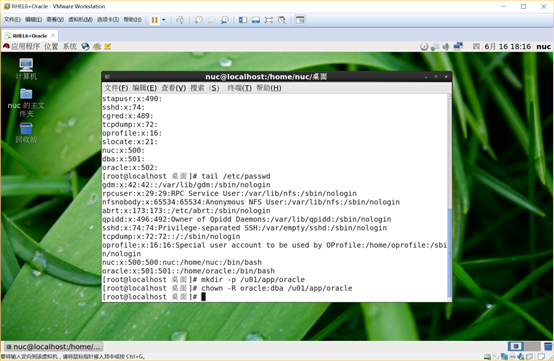
    执行以下命令创建目录：

mkdir -p /opt/app/oracle



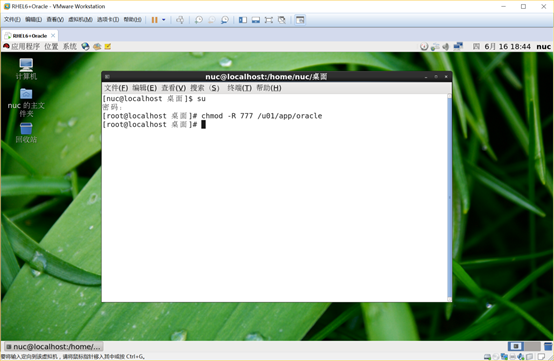
执行以下命令，将文件的属主改为Oracle，属组改为dba：

|  |
| --- |
| chown -R oracle:oracle /opt/app/oracle |



修改文件的访问权限为777（4+2+1,4+2+1,4+2+1）：

|  |
| --- |
| chmod -R 777 /opt/app/oracle |

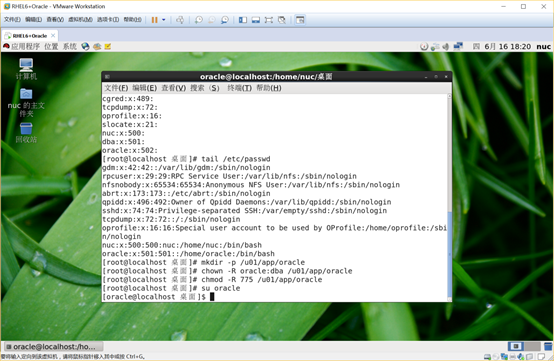


查看一下你刚建的文档权限，这个十分重要，用命令： ll    查看那个几个文件的权限，                                               
  
[oracle@server app]$ ll  
total 8  
**drwxrwxrwx**. 10 **oracle dba** 4096 Jun 19 20:22 **oracle**  
**drwxrwx**---.  5 **oracle dba** 4096 Jun 19 18:59 **oraInventory**

上面两个文件的权限影响后面执行的权限问题，一定记得检查！

#### **2.4.4 切换到Oracle用户**

|  |
| --- |
| su oracle |



#### **2.4.5 配置概要文件(oracle下)**

    输入以下命令：

|  |
| --- |
| cd /home/oracle  vi .bashrc |

    进入vi后按键盘上的字母i切换到插入模式，然后插入以下内容：

|  |
| --- |
| umask 002  ORACLE\_BASE=/opt/app/oracle      //这个决定后面按安装的目录  export ORACLE\_BASE    ORACLE\_HOME=$ORACLE\_BASE/product/11.2.0    //这个也是  export ORACLE\_HOME    ORACLE\_SID=dog         //和你后要建的数据库事例名字一样,我的是：dog，以防出现SID不一致的问题  export ORACLE\_SID    PATH=/usr/bin:/bin:usr/local/bin:/usr/bin/X11:/usr/bin/X11R6  PATH=$PATH:$ORACLE\_HOME/bin  export PATH |



4、添加/etc/hosts中dns解析信息（这个是root下执行，这个里面原来是有东西的，不是一个新的文件，将一些的文件追加到文件的最后）

192.168.1.102            host.data（这个是你安装linux时的主机名)

192.168.1.102             localhost

 查看主机名： hostname

5.root 用户下（文件里面是有东西的，在最后追加）

  vi /etc/pam.d/login

   session required /lib/security/pam\_limits.so

   session required pam\_limits.so

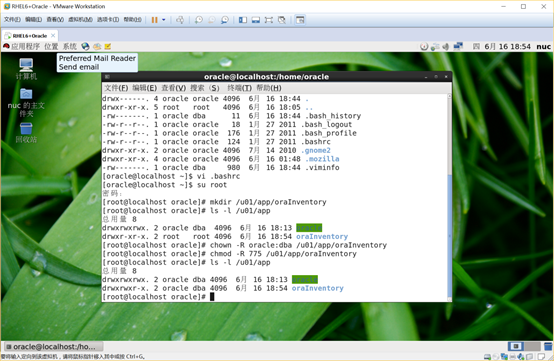
（重新加载这些参数，一定要重启系统，其他不使用）至此，准备工作全部完成！

## **3.安装Oracle**

### **3.1 创建安装目录并将其属组改为Oracle**

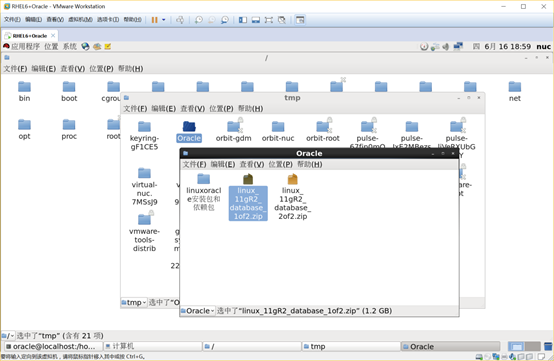
    获取root权限并执行以下命令：

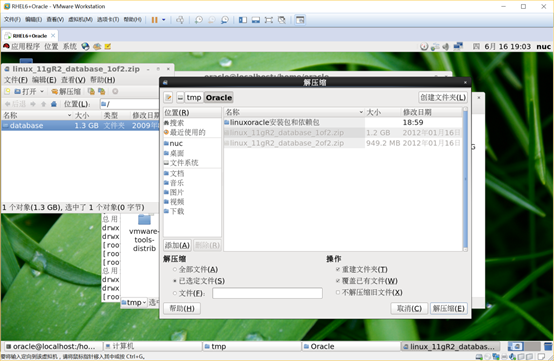
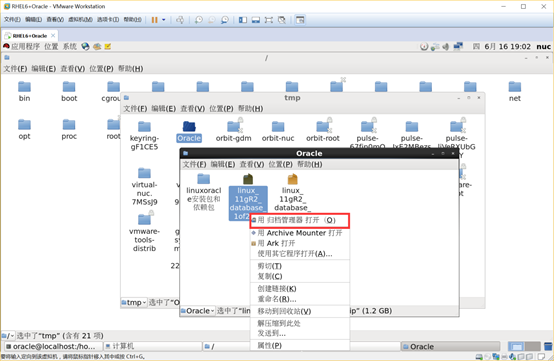
|  |
| --- |
| su root  mkdir /opt/app/oraInventory  ls -l /opt/app  chown -R oracle:oracle /opt/app/oraInventory  chmod -R 775 /opt/app/oraInventory  ls -l /u01/app |

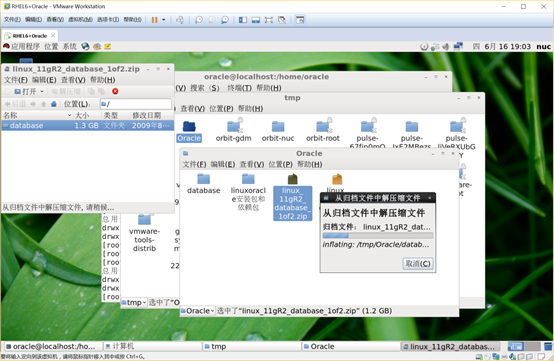


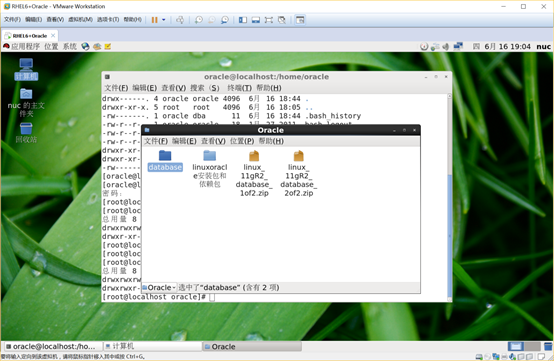
主要一定要看上面的文件是否权限一致，这特便是是oracle  和 oraInventory 的权限以及读写权限，简单的来说是他们前面的东西一致

### **3.2 解压Oracle安装文件**

    先将oracle安装程序拷贝到一个目录下（我解压到tmp的oracle目录下，如果没有，自己建一个目录，在外面复制安装程序，里面粘贴。如果文件无法导入到虚拟机，请检查VMtools是否正确安装）： 





另一个文件同理。最终得到如下结果：

这里，其实还有很多问题，若你用的是WinScp上传的话，可能是root权限。建议用oracle用户上传，若是root 的用户，那 你的

先root 下进行建文件夹，而后 将其改成oracle的 宿主 ，即：

[oracle@server app]$ mkdir  temo

[oracle@server app]$ chown -R oracle:oracle  temo

[oracle@server app]$ chmod 777  temo

然后你在oracle用户下按上面的操作，将其解压到temo 中，

而后最好我的建议是你查看一下刚刚解压后的文件的权限，若database中的文件都是oracle 用户就最好了。

### **3.3 运行安装文件(oracle用户下安装，因为这个是oracle用户用的)**

如你当前系统环境是英文的环境一些不用看

若你是当前系统环境是中文的环境必须看，否则会出现框框乱码问题，

（

\*\*——设置编码方式：在oracle用户下：

export LANG=en\_US.UTF-8

这个你可以参考博客：http://www.cnblogs.com/nucdy/p/5598497.html

有关一个“red hat 系统将中文设置为英文环境“ 文章

若你已经运行了，没有关系，点击oracle的那个close ，

然后在命令行中以oracle用户下输入

export LANG=en\_US.UTF-8

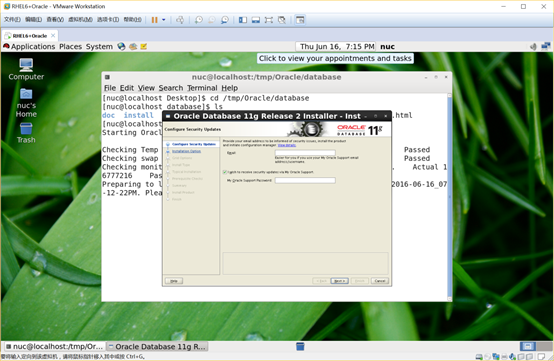
而后执行下面继续运行没有问题

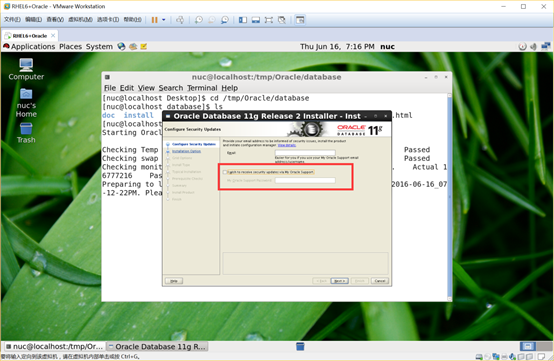
\*\*\*）

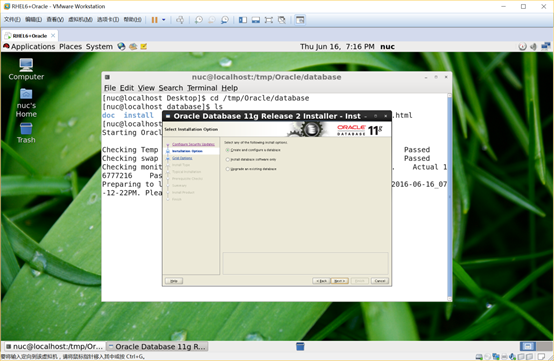
这个是在oracle用户  下运行的

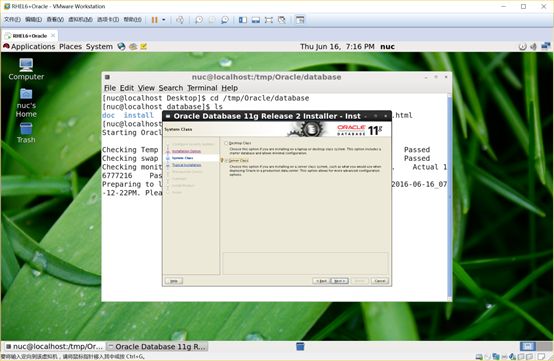
[oracle@server app]$

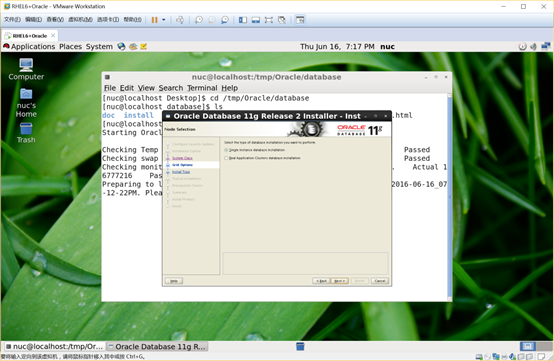
|  |
| --- |
| cd /tmp/Oracle/database          //这个是进入到你oracle解压后的目录下，根据你自己  ./runInstaller |









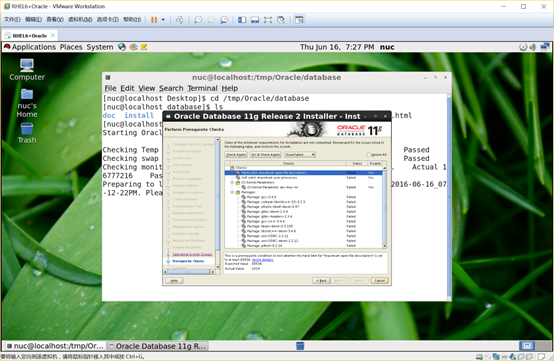


下一步是典型安装：毕竟现在在是用来进行学习的，先安装上即可，不用先急的去安装一个自己手工设置的数据事例，以后在进行尝试也不迟，况且这样做完了，你还得进行配置更多的其他东西，你可以网上搜索一下，安装完还是要需要进行一些数据库事例的相关配置很麻烦的，这个先不讲，以下我主要介绍是典型安装的过程

选择typical install

然后，选择下一步，下一步

接下来是会出现，一下界面，你按照这个继续进行中就行

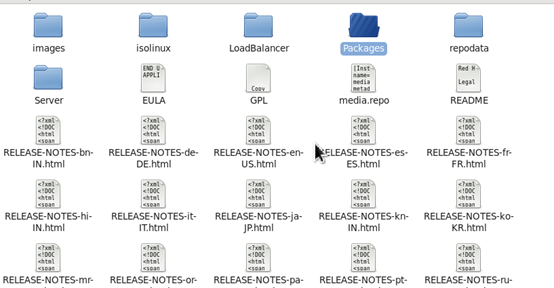


由于Oracle安装时缺少依赖包，我们需要添加这些依赖包，命令另起一个终端，输入：

|  |
| --- |
|  |

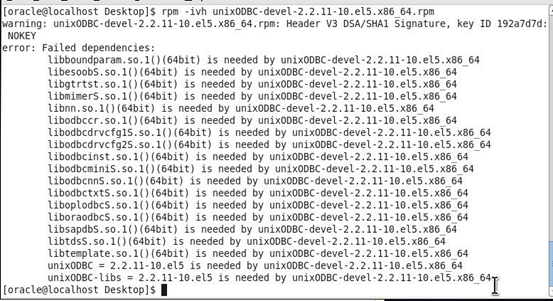
安装依赖包比较麻烦，可以直接把老师发的iso文件放入linux，(/media/RHEL\_6.2 i386 Disc 1/Packages) ，这个目录找上面对应的所需要的.rmp 文件，进行安装

rpm -ivh xxx.rpm安装即可

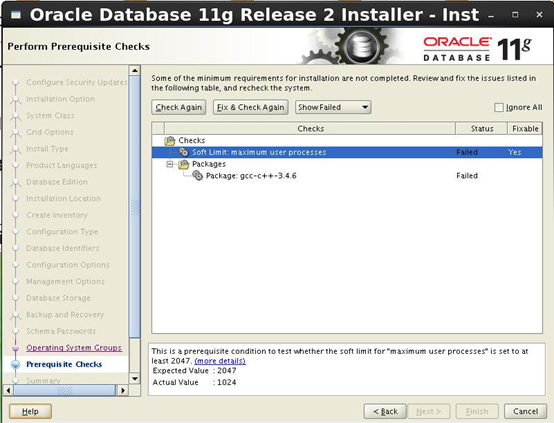


在packages找到相应的rpm文件,在终端执行命令 rpm -ivh xxx.rpm安装即可

**可能出现的错误（包很多或者修改文件权限不够）：例如**

****

IMG_288

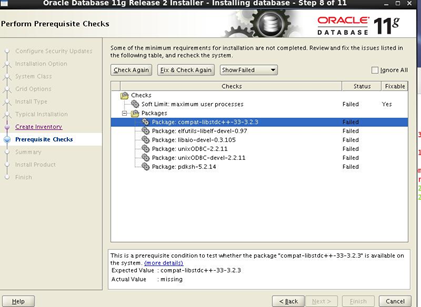


其实，我想说的是出现这个问题，是应为你在安装linux时候就应该选择定制软件包的时候，选择上那个“开发” ，将那几个能选择的都选择上肯定不会出现这种情况。

当然，你也可以自己找找，里面有关这个gcc c 啥啥的东西，后面忘了，还有gcc c++啥的包，就不会出现上面的问题

解决这个问题可以参考http://blog.csdn.net/zb0567/article/details/41048263或者百度。

(其中，在这里若出现了那个soft Limit:maximux usr pricess 情况，即：maximum user processes)，是你之前配置的 /etc/security/limits.conf中出现问题了，请去检查相关的是否写错。



同时。老师那个/etc/sysctl.conf中信息的一个信息改为：kernel.shmall = 2097152

/etc/sysctl.conf中信息

fs.aio-max-nr = 1048576

 fs.file-max = 6815744

 kernel.shmall = 2097152

 kernel.shmmax = 536870912

 kernel.shmmni = 4096

 kernel.sem = 250 32000 100 128

 net.ipv4.ip\_local\_port\_range = 9000 65500

 net.core.rmem\_default = 262144

 net.core.rmem\_max = 4194304

 net.core.wmem\_default = 262144

 net.core.wmem\_max = 1048586

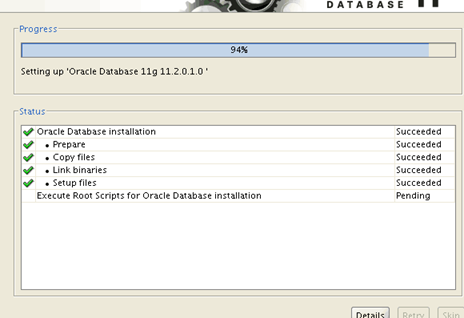
 net.ipv4.tcp\_wmem = 262144 262144 262144

 net.ipv4.tcp\_rmem = 4194304 4194304 4194304

找到相应依赖包，如果不能成功的话,su root ——>输入密码  然后找到对应的rpm文件再重新在终端执行命令 rpm -ivh xxx.rpm。

大家可以参考文档<http://www.jb51.net/article/53769.htm>

而后点击安装完上面的rpm后，点击Check again 即可，没问题，就下一步



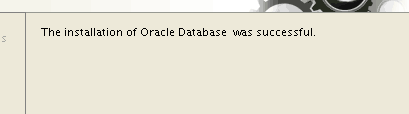
这块，没有截图，有一个从新弹出一个框，

而后，你可以将点击那个修改，进行将一些用户进行解锁，

比如较为长用的一个用户Scott，将对勾进行点击去掉，而后输入密码，确认密码，

点击确定，对话框点击yes

后面就行下一步，继续即可



看到这个页面可以初步判断oracle初步安装成功，接下来按照步骤创建数据库即可。

我安装参考的文档：http://www.jb51.net/article/53769.htm

好，上面如果你能成功的安装了，就暂时说明好了，但是你还得去oracle中查看一下，是否启动。

(我一般是将的关闭防火墙：chkconfig iptables off)读者可以不去选择去执行

手工启动：(这里是启动一些服务，就行windows中的服务一些，需要手工启动，读者进操作一些的命令即可)

[oracle@crmdb ~]$ lsnrctl start           //success 表明成功

[oracle@crmdb ~]$ dbstart   //启动监听器,我自己试了一下，没启动也没有事情

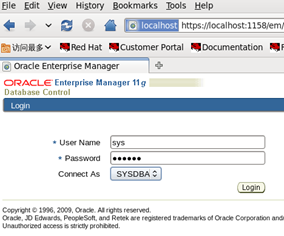
[oracle@crmdb ~]$ emctl start dbconsole

//自后出现啥啥啥success，代表成功 ，这个是可以启动浏览器的进行的操作

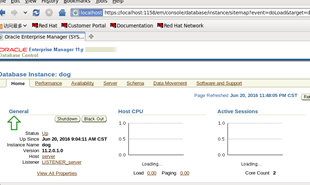
/\*\*浏览器中输入https://localhost:1158/em/

//usrname :sys

//password:自己设置的密码



登录即可，出现



就行了

\*\*\*\*/

//下面我没有操作,而是直接进行后面步骤:“二、去试试你安装的数据库，查看一下是否能用” ，如果能用，说明你的已经装成功了，下面就不用去管了

应对shared memory realm does not exist 错误的处理

1、实例没有启动

sqlplus /nolog

connect / as sysdba

startup

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

监听器启动出现问题：

ORACLE\_HOME\_LISTNER is not SET, unable to auto-start Oracle Net Listener

使用oracle用户执行dbstart启动数据库提示如下：

ORACLE\_HOME\_LISTNER is not SET, unable to auto-start Oracle Net Listener

Usage: /oracle/11g/bin/dbstart ORACLE\_HOME

Processing Database instance "orcl": log file /oracle/11g/startup.log

修改dbstart以及dbshut两个文件

--------------------------

第一步:

把ORACLE\_HOME\_LISTNER=/ade/vikrkuma\_new/oracle注释掉

然后修改/home/oracle/.bash\_profile

添加export ORACLE\_HOME\_LISTNER=$ORACLE\_HOME一句

生效变量：

[root@oracle01 ~]# source /home/oracle/.bash\_profile

图为：

---------------------------

第二步:

#vi /etc/oratab

找到orcl=/u01/app/oracle:N这一行

改为：

orcl=/u01/app/oracle:Y

也就是将最后的N改为Y，意思是将不允许自动启动改为允许自动启动。即可解决问题$

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

自动启动的脚本：

\*\*\*\*\*\*\*\*\*\*1\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

首先修改/etc/oratab文件

#vi /etc/oratab

找到orcl=/u01/app/oracle:N这一行

改为：

orcl=/u01/app/oracle:Y

也就是将最后的N改为Y，意思是将不允许自动启动改为允许自动启动。

\*\*\*\*\*\*\*\*\*\*2\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

启动了Linux系统之后，转到  /etc/init.d  目录下；

[root@oracle ~]# cd /etc/init.d

使用 vi 命令，新建一个以 oracle 命名的文件（并将以下代码复制至文件中）

[root@oracle init.d]# vi oracle

以下是代码：

曾经出现过一个问题：

“服务不支持 chkconfig”：——这个提示是因为oracle自启动脚本前面必须要有以下这两行

#chkconfig: 2345 00 01

#description:oracle 11g service

--------------------------------------------------------------------------------------------------------------

#!/bin/sh

# chkconfig: 345 61 61

# description: Oracle 11g AutoRun Services

# /etc/init.d/oracle

#

# Run-level Startup script for the Oracle Instance, Listener, and

# Web Interface

export ORACLE\_BASE=/opt/oracle

export ORACLE\_HOME=$ORACLE\_BASE/oracle

export ORACLE\_SID=zqaccp

export PATH=$PATH:$ORACLE\_HOME/bin

ORA\_OWNR="oracle"

# if the executables do not exist -- display error

if [ ! -f $ORACLE\_HOME/bin/dbstart -o ! -d $ORACLE\_HOME ]

then

     echo "Oracle startup: cannot start"

     exit 1

fi

# depending on parameter -- startup, shutdown, restart

# of the instance and listener or usage display

case "$1" in

 start)

     # Oracle listener and instance startup

     su $ORA\_OWNR -lc $ORACLE\_HOME/bin/dbstart

     echo "Oracle dbserver Start Succesful!OK."

     su $ORA\_OWNR -lc "$ORACLE\_HOME/bin/emctl start dbconsole"

     echo "Oracle dbconsole Start Succesful!OK."

     su $ORA\_OWNR -lc "$ORACLE\_HOME/bin/lsnrctl start"

     echo "Oracle Listeners Start Succesful!OK."

     ;;

 stop)

     # Oracle listener and instance shutdown

     su $ORA\_OWNR -lc $ORACLE\_HOME/bin/dbshut

     echo "Oracle Stop Succesful!OK."

     ;;

 reload|restart)

     $0 stop

     $0 start

     ;;

 \*)

     echo $"Usage: `basename $0` {start|stop|reload|reload}"

     exit 1

esac

exit 0

-------------------------------------------------------------------------------------------------------------

 在编辑完成之后，使用 :x  命令保存此文件。

赋予执行权限

[root@oracle init.d]# chmod 750 /etc/init.d/oracle

链接:

[root@oracle init.d]# ln -s /etc/init.d/oracle /etc/rc1.d/K61oracle

[root@oracle init.d]# ln -s /etc/init.d/oracle /etc/rc3.d/S61oracle

执行以下命令:

[root@oracle init.d]# chkconfig --level 345 oracle on

[root@oracle init.d]# chkconfig --add oracle         //添加到服务里

**二、去试试你安装的数据库，查看一下是否能用**

 [oracle@mylinux database]$ sqlplus / as sysdba

SQL\*Plus: Release 11.2.0.3.0 Production on Sat Nov 5 05:50:34 2011

Copyright (c) 1982, 2011, Oracle.  All rights reserved.

Connected to:

 Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 – Production

 With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> select \* from v$version;

BANNER

 ——————————————————————————–

 Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 – Production

 PL/SQL Release 11.2.0.3.0 – Production

 CORE    11.2.0.3.0      Production

 TNS for Linux: Version 11.2.0.3.0 – Production

 NLSRTL Version 11.2.0.3.0 – Production

//你也可以用账号scott 进行登录，（注意:在安装的时候要进行解锁，设置密码，和windows中安装是进行解锁操作一样）

登陆：      sql>conn scott/你设置密码

查看dept:  sql>select \* from dept;

-------------------------------------------

要让修改的ip永久生效就必须修改配置文件/etc/sysconfig/network-scrips/ifcfg-eth0

DEVICE=eth0

BOOTPROTO=none

HWADDR=00:13:D3:90:95:4D

ONBOOT=yes

TYPE=Ethernet

NETMASK=255.255.255.0

IPADDR=192.168.2.102

USERCTL=no

IPV6INIT=no

PEERDNS=yes

IGATEWAY=61.144.47.225

GATEWAY=61.144.47.225

 DEVICE=eth0

 BOOTPROTO=none

 BROADCAST=192.168.1.255

 IPADDR=192.168.1.33

 NETMASK=255.255.255.0

 NETWORK=192.168.1.0

 ONBOOT=yes

 USERCTL=no

 PEERDNS=no

 TYPE=Ethernet

# **7.Linux**

## 系统信息arch 显示机器的处理器架构(1)  uname -m 显示机器的处理器架构(2)  uname -r 显示正在使用的内核版本  dmidecode -q 显示硬件系统部件 - (SMBIOS / DMI)  hdparm -i /dev/hda 罗列一个磁盘的架构特性  hdparm -tT /dev/sda 在磁盘上执行测试性读取操作  cat /proc/cpuinfo 显示CPU info的信息  cat /proc/interrupts 显示中断  cat /proc/meminfo 校验内存使用  cat /proc/swaps 显示哪些swap被使用  cat /proc/version 显示内核的版本  cat /proc/net/dev 显示网络适配器及统计  cat /proc/mounts 显示已加载的文件系统  lspci -tv 罗列 PCI 设备  lsusb -tv 显示 USB 设备  date 显示系统日期  cal 2007 显示2007年的日历表  date 041217002007.00 设置日期和时间 - 月日时分年.秒  clock -w 将时间修改保存到 BIOS

## 关机（**系统的关机、重启以及登出 )** shutdown -h now 关闭系统(1)  init 0 关闭系统(2)  telinit 0 关闭系统(3)  shutdown -h hours:minutes & 按预定时间关闭系统  shutdown -c 取消按预定时间关闭系统  shutdown -r now 重启(1)  reboot 重启(2)  logout 注销

## 文件和目录

## cd /home 进入 '/ home' 目录'  cd .. 返回上一级目录  cd ../.. 返回上两级目录  cd 进入个人的主目录  cd ~user1 进入个人的主目录  cd - 返回上次所在的目录  pwd 显示工作路径  ls 查看目录中的文件  ls -F 查看目录中的文件  ls -l 显示文件和目录的详细资料  ls -a 显示隐藏文件  ls \*[0-9]\* 显示包含数字的文件名和目录名  tree 显示文件和目录由根目录开始的树形结构(1)  lstree 显示文件和目录由根目录开始的树形结构(2)  mkdir dir1 创建一个叫做 'dir1' 的目录'  mkdir dir1 dir2 同时创建两个目录  mkdir -p /tmp/dir1/dir2 创建一个目录树  rm -f file1 删除一个叫做 'file1' 的文件'  rmdir dir1 删除一个叫做 'dir1' 的目录'  rm -rf dir1 删除一个叫做 'dir1' 的目录并同时删除其内容  rm -rf dir1 dir2 同时删除两个目录及它们的内容  mv dir1 new\_dir 重命名/移动 一个目录  cp file1 file2 复制一个文件  cp dir/\* . 复制一个目录下的所有文件到当前工作目录  cp -a /tmp/dir1 . 复制一个目录到当前工作目录  cp -a dir1 dir2 复制一个目录  ln -s file1 lnk1 创建一个指向文件或目录的软链接  ln file1 lnk1 创建一个指向文件或目录的物理链接  touch -t 0712250000 file1 修改一个文件或目录的时间戳 - (YYMMDDhhmm)  file file1 outputs the mime type of the file as text  iconv -l 列出已知的编码  iconv -f fromEncoding -t toEncoding inputFile > outputFile creates a new from the given input file by assuming it is encoded in fromEncoding and converting it to toEncoding.  find . -maxdepth 1 -name \*.jpg -print -exec convert "{}" -resize 80x60 "thumbs/{}" \; batch resize files in the current directory and send them to a thumbnails directory (requires convert from Imagemagick)  **文件搜索** find / -name file1 从 '/' 开始进入根文件系统搜索文件和目录  find / -user user1 搜索属于用户 'user1' 的文件和目录  find /home/user1 -name \\*.bin 在目录 '/ home/user1' 中搜索带有'.bin' 结尾的文件  find /usr/bin -type f -atime +100 搜索在过去100天内未被使用过的执行文件  find /usr/bin -type f -mtime -10 搜索在10天内被创建或者修改过的文件  find / -name \\*.rpm -exec chmod 755 '{}' \; 搜索以 '.rpm' 结尾的文件并定义其权限  find / -xdev -name \\*.rpm 搜索以 '.rpm' 结尾的文件，忽略光驱、捷盘等可移动设备  locate \\*.ps 寻找以 '.ps' 结尾的文件 - 先运行 'updatedb' 命令  whereis halt 显示一个二进制文件、源码或man的位置  which halt 显示一个二进制文件或可执行文件的完整路径  挂载一个文件系统  mount /dev/hda2 /mnt/hda2 挂载一个叫做hda2的盘 - 确定目录 '/ mnt/hda2' 已经存在  umount /dev/hda2 卸载一个叫做hda2的盘 - 先从挂载点 '/ mnt/hda2' 退出  fuser -km /mnt/hda2 当设备繁忙时强制卸载  umount -n /mnt/hda2 运行卸载操作而不写入 /etc/mtab 文件- 当文件为只读或当磁盘写满时非常有用  mount /dev/fd0 /mnt/floppy 挂载一个软盘  mount /dev/cdrom /mnt/cdrom 挂载一个cdrom或dvdrom  mount /dev/hdc /mnt/cdrecorder 挂载一个cdrw或dvdrom  mount /dev/hdb /mnt/cdrecorder 挂载一个cdrw或dvdrom  mount -o loop file.iso /mnt/cdrom 挂载一个文件或ISO镜像文件  mount -t vfat /dev/hda5 /mnt/hda5 挂载一个Windows FAT32文件系统  mount /dev/sda1 /mnt/usbdisk 挂载一个usb 捷盘或闪存设备  mount -t smbfs -o username=user,password=pass //WinClient/share /mnt/share 挂载一个windows网络共享  **磁盘空间** df -h 显示已经挂载的分区列表  ls -lSr |more 以尺寸大小排列文件和目录  du -sh dir1 估算目录 'dir1' 已经使用的磁盘空间'  du -sk \* | sort -rn 以容量大小为依据依次显示文件和目录的大小  rpm -q -a --qf '%10{SIZE}t%{NAME}n' | sort -k1,1n 以大小为依据依次显示已安装的rpm包所使用的空间 (fedora, redhat类系统)  dpkg-query -W -f='${Installed-Size;10}t${Package}n' | sort -k1,1n 以大小为依据显示已安装的deb包所使用的空间 (ubuntu, debian类系统)  ****用户和群组**** groupadd group\_name 创建一个新用户组  groupdel group\_name 删除一个用户组  groupmod -n new\_group\_name old\_group\_name 重命名一个用户组  useradd -c "Name Surname " -g admin -d /home/user1 -s /bin/bash user1 创建一个属于 "admin" 用户组的用户  useradd user1 创建一个新用户  userdel -r user1 删除一个用户 ( '-r' 排除主目录)  usermod -c "User FTP" -g system -d /ftp/user1 -s /bin/nologin user1 修改用户属性  passwd 修改口令  passwd user1 修改一个用户的口令 (只允许root执行)  chage -E 2005-12-31 user1 设置用户口令的失效期限  pwck 检查 '/etc/passwd' 的文件格式和语法修正以及存在的用户  grpck 检查 '/etc/passwd' 的文件格式和语法修正以及存在的群组  newgrp group\_name 登陆进一个新的群组以改变新创建文件的预设群组  ****文件的权限 - 使用 "+" 设置权限，使用 "-" 用于取消**** ls -lh 显示权限  ls /tmp | pr -T5 -W$COLUMNS 将终端划分成5栏显示  chmod ugo+rwx directory1 设置目录的所有人(u)、群组(g)以及其他人(o)以读（r ）、写(w)和执行(x)的权限  chmod go-rwx directory1 删除群组(g)与其他人(o)对目录的读写执行权限  chown user1 file1 改变一个文件的所有人属性  chown -R user1 directory1 改变一个目录的所有人属性并同时改变改目录下所有文件的属性  chgrp group1 file1 改变文件的群组  chown user1:group1 file1 改变一个文件的所有人和群组属性  find / -perm -u+s 罗列一个系统中所有使用了SUID控制的文件  chmod u+s /bin/file1 设置一个二进制文件的 SUID 位 - 运行该文件的用户也被赋予和所有者同样的权限  chmod u-s /bin/file1 禁用一个二进制文件的 SUID位  chmod g+s /home/public 设置一个目录的SGID 位 - 类似SUID ，不过这是针对目录的  chmod g-s /home/public 禁用一个目录的 SGID 位  chmod o+t /home/public 设置一个文件的 STIKY 位 - 只允许合法所有人删除文件  chmod o-t /home/public 禁用一个目录的 STIKY 位  ****文件的特殊属性 - 使用 "+" 设置权限，使用 "-" 用于取消**** chattr +a file1 只允许以追加方式读写文件  chattr +c file1 允许这个文件能被内核自动压缩/解压  chattr +d file1 在进行文件系统备份时，dump程序将忽略这个文件  chattr +i file1 设置成不可变的文件，不能被删除、修改、重命名或者链接  chattr +s file1 允许一个文件被安全地删除  chattr +S file1 一旦应用程序对这个文件执行了写操作，使系统立刻把修改的结果写到磁盘  chattr +u file1 若文件被删除，系统会允许你在以后恢复这个被删除的文件  lsattr 显示特殊的属性  ****打包和压缩文件**** bunzip2 file1.bz2 解压一个叫做 'file1.bz2'的文件  bzip2 file1 压缩一个叫做 'file1' 的文件  gunzip file1.gz 解压一个叫做 'file1.gz'的文件  gzip file1 压缩一个叫做 'file1'的文件  gzip -9 file1 最大程度压缩  rar a file1.rar test\_file 创建一个叫做 'file1.rar' 的包  rar a file1.rar file1 file2 dir1 同时压缩 'file1', 'file2' 以及目录 'dir1'  rar x file1.rar 解压rar包  unrar x file1.rar 解压rar包  tar -cvf archive.tar file1 创建一个非压缩的 tarball  tar -cvf archive.tar file1 file2 dir1 创建一个包含了 'file1', 'file2' 以及 'dir1'的档案文件  tar -tf archive.tar 显示一个包中的内容  tar -xvf archive.tar 释放一个包  tar -xvf archive.tar -C /tmp 将压缩包释放到 /tmp目录下  tar -cvfj archive.tar.bz2 dir1 创建一个bzip2格式的压缩包  tar -jxvf archive.tar.bz2 解压一个bzip2格式的压缩包  tar -cvfz archive.tar.gz dir1 创建一个gzip格式的压缩包  tar -zxvf archive.tar.gz 解压一个gzip格式的压缩包  zip file1.zip file1 创建一个zip格式的压缩包  zip -r file1.zip file1 file2 dir1 将几个文件和目录同时压缩成一个zip格式的压缩包  unzip file1.zip 解压一个zip格式压缩包  ****RPM 包 - （Fedora, Redhat及类似系统）****  rpm -ivh package.rpm 安装一个rpm包  rpm -ivh --nodeeps package.rpm 安装一个rpm包而忽略依赖关系警告  rpm -U package.rpm 更新一个rpm包但不改变其配置文件  rpm -F package.rpm 更新一个确定已经安装的rpm包  rpm -e package\_name.rpm 删除一个rpm包  rpm -qa 显示系统中所有已经安装的rpm包  rpm -qa | grep httpd 显示所有名称中包含 "httpd" 字样的rpm包  rpm -qi package\_name 获取一个已安装包的特殊信息  rpm -qg "System Environment/Daemons" 显示一个组件的rpm包  rpm -ql package\_name 显示一个已经安装的rpm包提供的文件列表  rpm -qc package\_name 显示一个已经安装的rpm包提供的配置文件列表  rpm -q package\_name --whatrequires 显示与一个rpm包存在依赖关系的列表  rpm -q package\_name --whatprovides 显示一个rpm包所占的体积  rpm -q package\_name --scripts 显示在安装/删除期间所执行的脚本l  rpm -q package\_name --changelog 显示一个rpm包的修改历史  rpm -qf /etc/httpd/conf/httpd.conf 确认所给的文件由哪个rpm包所提供  rpm -qp package.rpm -l 显示由一个尚未安装的rpm包提供的文件列表  rpm --import /media/cdrom/RPM-GPG-KEY 导入公钥数字证书  rpm --checksig package.rpm 确认一个rpm包的完整性  rpm -qa gpg-pubkey 确认已安装的所有rpm包的完整性  rpm -V package\_name 检查文件尺寸、 许可、类型、所有者、群组、MD5检查以及最后修改时间  rpm -Va 检查系统中所有已安装的rpm包- 小心使用  rpm -Vp package.rpm 确认一个rpm包还未安装  rpm2cpio package.rpm | cpio --extract --make-directories \*bin\* 从一个rpm包运行可执行文件  rpm -ivh /usr/src/redhat/RPMS/`arch`/package.rpm 从一个rpm源码安装一个构建好的包  rpmbuild --rebuild package\_name.src.rpm 从一个rpm源码构建一个 rpm 包  ****YUM 软件包升级器 - （Fedora, RedHat及类似系统）**** yum install package\_name 下载并安装一个rpm包  yum localinstall package\_name.rpm 将安装一个rpm包，使用你自己的软件仓库为你解决所有依赖关系  yum update package\_name.rpm 更新当前系统中所有安装的rpm包  yum update package\_name 更新一个rpm包  yum remove package\_name 删除一个rpm包  yum list 列出当前系统中安装的所有包  yum search package\_name 在rpm仓库中搜寻软件包  yum clean packages 清理rpm缓存删除下载的包  yum clean headers 删除所有头文件  yum clean all 删除所有缓存的包和头文件  ****DEB 包 (Debian, Ubuntu 以及类似系统)**** dpkg -i package.deb 安装/更新一个 deb 包  dpkg -r package\_name 从系统删除一个 deb 包  dpkg -l 显示系统中所有已经安装的 deb 包  dpkg -l | grep httpd 显示所有名称中包含 "httpd" 字样的deb包  dpkg -s package\_name 获得已经安装在系统中一个特殊包的信息  dpkg -L package\_name 显示系统中已经安装的一个deb包所提供的文件列表  dpkg --contents package.deb 显示尚未安装的一个包所提供的文件列表  dpkg -S /bin/ping 确认所给的文件由哪个deb包提供  ****APT 软件工具 (Debian, Ubuntu 以及类似系统)**** apt-get install package\_name 安装/更新一个 deb 包  apt-cdrom install package\_name 从光盘安装/更新一个 deb 包  apt-get update 升级列表中的软件包  apt-get upgrade 升级所有已安装的软件  apt-get remove package\_name 从系统删除一个deb包  apt-get check 确认依赖的软件仓库正确  apt-get clean 从下载的软件包中清理缓存  apt-cache search searched-package 返回包含所要搜索字符串的软件包名称  ****查看文件内容****  cat file1 从第一个字节开始正向查看文件的内容  tac file1 从最后一行开始反向查看一个文件的内容  more file1 查看一个长文件的内容  less file1 类似于 'more' 命令，但是它允许在文件中和正向操作一样的反向操作  head -2 file1 查看一个文件的前两行  tail -2 file1 查看一个文件的最后两行  tail -f /var/log/messages 实时查看被添加到一个文件中的内容  ****文本处理**** cat file1 file2 ... | command <> file1\_in.txt\_or\_file1\_out.txt general syntax for text manipulation using PIPE, STDIN and STDOUT  cat file1 | command( sed, grep, awk, grep, etc...) > result.txt 合并一个文件的详细说明文本，并将简介写入一个新文件中  cat file1 | command( sed, grep, awk, grep, etc...) >> result.txt 合并一个文件的详细说明文本，并将简介写入一个已有的文件中  grep Aug /var/log/messages 在文件 '/var/log/messages'中查找关键词"Aug"  grep ^Aug /var/log/messages 在文件 '/var/log/messages'中查找以"Aug"开始的词汇  grep [0-9] /var/log/messages 选择 '/var/log/messages' 文件中所有包含数字的行  grep Aug -R /var/log/\* 在目录 '/var/log' 及随后的目录中搜索字符串"Aug"  sed 's/stringa1/stringa2/g' example.txt 将example.txt文件中的 "string1" 替换成 "string2"  sed '/^$/d' example.txt 从example.txt文件中删除所有空白行  sed '/ \*#/d; /^$/d' example.txt 从example.txt文件中删除所有注释和空白行  echo 'esempio' | tr '[:lower:]' '[:upper:]' 合并上下单元格内容  sed -e '1d' result.txt 从文件example.txt 中排除第一行  sed -n '/stringa1/p' 查看只包含词汇 "string1"的行  sed -e 's/ \*$//' example.txt 删除每一行最后的空白字符  sed -e 's/stringa1//g' example.txt 从文档中只删除词汇 "string1" 并保留剩余全部  sed -n '1,5p;5q' example.txt 查看从第一行到第5行内容  sed -n '5p;5q' example.txt 查看第5行  sed -e 's/00\*/0/g' example.txt 用单个零替换多个零  cat -n file1 标示文件的行数  cat example.txt | awk 'NR%2==1' 删除example.txt文件中的所有偶数行  echo a b c | awk '{print $1}' 查看一行第一栏  echo a b c | awk '{print $1,$3}' 查看一行的第一和第三栏  paste file1 file2 合并两个文件或两栏的内容  paste -d '+' file1 file2 合并两个文件或两栏的内容，中间用"+"区分  sort file1 file2 排序两个文件的内容  sort file1 file2 | uniq 取出两个文件的并集(重复的行只保留一份)  sort file1 file2 | uniq -u 删除交集，留下其他的行  sort file1 file2 | uniq -d 取出两个文件的交集(只留下同时存在于两个文件中的文件)  comm -1 file1 file2 比较两个文件的内容只删除 'file1' 所包含的内容  comm -2 file1 file2 比较两个文件的内容只删除 'file2' 所包含的内容  comm -3 file1 file2 比较两个文件的内容只删除两个文件共有的部分  ****字符设置和文件格式转换**** dos2unix filedos.txt fileunix.txt 将一个文本文件的格式从MSDOS转换成UNIX  unix2dos fileunix.txt filedos.txt 将一个文本文件的格式从UNIX转换成MSDOS  recode ..HTML < page.txt > page.html 将一个文本文件转换成html  recode -l | more 显示所有允许的转换格式  ****文件系统分析**** badblocks -v /dev/hda1 检查磁盘hda1上的坏磁块  fsck /dev/hda1 修复/检查hda1磁盘上linux文件系统的完整性  fsck.ext2 /dev/hda1 修复/检查hda1磁盘上ext2文件系统的完整性  e2fsck /dev/hda1 修复/检查hda1磁盘上ext2文件系统的完整性  e2fsck -j /dev/hda1 修复/检查hda1磁盘上ext3文件系统的完整性  fsck.ext3 /dev/hda1 修复/检查hda1磁盘上ext3文件系统的完整性  fsck.vfat /dev/hda1 修复/检查hda1磁盘上fat文件系统的完整性  fsck.msdos /dev/hda1 修复/检查hda1磁盘上dos文件系统的完整性  dosfsck /dev/hda1 修复/检查hda1磁盘上dos文件系统的完整性  ****初始化一个文件系统****  mkfs /dev/hda1 在hda1分区创建一个文件系统  mke2fs /dev/hda1 在hda1分区创建一个linux ext2的文件系统  mke2fs -j /dev/hda1 在hda1分区创建一个linux ext3(日志型)的文件系统  mkfs -t vfat 32 -F /dev/hda1 创建一个 FAT32 文件系统  fdformat -n /dev/fd0 格式化一个软盘  mkswap /dev/hda3 创建一个swap文件系统  ****SWAP文件系统****  mkswap /dev/hda3 创建一个swap文件系统  swapon /dev/hda3 启用一个新的swap文件系统  swapon /dev/hda2 /dev/hdb3 启用两个swap分区  ****备份****  dump -0aj -f /tmp/home0.bak /home 制作一个 '/home' 目录的完整备份  dump -1aj -f /tmp/home0.bak /home 制作一个 '/home' 目录的交互式备份  restore -if /tmp/home0.bak 还原一个交互式备份  rsync -rogpav --delete /home /tmp 同步两边的目录  rsync -rogpav -e ssh --delete /home ip\_address:/tmp 通过SSH通道rsync  rsync -az -e ssh --delete ip\_addr:/home/public /home/local 通过ssh和压缩将一个远程目录同步到本地目录  rsync -az -e ssh --delete /home/local ip\_addr:/home/public 通过ssh和压缩将本地目录同步到远程目录  dd bs=1M if=/dev/hda | gzip | ssh user@ip\_addr 'dd of=hda.gz' 通过ssh在远程主机上执行一次备份本地磁盘的操作  dd if=/dev/sda of=/tmp/file1 备份磁盘内容到一个文件  tar -Puf backup.tar /home/user 执行一次对 '/home/user' 目录的交互式备份操作  ( cd /tmp/local/ && tar c . ) | ssh -C user@ip\_addr 'cd /home/share/ && tar x -p' 通过ssh在远程目录中复制一个目录内容  ( tar c /home ) | ssh -C user@ip\_addr 'cd /home/backup-home && tar x -p' 通过ssh在远程目录中复制一个本地目录  tar cf - . | (cd /tmp/backup ; tar xf - ) 本地将一个目录复制到另一个地方，保留原有权限及链接  find /home/user1 -name '\*.txt' | xargs cp -av --target-directory=/home/backup/ --parents 从一个目录查找并复制所有以 '.txt' 结尾的文件到另一个目录  find /var/log -name '\*.log' | tar cv --files-from=- | bzip2 > log.tar.bz2 查找所有以 '.log' 结尾的文件并做成一个bzip包  dd if=/dev/hda of=/dev/fd0 bs=512 count=1 做一个将 MBR (Master Boot Record)内容复制到软盘的动作  dd if=/dev/fd0 of=/dev/hda bs=512 count=1 从已经保存到软盘的备份中恢复MBR内容  ****光盘**** cdrecord -v gracetime=2 dev=/dev/cdrom -eject blank=fast -force 清空一个可复写的光盘内容  mkisofs /dev/cdrom > cd.iso 在磁盘上创建一个光盘的iso镜像文件  mkisofs /dev/cdrom | gzip > cd\_iso.gz 在磁盘上创建一个压缩了的光盘iso镜像文件  mkisofs -J -allow-leading-dots -R -V "Label CD" -iso-level 4 -o ./cd.iso data\_cd 创建一个目录的iso镜像文件  cdrecord -v dev=/dev/cdrom cd.iso 刻录一个ISO镜像文件  gzip -dc cd\_iso.gz | cdrecord dev=/dev/cdrom - 刻录一个压缩了的ISO镜像文件  mount -o loop cd.iso /mnt/iso 挂载一个ISO镜像文件  cd-paranoia -B 从一个CD光盘转录音轨到 wav 文件中  cd-paranoia -- "-3" 从一个CD光盘转录音轨到 wav 文件中（参数-3）  cdrecord --scanbus 扫描总线以识别scsi通道  dd if=/dev/hdc | md5sum 校验一个设备的md5sum编码，例如一张 CD  **网络 - （以太网和WIFI无线）** ifconfig eth0 显示一个以太网卡的配置  ifup eth0 启用一个 'eth0' 网络设备  ifdown eth0 禁用一个 'eth0' 网络设备  ifconfig eth0 192.168.1.1 netmask 255.255.255.0 控制IP地址  ifconfig eth0 promisc 设置 'eth0' 成混杂模式以嗅探数据包 (sniffing)  dhclient eth0 以dhcp模式启用 'eth0'  route -n show routing table  route add -net 0/0 gw IP\_Gateway configura default gateway  route add -net 192.168.0.0 netmask 255.255.0.0 gw 192.168.1.1 configure static route to reach network '192.168.0.0/16'  route del 0/0 gw IP\_gateway remove static route  echo "1" > /proc/sys/net/ipv4/ip\_forward activate ip routing  hostname show hostname of system  host www.example.com lookup hostname to resolve name to ip address and viceversa(1)  nslookup www.example.com lookup hostname to resolve name to ip address and viceversa(2)  ip link show show link status of all interfaces  mii-tool eth0 show link status of 'eth0'  ethtool eth0 show statistics of network card 'eth0'  netstat -tup show all active network connections and their PID  netstat -tupl show all network services listening on the system and their PID  tcpdump tcp port 80 show all HTTP traffic  iwlist scan show wireless networks  iwconfig eth1 show configuration of a wireless network card  hostname show hostname  host www.example.com lookup hostname to resolve name to ip address and viceversa  nslookup www.example.com lookup hostname to resolve name to ip address and viceversa  whois www.example.com lookup