



Linux基础

Helight@QQ.com



1 Linux概述

Linux的性质

Linux为一种自由软件，是一种真正多任务和多用户的网络操作系统。

软件分类

商业软件：用户必须向发行商购软件的许可证，用户只具有使用权、用户不能无法复制和拷贝。

共享软件：用户可以免费使用软件的试用版，在试用期间用户可以复制，但是使用结束后用户必须购买软件的许可证。

自由软件：用户可以免费的永远使用软件且可以任意复制。



GNU是自由软件的积极倡导者Richard Stallman组织开发的一个完全基于自由软件的软件体系。

其标志头像如图。GNU本意是非洲大陆的一种牛羚。



为保证GNU软件可以自由地“使用、复制、修改和发布”，所有GNU软件都遵守一份协议条款，即GNU通用公共许可证（GNU General Public License, GPL）。

GPL 是Richard Stallman在开放源代码软件发行的实践中，逐渐总结出的一套保护自由软件的条款。当人们提起商业软件版权时，总会用到Copyright，而在GPL中，人们则使用“CopyLeft”（“反版权”）。Copyright禁止用户对所购得软件拷贝、更改，禁止用户对软件重分发等；而GPL对软件的限制同商业软件完全不同，所以人们习惯上用CopyLeft来表示自由软件的特性。



Linux操作系统及其上的大部分软件在版权上使用自由软件基金会的GNU（**General Public License**）通用公共许可证（**GPL**）。该版权的使用方法的两段话：

- （1）您有权采用收费或免费的方式发布该软件，但您也必须告知您的收受者，他们同样拥有此权利，即用收费或免费的方式再次发送，您不得要求收受者放弃此权利。
- （2）您可以取得源代码并作您喜欢的修改，也可将之出售图利，但您的收受人若要求您提供源代码，您不能拒绝，或至少告知其在何处可以找到源代码。您的收受人也可将该源代码进行修改或出售，您不得要求收受人放弃此权利。



Linux与NT/2000/2003的主要差别

(1) 多用户方面

Linux: 同时允许多个用户、多个桌面

Win NT/2k/2003: 同时只能允许一个用户、一个桌面

(2) GUI界面方面

Linux: GUI界面采用x-windows且与内核是相互独立

Win NT/2k/2003: GUI与内核是集成在一起的

(3) 共享资源相互访问

Linux: NFS、Samba实现Linux及Windows主机之间相互访问

Win NT/2k/2003: 利用网上邻居实现相互访问

(4) 配置文件

Linux: 有多个配置文件

Win NT/2k/2003: 所有配置集中在注册表中



Linux操作系统及其上的大部分软件在版权上使用自由软件基金会的GNU（**General Public License**）通用公共许可证（**GPL**）。该版权的使用方法的两段话：

- （1）您有权采用收费或免费的方式发布该软件，但您也必须告知您的收受者，他们同样拥有此权利，即用收费或免费的方式再次发送，您不得要求收受者放弃此权利。
- （2）您可以取得源代码并作您喜欢的修改，也可将之出售图利，但您的收受人若要求您提供源代码，您不能拒绝，或至少告知其在何处可以找到源代码。您的收受人也可将该源代码进行修改或出售，您不得要求收受人放弃此权利。



Linux常见发行版本

- RedHat Linux
- fedora
- centOS
- SUSE Linux
- Mandarke Linux
- Turbo Linux
- 红旗Linux
- ...



- **Fedora**和**Redhat**这两个**Linux**的发行版联系很密切。**Redhat**自**9.0**以后，不再发布桌面版的，而是把这个项目与开源社区合作，于是就有了**Fedora**这个**Linux**发行版。**Fedora**可以说是**Redhat**桌面版本的延续，只不过是与开源社区合作。
- **Redhat**的**Fedora Core Linux**和**Enterprise Linux**，都需要遵循**GNU**协议即需要发布自己的源代码。

所以，对于免费的**Fedora Core Linux**，从**Redhat**网站上可以直接下载**ISO**刻盘，还能下载到**SRPM**的**ISO**，即程序包源码光盘。对于收费的**Enterprise Linux**系列，是一款商业产品，所以网站上不能下载到**ISO**文件，需要购买正式授权方可。由于**Enterprise Linux**也需要遵循**GNU**协议，故必须发布源代码。所以在**Redhat**的网站上，可以获得**AS/ES/WS**系列的**SRPM**源码**ISO**文件。这些文件可以被自由的下载，修改代码，重新编译使用。

CentOS社区将**Redhat**的网站上的所有源代码下载下来，进行重新编译。重新编译后，由于**AS/ES/WS**是商业产品，必须将所有**Redhat**的**Logo**和标识改成自己的**CentOS**标识。比如将**AS4**原版的**SRPM**源码编译后，就成为了**CentOS 4.0**。**AS4Update1**的源码编译后，就成为了**CentOS4.1**。**AS4Update2**的源码编译后，就成为了**CentOS4.2**。同理，**CentOS**的**3.x/4.x**都对应着相应的版本。

所以我们说，**CentOS**就是**Redhat**的**AS/ES/WS**的免费版本。使用**CentOS**，可以获得和**AS/ES**相同的性能和感受。



2、Linux基本操作命令

- 文件和目录操作的命令
- 文件归档和压缩的命令
- 用户和用户组管理命令
- 系统管理命令



1. 文件系统

指的计算机系统内的文件和目录的集合。

2. Linux目录结构如图:

/bin: 基本系统程序

/boot: 启动相关文件

/dev: 设备文件

/etc: 配置文件

/home: 所有用户的个人主目录

/lib: 库文件

/mnt: 临时挂装其他文件系统

/proc: 虚拟文件系统, 其文件放在内存而非磁盘。

/root: 超级用户主目录。

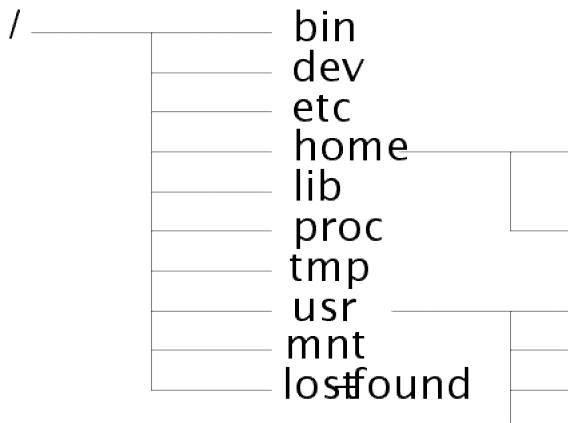
/sbin: 系统管理员所使用的可执行文件

/tmp: 程序运行时的临时信息。

/usr: 软件包程序、系统源代码等

/var: 大小经常发生变化的文件, 如日志文件等。

/lost+found: 系统发生故障时, 存放找不到正确存储位置的文





ls:查看指定目录下的内容(文件和目录)

格式:ls 参数 name

- a 显示所有文件及目录 (ls内定将档案名或目录名称开头为“.”的视为隐藏文件，不会列出)
- l 除文件名称外，并将文件类型、权限、拥有者、文件大小等信息详细列出
- t 将文件依建立时间之先后次序列出
- F 在列出的文件名称后加一符号；例如可执行文件则加 “*”，目录则加 “/”
- R 若目录下有文件，则以下的文件都依序列出

例:

`ll#ls -al /home`(列出目录/home 下的所有文件的详细信息,包括隐藏文件)



3. 文件颜色的区别

白色：普通文件

红色：压缩文件

蓝色：目录文件

青绿色：链接文件

黄色：设备文件（/dev）

绿色：可执行文件（/bin、/sbin）

粉红色：图片文件

4. 文件属性、权限



```
[ ]#ls -l /bin/ls
```

```
-rwxr-xr-x 1 root root 67668 Feb 19 2003
```

```
/bin/ls
```

文件名

文件最后修改的日期和时间

文件的字节数

共享该文件访问的用户组组名

文件属主的登陆名

链接数

第1位字符:文件类型

d:目录;**l**:表明是链接;**-**:普通文件

第2到10位:权限

文件属主权限: **rwX** (2.3.4位)文件属主所在组成员权限: **rwX** (5.6.7位)其他成员权限: **rwX** (8.9.10位)

```
[root@localhost ~]# ll /bin/ls
-rwxr-xr-x 1 root root 97972 2007-10-31 /bin/ls
[root@localhost ~]#
```



cd :改变当前目录

格式:**cd <dir>** 将当前目录切换到**dir**

- ❑ 其中 **dir**的名字表示法可为绝对路径或相对路径。若目录名称省略，则变换至使用者的 **home directory** (也就是刚 **login** 时所在的目录)。
- ❑ 另外，"**~**" 也表示为 **home directory** 的意思，"**.**" 则是表示目前所在的目录，"**..**" 则表示目前目录位置的上一层目录。
- ❑ 范例：跳到 **/usr/bin/**：
cd /usr/bin
- ❑ 跳到自己的 **home directory**：
cd ~
- ❑ 跳到目前目录的上上两层：
cd ../..



pwd 察看当前路径

例

```
[root@localhost ~]# pwd
/root
[root@localhost ~]# cd /home
[root@localhost home]# pwd
/home
```

man命令

察看命令或函数的用法。



chmod :更改文件访问权限

格式：**chmod** [参数] **mode filename**

mode : 权限设定字串，有两种方式：

(1) [u_go] [+₋=] [rwx]

u 表示该文件的拥有者，**g** 表示与该文件的拥有者属于同一个组(**group**)，**o** 表示其他以外的人，**a** 表示这三者皆是。

+ 表示增加权限、**-** 表示取消权限、**=** 表示唯一设定权限(只设定一种)。

r 表示可读取，**w** 表示可写入，**x** 表示可执行。



- ❑ 将文件 `file1.txt` 设为所有人皆可读取：
`chmod ugo+r file1.txt` 或
`chmod a+r file1.txt`
- ❑ 将文件 `file1.txt` 与 `file2.txt` 设为:该文件拥有者和其同一组的成员可写入，但其他以外的人不可写入：
`chmod ug+w,o-w file1.txt file2.txt`
- ❑ 将 `ex1` 设定为只有该文件拥有者可以执行：
`chmod u+x,go-x ex1`



参数: **-R**对目前目录下的所有档案与子目录进行相同的权限变更(即以递归的方式逐个变更)

▣将目前目录下的所有文件与子目录皆设为任何人可读取：

chmod -R a+r*

```
[root@localhost wjl]# pwd
/home/wjl
[root@localhost wjl]# ll
总计 15756
-rw----- 1 root root 16100505 2005-12-16 VMwareTools-5.5.1-19175.tar.gz
drwx--x--x 7 root root 4096 2005-12-16 vmware-tools-distrib
[root@localhost wjl]# cd ..
[root@localhost home]# ll
总计 8
drwx----- 5 wjl wjl 4096 07-20 00:06 wjl

[root@localhost /]# cd /home
[root@localhost home]# chmod -R a+r *
[root@localhost home]# ll
总计 8
drwx--r-- 5 wjl wjl 4096 07-20 00:06 wjl
[root@localhost home]# cd wjl
[root@localhost wjl]# ll
总计 15756
-rw-r--r-- 1 root root 16100505 2005-12-16 VMwareTools-5.5.1-19175.tar.gz
drwxr-xr-x 7 root root 4096 2005-12-16 vmware-tools-distrib
```



(2)用数字来表示权限

格式: **chmod abc file**

r=4, w=2, x=1

若要**rw**x属性则 $4+2+1=7$;

若要**rw**-属性则 $4+2=6$;

若要**r**-x属性则 $4+1=5$ 。

例:

chmod 777 file (chmod a=rwx file)

chmod 771 file (chmod ug=rwx,o=x file)



chown:改变文件或目录的拥有者.

格式: `chmod[参数] user[:group] file`

说明: user : 新的文件拥有者的使用者 ID

group : 新的文件拥有者的使用者组名 (group)

-R : 对目前目录下的所有档案与子目录进行相同的拥有者变更
(即以递归的方式逐个变更)

例 :

1. 将file1.txt 的拥有者设为 guoup1组的rose :

```
chown rose:group1 file1.txt
```

2. 将目前目录下的所有文件与子目录的拥有者皆设为group1组的使用者
rose :

```
chmod -R rose:group1 *
```



chgrp: 修改文件或目录所属的用户组.

格式: `chmod[参数] 用户组名 文件或目录`

说明:

-R : 对目前目录下的所有档案与子目录进行相同的拥有者变更 (即以递归的方式逐个变更)

例 :

将file1.txt 所属组设为 guoup1 :

```
chgrp group1 file1.txt
```



rm : 删除文件及目录。

格式: **rm [options] name...**

说明:

- f 直接删除, 无需逐一确认, 即使原文件属性设为只读。
- r 将目录及以下的文件逐一删除。

例:

1. 删除所有C语言文件, 删除前逐一询问确认 :

```
rm *.c
```

2. 将 Finished 子目录及子目录中所有文件删除 :

```
rm -r Finished (rm -rf Finished)
```

2010-1-11



cp:将一个文件拷贝至另一文件,
或将一个或数个文件拷贝至另一目录。

格式: **cp [options] source dest**
cp [options] source... directory

说明:

- r 若 **source**是目录, 则将目录复制至目的地。
- f 若目的地已经有相同文件名的文件存在, 则在复制前先予以删除再行复制。

例:

将文件 **aaa** 复制(已存在), 并命名为 **bbb** :

```
cp aaa bbb
```

将所有的C语言文件拷贝至 **Finished**目录中 :

```
cp *.c Finished
```



mv: 将一个文件移至另一文件(重命名), 或将数个文件移至另一目录。

格式: **mv [options] source dest**
mv [options] source... directory

例:

将档案 aaa 更名为 bbb :

```
mv aaa bbb
```

将所有的C语言文件移至/home/user1目录中 :

```
mv 2010-1-11 *.c /home/user1
```




mkdir :创建目录

格式:mkdir 目录名

例:mkdir xyz (在当前目录)

rmdir: 删除空目录

说明: **-p**删除目录下所有的空目录。如有非空子目录则保留, 如所有的子目录都空, 则删除该目录。

例: rmdir -p a/b/c 近似于:

2010-1-11 `rmdir a/b/c a/b a`



touch:改变文件的时间戳

格式: **touch** 文件名

说明: 如果文件不存在, 则创建该文件。



cat

1)显示文件内容

cat [参数] filename

-n 或 -number: 由 1 开始对所有输出的行数编号

2)连接多个文件

cat file1...fileN > fileName



more

格式: `more [-dlfpcsu] [-num] [+ / pattern]`
`[+linenum] [fileNames..]`

说明: 类似 **cat (全部显示)**, 不过会以**一页一页的显示**方便使用者**逐页阅读**, 而最基本的指令就是按空白键 (**space**) 就往下**一页**显示, 按 **b** 键就会**往回** (**back**) **一页**显示, 按**q**退出。

参数:

-num 一次显示的行数

+num 从第 **num** 行开始显示

fileNames 欲显示内容的文件, 可为复数个数

例:

more -s testfile 逐页显示 **testfile** 文件内容, 如有连续两行以上空白行则以一行空白行显示。

more +20 testfile 从第 **20** 行开始显示 **testfile** 文件内容。



less

格式: `less [Option] filename`

说明:

less 的作用与 **more** 十分相似，都可以用来浏览文本文件的内容，不同的是 **less** 允许使用者往回卷动以浏览已经看过的部分，全部显示完后不退出，同时因为 **less** 并未在一开始就读入整个文件，因此在遇上大型文件的开启时，会比一般的文本编辑器（如 **vi**）来的快速。



head

head filename 默认显示前10行

head -20 filename 显示前20行

tail

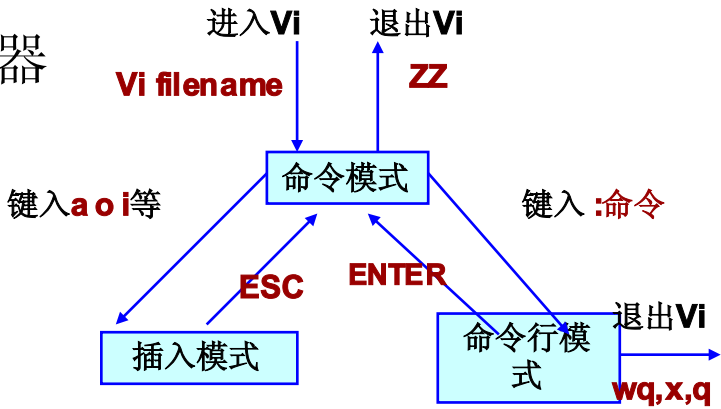
tail filename

tail -20 filename

tail -f filename



Vi编辑器





编辑模式

注意：在编辑模式下，输入的字符被看成是命令进行光标移动、剪切、拷贝、粘贴、删除、插入等编辑动作。

➤ 光标移动(区分大小写)

j,k,h,l:上,下,左,右

Ctrl+b:向上翻页

Ctrl+f: 向下翻页

H: 屏幕最上行

L: 屏幕最下行

\$:右移至本行末尾

O: 左移本行开头

gg:文件首行 G:文件末行

2010-1-11



➤ 剪切拷贝粘贴

x: 删除光标所在的字符

dd:删除光标所在行

ndd: 删除**n**行

yy: 复制当前行

nyy: 复制**n**行

p:将剪切或复制的内容粘贴到光标的下行

P:将剪切或复制的内容粘贴到光标的上行

➤ 搜索字符串

/str: 正向搜索

? str: 反向搜索

n: 搜索下一个

Shift+n: 反向搜索下一个



➤ 撤销

u: 撤销上一个命令。

.: 在新的光标位置重复最后一条命令。

插入模式:

注意: 输入以下命令后, 屏幕下方会出现
“insert”, 表示进入插入模式

i: 在光标左侧输入文字

a: 在光标右侧输入文字

o: 在光标所在行的下行增添新行

O: 在光标所在行的上行增添新行



命令模式（编辑模式下输入：）

注意：命令模式的命令均以：开始

：w 保存

：q退出

:wq保存并退出（等于：x）

：q! 不保存退出

：f 显示当前文件名、文件行数、当前光标所在的行数等信息

:%s/oldstr/newstr替换字符串

:%s/oldstr/newstr/gc替换字符串，替换前询问。



gedit

在图形模式下，可使用**gedit**软件进行编辑修改。

`[]$gedit filename`



grep:在文件中查找指定的字符串.

说明:

-r:以递归方式查询目录下的所有子目录的文件。

-i:不区分字符串的大小写。

例: **grep 'user1' /etc/passwd**

locate: 查找所有名称中包含制定字符串的文件或目录

例: **locate ipconfig**



find :目录中搜索文件

格式 **find** [起始目录][搜索条件]

说明:

- **起始目录**:指定目录路径,系统从这里开始沿着目录树向下查找文件。它是一个路径列表,相互用空格分离,如果不写起始目录,那么默认为当前目录。
- **搜索条件**
 - name ‘字符串’: 查找文件名中包含所给字符串的所有文件,支持通配符*和?。
 - atime n: 搜索在n天前访问过的文件。
 - ctime n: 搜索在n天前修改过的文件。
 - group ‘组名’: 搜索属于指定用户组的文件。
 - user ‘用户名’: 搜索属于用户名的文件。
 - size n: 搜索文件大小是n的文件。
 - type x: 搜索类型为x的文件(b块设备文件, c字符设备文件, d目录文件, f普通文件, l链接文件等)。



find命令查找文件的几种方法举例:

(1) 根据文件名查找

find / -name lilo.conf "/"表示搜索整个硬盘。

(2) 快速查找文件

根据文件名查找文件会遇到一个实际问题,就是要花费相当长的一段时间,特别是大型Linux文件系统和大容量硬盘文件放在很深的子目录中时。如果我们知道了这个文件存放在某个目录中,那么只要在这个目录中往下寻找就能节省很多时间。

find /etc -name smb.conf

(3) 根据部分文件名查找方法

有时我们知道只某个文件包含有**abvd**这4个字,那么要查找系统中所有包含有这4个字符的文件可以输入下面命令:

find / -name '*abvd*'

(4) 使用混合查找方式查找文件

find命令可以使用混合查找的方法,例如,我们想在/etc目录中查找大于**500000**字节,并且在**24**小时内修改的某个文件,则可以使用**-and**(与)把两个查找参数链接起来组合成一个混合的查找方式。

find /etc -size +500000c -and -ctime +1



文件链接

硬链接

实质 : 一个文件拥有多个有效的路径名和文件名.

实际上是同一个文件, 对一个文件进行修改等操作, 另一个文件有影响.

```
#ln test.c testlnk1.c
```

```
#ln test.c testlnk2.c
```

```
#ls -l test.c (共3个连接)
```

```
-rw-r--r-- 3 root root 4096 sep 1 20:00 test.c
```

```
#rm test.c (只删除一个文件, 当3个都删除时, 才会被真正删除)
```

局限性: 不能跨 Filesystem; 不能 link 目录。



软链接

链接文件实际是个“指针文件”,保存所指向的
文件的路径名.

#ln -s oldfile linkfile(不加参数s,则默认硬
连接)

如果删除原文件,则链接文件就会打不开.



file

file通过探测文件内容判断文件类型。

格式: **file** 文件名

说明:使用**file**命令可以知道某个文件究竟是二进制（**ELF**格式）的可执行文件, 还是**Shell Script**文件, 或者是其它的什么格式。**file**能识别的文件类型有目录、**Shell**脚本、英文文本、二进制可执行文件、**C**语言源文件、文本文件、**DOS**的可执行文件等。

例:

如果我们看到一个没有后缀的文件**file1**, 可以使用下面命令:

```
$ file file1
```

2010-1-11



tar 归档文件命令

格式: **tar** [选项] file...

-c 创建归档文件

-v 强制**tar**执行冗长模式,即在归档时打印每一个文件名.

-f 告诉**tar**下一个参数是创建的归档文件的文件名.

-x 解开归档文件

-t 查看归档文件的文件信息.

例:

```
tar cvf abc.tar /home/test
```

```
tar tvf abc.tar
```

```
tar xvf abc.tar
```

2010-1-11



压缩/解压缩文件

- **.gz**
压缩: `gzip FileName`
解压1: `gunzip FileName.gz`
解压2: `gzip -d FileName.gz`
- **.zip**
压缩: `zip FileName.zip fileName(zip -r FileName.zip dirName 压缩目录)`
解压: `unzip FileName.zip`
- **.bz2**
压缩: `bzip2 FileName`
解压1: `bunzip2 FileName.bz2`
解压2: `bzip2 -d FileName.bz2`
- **.Z**
压缩: `compress FileName`
解压: `uncompress FileName.Z`



➤ **.tar.gz**

解压: `tar zxvf FileName.tar.gz`

压缩: `tar zcvf FileName.tar.gz DirName`

➤ **.tar.bz2**

解压: `tar jxvf FileName.tar.bz2`

压缩: `tar jcvf FileName.tar.bz2 DirName`

➤ **.tar.Z**

解压: `tar Zxvf FileName.tar.Z`

压缩: `tar Zcvf FileName.tar.Z DirName`



rpm

RedHat Package Management ， 用于管理各种软件。

如：

vsftpd-1.1.3-8.i386.rpm

rpm包的文件名中包含了这个软件包的版本信息，操作系统信息，硬件要求等等



✓ 安装rpm包:

```
rpm -ivh ****.rpm
```

✓ 删除rpm包:

```
rpm -e **** (注意: 不带.rpm后缀, 版本号带不带都可以)
```

✓ 查询系统是否已安装指定的rpm包:

```
rpm -qa *** (不带版本号)
```

```
或 rpm -qa |grep "****"
```

✓ 查询系统所有安装的rpm包

```
rpm -qa
```



总结：

获得软件包相关的信息用 **rpm -q**

q表示查询**query**，后面可以跟其他选项：

i 表示**info**，获得软件包的信息；

l 表示**list**，获得文件列表；

a 表示**all**，在所有包中执行查询；

f 表示**file**，根据文件进行相关的查询；

p 表示**package**，根据软件包进行查询

rpm软件包下载：<http://rpm.pbone.net/>



用户和用户组管理

Linux系统中，每一个用户都有一个唯一的身份标识，称为用户**ID**（**UID**）。每一个用户至少属于一个用户组。用户组是由系统管理员创建的、有多个用户组成的用户群体。每一个用户组也有一个唯一的身份标识，成为用户组**ID**。不同的用户和用户组对系统拥有不同的权限。对文件或目录的访问，以及对程序的执行都需要调用者拥有相符合的身份。



用户管理包括添加用户、删除用户、修改用户属性等，相关文件包括 `/etc/passwd`、`/etc/shadow`

```
[root@localhost wjl]# ls -l /usr/sbin/useradd /usr/sbin/adduser  
-rwxrwxrwx 1 root root    7 07-20 04:52 /usr/sbin/adduser -> useradd  
-rwxr-x--- 1 root root 74612 2007-10-09 /usr/sbin/useradd
```

✓ 增加用户： `adduser (useradd) UserName`

✓ 删除用户： `deluser (userdel) UserName`

✓ 设置或修改密码： `passwd UserName` (注意：超级用户 `root` 可更改其他用户密码，普通用户只能更改自己的。)



✓ /etc/passwd 文件：保存用户相关信息

每个用户占一行，内容格式为：

用户名：密码：UID：GID：用户全名：个人主目录：登录shell

例如：

wjl: x: 500: 500: wangjingli: /home/wjl: /bin/bash

✓ /etc/shadow文件:存放加密的口令

用户名：加密的密码：从1970.1.1到上次修改密码的天数：密码的最少使用天数：密码有效期：密码失效前警告的天数：从密码过期到彻底停用的天数：账号失效日期：保留

✓ 临时禁用用户

在/etc/passwd文件中，最前面加“#”或密码前加*；

在/etc/shadow文件中，密码字段前加“*”。

例：

~~wjl~~:~~x~~: 500: 500: wangjingli: /home/wjl: /bin/bash₅₁



忘记root密码,解决方法:

以单用户方式进入系统, 修改root的密码然后再重新启动。

1.对于LILO:

当系统启动到出现LILO引导画面时, 对于图形引导方式按TAB键进入文本方式, 然后在LILO处输入linux single回车即可进入免密码的控制台, 进入以后使用passwd命令修改root的密码即可。

2.对于GRUB:

GRUB:

a). 在引导装载程序菜单上, 键入e进入编辑模式, 将会出现一个引导项目列表。

b). 查找kernel开始的行,例如:kernel /vmlinuz-2.4.18-0.4 root=/dev/hda2

按上下箭头键直到这一行被突出显示, 然后按 e 进行编辑:在文本结尾处空一格键入 single,告诉 GRUB 引导单用户 Linux 模式。

c). 按 Enter 键使编辑结果生效,并回到编辑模式屏幕, 再按b, GRUB 就会引导单用户 Linux 模式。

d).结束载入后, 将会出现一个与以下相似的 shell 提示:
sh-2.05#

e).改变root用户密码, 键入:

bash# passwd root

重新键入两遍新密码。结束后, 密码就会被改变。

d).在提示下键入 reboot 来重新引导,用新密码登陆系统。



su: 转成超级用户或用户切换

whoami: 查看当前用户的登陆名

```
[root@localhost ~]# whoami
root
[root@localhost ~]# su wjl
[wjl@localhost root]$ whoami
wjl
[wjl@localhost root]$ su root
□令 :
[root@localhost ~]# whoami
root
```



用户组管理

主要是对用户组的添加、修改、删除等操作，所涉及的文件有：`/etc/group`、`/etc/gshadow`等。

✓ **groupadd**: 添加用户组

例: `groupadd teacher`

✓ **groupdel**: 删除用户组

例: `groupdel teacher`

✓ **groupmod**: 修改用户组

`-g`: 更改用户组gid。

`-n`: 更改用户组名称。

例: `groupmod -g 506 -n teacher1 teacher`



挂载命令 (Mount) 几种格式:

```
# mount [-ahlV]
# mount -t type /dev/hdxx /mountpoint
# mount -o [options]
```

参数说明:

- a : 依照 `/etc/fstab` 的内容将所有相关的磁盘都挂上来
- h : 只列出 `mount` 相关的参数, 并不挂载任何设备
- l : 列出目前已经挂载的设备、文件系统名称与挂载点
- V : 列出 `mount` 的版本信息

type: 将后面 `/dev/hdxx` 这个设备以 **type** 的文件格式挂载到 `/mountpoint` 这个点, 常见的有:

vfat, msdos :	Windows 系统的文件格式
ext, ext2:	Linux 的主要文件格式
iso9660 :	光盘文件格式
nfs, ntfs :	Windows 2000 使用 NTFS 格式

-o : 这个参数后面的选项有:

rw:	让 <code>mount</code> 的扇区为可擦写
ro:	设定为只读属性
remount :	让系统本来挂载的扇区重新被挂载



范例:

```
[root@test /root]# mount -a
```

```
[root@test /root]# mount -t iso9660 /dev/cdrom  
/mnt/cdrom
```

挂载光盘

```
[root@test /root]# mount -t vfat /dev/fd0 /mnt/floppy
```

挂载 windows 文件系统的软盘

```
[root@test /root]# mount -t ext2 /dev/fd0 /mnt/floppy
```

挂载 Linux 文件系统的软盘

```
[root@test /root]# mount -t ext2 /dev/hdc6 /home
```

挂载 Linux 文件格式硬盘



卸载命令umount

```
# umount /mountpoint
```

卸载/mountpoint目录上所挂载的文件系统。

例如：

```
# mount /dev/sda1 /mnt/usb(挂载u盘)
```

```
#umount /mnt/usb（卸载u盘）
```



fdisk

查看磁盘分区,或删除修改分区等

`fdisk /dev/had`

查看磁盘等容量:

quota:系统管理员查看某个用户的磁盘使用情况

格式: `quota -u` 用户名

free

查看内存使用情况



df 查看已挂载磁盘的总容量、使用容量等信息

- k: 使用 **KBytes** 显示结果
- m: 使用 **MBytes** 显示结果
- h: 输出结果带上单位

du 查看目录占用的磁盘容量信息.

如果不指定文件或目录名, 则默认对当前目录进行统计。

- a :全部的文件与目录都列出来,默认值是指列出目录的值
- c :最后加总 **total**
- b :列出的值以 **bytes** 输出
- k :列出的值以 **KB** 输出
- m :列出的值以 **MB** 输出
- s :只列出最后加总的值(块数, 1块=1024字节)

2010-1-11

- h:输出结果带上单位



file: 查看文件类型

格式: **file** 文件名

stat:显示文件相关信息

格式: **stat** 文件名

```
[root@localhost ~]# file install.log
install.log: UTF-8 Unicode text
[root@localhost ~]# stat install.log
  File: 'install.log'
  Size: 62406          Blocks: 144          IO Block: 4096   普通文件
Device: fd00h/64768d  Inode: 491522       Links: 1
Access: (0644/-rw-r--r--)  Uid: (  0/   root)   Gid: (  0/   root)
Access: 2008-07-23 10:50:49.000000000 +0800
Modify: 2008-07-20 07:08:52.000000000 +0800
Change: 2008-07-20 07:11:54.000000000 +0800
```



date: 查看/修改系统时间

uname: 显示系统信息

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 2.6.23.1-42.fc8 #1 SMP Tue Oct 30 13:55:12 EDT 2007
i686 athlon i386 GNU/Linux
```

hostname: 显示当前本地主机名称

alias/unalias: 设定/取消别名

格式:

查看: **alias**

设定: **alias ls='ls -l'**



&:指定后台运行

格式: 命令 **&**

jobs:显示目前正在后台运行的任务列表。

fg:将后台运行的程序移至前台。

格式: **fg** 序号 (通过**jobs**命令显示的)

clear: 清屏并把光标移至左上角。

who:查看系统当前有哪些用户登录

```
[root@localhost wjl]# who
root      tty2          2008-07-23 08:01
root      tty7          2008-07-23 08:00 (:0)
root      pts/1        2008-07-23 16:26 (:0.0)
root      pts/2        2008-07-23 15:17 (:0.0)
```



ps:检测进程情况

ps -aux:查看所有进程

top: 动态实时显示系统进程情况

kill:杀死指定进程

格式: **kill pid**

说明:

- 1、pid可通过ps查询。
- 2、对无法自动结束的进程，可使用参数“-9”来强行结束（可能会造成数据丢失，如vi）。



/proc文件系统

proc不是真正的文件系统，不占用外存空间，只是以文件的方式为访问Linux内核数据提供接口，用户和应用程序可通过**/proc**获得系统的运行信息。系统信息是动态变化的，所以用户读取**/pro**文件时只是瞬时的。

1、查看进程信息

/proc目录中，每个一数字命名的子目录对应着系统中的一个进程，数字就是进程号**pid**。

例：查看**vi**进程（通过**ps**查看到**pid=4929**）

```
4929 pts/1    S+      0:00 vim abc.c
4953 pts/2    R+      0:00 ps -ax
[root@localhost proc]# cd 4929
[root@localhost 4929]# ls
attr          coredump_filter  exe            loginuid      mountstats    sched          statm
auxv          cpuset           fd             maps          oom_adj       schedstat      status
clear_refs   cwd              fdinfo         mem           oom_score     snaps          task
cmdline      environ          io             mounts        root          stat           wchan
[root@localhost 4929]# ll exe
lrwxrwxrwx 1 root root 0 07-23 17:02 exe -> /usr/bin/vim
[root@localhost 4929]# ll cwd
lrwxrwxrwx 1 root root 0 07-23 17:02 cwd -> /home/wjl
[root@localhost 4929]#
```




2、查看cpu信息

cpuinfo:cpu信息

meminfo:内存信息

interrupts:系统中断号

uptime:系统运行时间

filesystems:系统支持的文件系统类型

3、查看系统模块信息

modules

4、查看系统版本信息

version

5、查看系统分区信息



关机命令

halt

Shutdown -h now

shutdown -h 12: 00

文本模式屏幕: Ctrl+alt+del

图形界面屏幕: (1)

ctrl+alt+backspace

(2) Ctrl+alt+del

注意: 不要直接关闭电源, 以免丢失数据或破坏文件系统



重新启动系统命令

`shutdown -r`

`shutdown -r 10` （10分钟之后重新启动）

`reboot`