

附录 2 Linux 常用命令

在文本模式和终端模式下，经常使用 Linux 命令来查看系统的状态和监视系统的操作，如对文件和目录进行浏览、操作等。在 Linux 较早的版本中，由于不支持图形化操作，用户基本上都是使用命令行方式对系统进行操作，所以掌握常用的 Linux 命令是必要的，本章将对 Linux 的常用命令进行分类介绍。

本章学习要点：

- ✧ Linux 系统的终端窗口和命令基础。
- ✧ 文件目录类命令。
- ✧ 系统信息类命令。
- ✧ 进程管理类命令及其他常用命令。

2.1 Linux 命令基础

掌握 Linux 命令对于管理 Linux 网络操作系统是非常必要的。

2.1.1 Linux 命令特点

在 Linux 系统中命令区分大小写。在命令行中，可以使用 Tab 键来自动补齐命令，即可以只输入命令的前几个字母，然后按 Tab 键，系统将自动补齐该命令，若命令不止一个，则显示出所有和输入字符相匹配的命令。

按 Tab 键时，如果系统只找到一个和输入字符相匹配的目录或文件，则自动补齐；如果没有匹配的内容或有多个相匹配的名字，系统将发出警鸣声，再按一下 Tab 键将列出所有相匹配的内容（如果有的话），以供用户选择。例如，在命令提示符后输入“mou”，然后按 Tab 键，系统将自动补全该命令为“mount”；如果在命令提示符后只输入“mo”，然后按 Tab 键，此时将警鸣一声，再次按 Tab 键，系统将显示所有以“mo”开头的命令。

另外，利用向上或向下的光标键，可以翻查曾经执行过的历史命令，并可以再次执行。

如果要在一个命令行上输入和执行多条命令，可以使用分号来分隔命令。例如：“cd /;ls”。

断开一个长命令行，可以使用反斜杠“\”，以将一个较长的命令分成多行表达，增强命令的可读性。执行后，Shell 自动显示提示符“>”，表示正在输入一个长命令，此时可继续在新行上输入命令的后续部分。

2.1.2 后台运行程序

一个文本控制台或一个仿真终端在同一时刻只能运行一个程序或命令，在未执行结束前，一般不能进行其他操作，此时可采用将程序在后台执行，以释放控制台或终端，使其仍能进行其他操作。要使程序以后台方式执行，只需在要执行的命令后跟上一个“&”符号即可，例如“find / -name httpd.conf &”。

2.2 文件目录类命令

文件目录类命令是对文件和目录进行各种操作的命令。

2.2.1 浏览目录类命令

1. pwd 命令

pwd 命令用于显示用户当前所在的目录。如果用户不知道自己当前所处的目录，就必须使用它。例如：

```
[root@RHEL4 etc]# pwd
```

```
/etc
```

2. cd 命令

cd 命令用来在不同的目录中进行切换。用户在登录系统后，会处于用户的家目录（\$HOME）中，该目录一般以/home 开始，后跟用户名，这个目录就是用户的初始登录目录（root 用户的家目录为/root）。如果用户想切换到其他的目录中，就可以使用 cd 命令，后跟想要切换的目录名。例如：

```
[root@RHEL4 etc]# cd          //改变目录位置至用户登录时的工作目录。
```

```
[root@RHEL4 etc]# cd dir1     //改变目录位置至当前目录下的 dir1 子目录下。
```

```
[root@RHEL4 etc]# cd ~        //改变目录位置至用户登录时的工作目录（用户的家目录）。
```

```
[root@RHEL4 etc]# cd ..       //改变目录位置至当前目录的父目录。
```

```
[root@RHEL4 etc]# cd ../user   //改变目录位置至当前目录的父目录下的 user 子目录下。
```

```
[root@RHEL4 etc]# cd /dir1/subdir1 //利用绝对路径表示改变目录到 /dir1/subdir1 目录下。
```

说明：在 Linux 系统中，用 “.” 代表当前目录；用 “..” 代表当前目录的父目录；用 “~” 代表用户的个人家目录（主目录）。例如，root 用户的个人主目录是/root，则不带任何参数的 “cd” 命令相当于 “cd ~”，即将目录切换到用户的家目录。

3. ls 命令

ls 命令用来列出文件或目录信息。该命令的语法为：

```
ls [参数] [目录或文件]
```

ls 命令的常用参数选项有：

- -a: 显示所有文件，包括以“.”开头的隐藏文件。
- -A: 显示指定目录下所有的子目录及文件，包括隐藏文件。但不显示“.”和“..”。
- -c: 按文件的修改时间排序。
- -C: 分成多列显示各行。
- -d: 如果参数是目录，只显示其名称而不显示其下的各个文件。往往与“-l”选项一起使用，以得到目录的详细信息。
- -l: 以长格形式显示文件的详细信息。
- -li: 在输出的第一列显示文件的 i 节点号。

例如：

```
[root@RHEL4 ~]#ls //列出当前目录下的文件及目录。
```

```
[root@RHEL4 ~]#ls -a //列出包括以“.”开始的隐藏文件在内的所有文件。
```

```
[root@RHEL4 ~]#ls -t //依照文件最后修改时间的顺序列出文件。
```

```
[root@RHEL4 ~]#ls -F //列出当前目录下的文件名及其类型。以/ 结尾表示为目录名，以* 结尾表示为可执行文件，以@ 结尾表示为符号连接。
```

```
[root@RHEL4 ~]#ls -l //列出当前目录下所有文件的权限、所有者、文件大小、修改时间及名称。
```

```
[root@RHEL4 ~]#ls -lg //同上，并显示出文件的所有者工作组名。
```

```
[root@RHEL4 ~]#ls -R //显示出目录下以及其所有子目录的文件名。
```

2.2.2 浏览文件类命令

1. cat 命令

cat 命令主要用于滚屏显示文件内容或是将多个文件合并成一个文件。该命令的语法为：

```
cat [参数] 文件名
```

cat 命令的常用参数选项有：

- -b: 对输出内容中的非空行标注行号。
- -n: 对输出内容中的所有行标注行号。

通常使用 cat 命令查看文件内容，但是 cat 命令的输出内容不能够分页显示，要查看超过一屏的文件内容，需要使用 more 或 less 等其他命令。如果在 cat 命令中没有指定参数，则 cat 会从标准输入（键盘）获取内容。

例如要查看 /soft/file1 文件的内容的命令为：

```
[root@RHEL4 ~]#cat /soft/file1
```

利用 cat 命令还可以合并多个文件。例如要把 file1 和 file2 文件的内容合并为 file3，且 file2 文件的内容在 file1 文件的内容前面，则命令为：

```
[root@RHEL4 ~]# cat file2 file1>file3
```

//如果 file3 文件存在，此命令的执行结果会覆盖 file3 文件中原有内容

```
[root@RHEL4 ~]# cat file2 file1>>file3
```

//如果 file3 文件存在，此命令的执行结果将把 file2 和 file1 文件的内容附加到 file3 文件中原有内容的后面。

说明：关于 “>” 和 “>>” 输出重定向的区别，详见第 4 章。

2. more 命令

在使用 cat 命令时，如果文件太长，用户只能看到文件的最后一部分。这时可以使用 more 命令，一页一页的分屏显示文件的内容。more 命令通常用于分屏显示文件内容。大部分情况下，可以不加任何参数选项执行 more 命令查看文件内容，执行 more 命令后，进入 more 状态，按【Enter】键可以向下移动一行，按【space】键可以向下移动一页；按【q】键可以退出 more 命令。该命令的语法为：

```
more [参数] 文件名
```

more 命令的常用参数选项有：

- -num: 这里的 num 是一个数字，用来指定分页显示时每页的行数。
- +num: 指定从文件的第 num 行开始显示。

例如：

```
[root@RHEL4 ~]#more file1      // 以分页方式查看 file1 文件的内容。
```

```
[root@RHEL4 ~]#cat file1 | more  //以分页方式查看 file1 文件的内容。
```

`more` 命令经常在管道中被调用用以实现各种命令输出内容的分屏显示。上面的第二个命令就是利用 `shell` 的管道功能分屏显示 `file1` 文件的内容。关于管道的内容在第 4 章中有详细的介绍。

3. less 命令

`less` 命令是 `more` 命令的改进版，比 `more` 命令的功能强大。`more` 命令只能向下翻页，而 `less` 命令可以向下、向上翻页，甚至可以前后左右的移动。执行 `less` 命令后，进入了 `less` 状态，按【Enter】键可以向下移动一行，按【space】键可以向下移动一页；，按【b】键可以向上移动一页；也可以用光标键向前、后、左、右移动；按【q】键可以退出 `less` 命令。

`less` 命令还支持在一个文本文件中进行快速查找。先按下斜杠键【/】，再输入要查找的单词或字符。`less` 命令会在文本文件中进行快速查找，并把找到的第一个搜索目标高亮度显示。如果希望继续查找，就再次按下斜杠键【/】，再按【Enter】键即可。

`less` 命令的用法与 `more` 基本相同，例如：

```
[root@RHEL4 ~]#less /etc/httpd/conf/httpd.conf  // 以分页方式查看 httpd.conf 文件的内容。
```

4. head 命令

`head` 命令用于显示文件的开头部分，默认情况下只显示文件的前 10 行内容。该命令的语法为：

```
head [参数] 文件名
```

`head` 命令的常用参数选项有：

- `-n num`：显示指定文件的前 `num` 行。
- `-c num`：显示指定文件的前 `num` 个字符。

例如：

```
[root@RHEL4 ~]#head -n 20 /etc/httpd/conf/httpd.conf //显示 httpd.conf 文件的前 20 行。
```

5. tail 命令

tail 命令用于显示文件的末尾部分，默认情况下只显示文件的末尾 10 行内容。该命令的语法为：

```
tail [参数] 文件名
```

tail 命令的常用参数选项有：

- -n num：显示指定文件的末尾 num 行。
- -c num：显示指定文件的末尾 num 个字符。
- +num：从第 num 行开始显示指定文件的内容。

例如：

```
[root@RHEL4 ~]#tail -n 20 /etc/httpd/conf/httpd.conf //显示 httpd.conf 文件的末尾 20 行。
```

2.2.3 目录操作类命令

1. mkdir 命令

mkdir 命令用于创建一个目录。该命令的语法为：

```
mkdir [参数] 目录名
```

上述目录名可以为相对路径，也可以为绝对路径。

mkdir 命令的常用参数选项有：

- -p：在创建目录时，如果父目录不存在，则同时创建该目录及该目录的父目录。

例如：

```
[root@RHEL4 ~]#mkdir dir1 //在当前目录下创建 dir1 子目录
```

```
[root@RHEL4 ~]#mkdir -p dir2/subdir2
```

//在当前目录的 dir2 目录中创建 subdir2 子目录，如果 dir2 目录不存在则同时创建

2. rmdir 命令

rmdir 命令用于删除空目录。该命令的语法为：

```
rmdir [参数] 目录名
```

上述目录名可以为相对路径，也可以为绝对路径。但所删除的目录必须为空目录。

rmdir 命令的常用参数选项有：

- -p: 在删除目录时，一起删除父目录，但父目录中必须没有其他目录及文件。
- 例如：

```
[root@RHEL4 ~]#rmdir dir1 //在当前目录下删除 dir1 空子目录
```

```
[root@RHEL4 ~]#rmdir -p dir2/subdir2
```

//删除当前目录中 dir2/subdir2 子目录，删除 subdir2 目录时，如果 dir2 目录无其他目录，则一起删除。

2.2.4 文件操作类命令

1. cp 命令

cp 命令主要用于文件或目录的复制。该命令的语法为：

```
cp [参数] 源文件 目标文件
```

cp 命令的常用参数选项有：

- -f: 如果目标文件或目录存在，先删除它们再进行复制（即覆盖），并且不提示用户。
- -i: 如果目标文件或目录存在，提示是否覆盖已有的文件。
- -R: 递归复制目录，即包含目录下的各级子目录。

例如：

//将/etc/inittab 文件复制到用户的家目录下，复制后的文件名为 inittab.bak。

```
[root@RHEL4 ~]#cp /etc/inittab ~/inittab.bak
```

//将/etc/init.d 目录（包含 rc.d 目录的文件及子目录）复制到/initbak 目录下。

```
[root@RHEL4 ~]#cp -R /etc/init.d/ /initbak
```

2. mv 命令

mv 命令主要用于文件或目录的移动或改名。该命令的语法为：

```
mv [参数] 源文件或目录 目标文件或目录
```

mv 命令的常用参数选项有：

- -i: 如果目标文件或目录存在时，提示是否覆盖目标文件或目录。
- -f: 无论目标文件或目录是否存在，直接覆盖目标文件或目录，不提示。

例如：

//将当前目录下的 testa 文件移动到/usr/目录下，文件名不变。

```
[root@RHEL4 /]# mv testa /usr/
```

//将/usr/testa 文件移动到根目录下，移动后的文件名为 tt。

```
[root@RHEL4 /]# mv /usr/testa /tt
```

3. rm 命令

rm 命令主要用于文件或目录的删除。该命令的语法为：

```
rm [参数] 文件名或目录名
```

rm 命令的常用参数选项有：

- -i: 删除文件或目录时提示用户。
- -f: 删除文件或目录时不提示用户。
- -R: 递归删除目录，即包含目录下的文件和各级子目录。

例如：

```
//删除当前目录下的所有文件，但不删除子目录和隐藏文件。
```

```
[root@RHEL4 test]# rm *
```

```
// 删除当前目录下的子目录 dir，包含其下的所有文件和子目录，并且提示用户确认。
```

```
[root@RHEL4 /]# rm -iR dir
```

4. touch 命令

touch 命令用于建立文件或更新文件的修改日期。该命令的语法为：

```
touch [参数] 文件名或目录名
```

touch 命令的常用参数选项有：

- -d yyyymmdd：把文件的存取或修改时间改为 yyyy 年 mm 月 dd 日。
- -a：只把文件的存取时间改为当前时间。
- -m：只把文件的修改时间改为当前时间。

例如：

```
[root@RHEL4 test]# touch aa //如果当前目录下存在 aa 文件，则把 aa 文件的存取和修改时间改为当前时间，如果不存在 aa 文件，则新建 aa 文件。
```

```
[root@RHEL4 /]# touch -d 20080808 aa //将 aa 文件的存取和修改时间改为 2008 年 8 月 8 日。
```

5. diff 命令

diff 命令用于比较两个文件内容的不同。该命令的语法为：

```
diff [参数] 源文件 目标文件
```

diff 命令的常用参数选项有：

- -a：将所有的文件当作文本文件处理。
- -b：忽略空格造成的不同。
- -B：忽略空行造成的不同。

- -q: 只报告什么地方不同，不报告具体的不同信息。
- -i: 忽略大小写的变化。

例如：

```
[root@RHEL4 test]# diff aa.txt bb.txt //比较 aa.txt 文件和 bb.txt 文件的不同。
```

6. ln 命令

ln 命令用于建立两个文件之间的链接关系。该命令的语法为：

```
ln [参数] 源文件或目录 链接名
```

ln 命令的常用参数选项有：

- -s: 建立符号链接（软链接），不加该参数时建立的链接为硬链接。

两个文件之间的链接关系有两种：一种称为硬链接，这时两个文件名指向的是硬盘上的同一块存储空间，对两个文件中的任何一个文件的内容进行修改都会影响到另一文件，这种链接关系称为硬链接。它可由 ln 命令不加任何参数建立。

利用 ll 命令查看/aa.txt 文件情况：

```
[root@RHEL4 /]# ll aa
```

```
-rw-r--r-- 1 root root 0 1 月 31 15:06 aa
```

```
[root@RHEL4 /]# cat aa
```

```
this is aa
```

由上面命令的执行结果可以看出 aa 文件的链接数为 1，文件内容为“this is aa”。

使用 ln 命令建立 aa 文件的硬链接 bb：

```
[root@RHEL4 /]# ln aa bb
```

上述命令产生了 bb 新文件，它和 aa 文件建立起了硬链接关系。

```
[root@RHEL4 /]# ll aa bb
```

```
-rw-r--r-- 2 root root 11 1月 31 15:44 aa
```

```
-rw-r--r-- 2 root root 11 1月 31 15:44 bb
```

```
[root@RHEL4 /]# cat bb
```

```
this is aa
```

可以看出，aa 和 bb 的大小相同，内容相同。再看详细信息第 2 列，原来 aa 文件的链接数为 1，说明这块硬盘空间只有 aa 文件指向，而建立起 aa 和 bb 的硬链接关系后，这块硬盘空间就有 aa 和 bb 两个文件同时指向它，所以 aa 和 bb 的链接数都变为 2。

此时，如果修改 aa 或 bb 任意一个文件的内容，另外一个文件的内容也将随之变化。如果删除其中一个文件（不管是哪一个），就是删除了该文件和硬盘空间的指向关系，该硬盘空间不会释放，另外一个文件的内容也不会发生改变，但是该文件的链接数会减少一个。

说明：只能对文件建立硬链接，不能对目录建立硬链接。

另外一种链接方式称为符号链接（软链接），是指一个文件指向另外一个文件的文件名。软链接类似于 Windows 系统中的快捷方式。软链接由 `ln -s` 命令建立。

首先查看一下 aa 文件的信息：

```
[root@RHEL4 /]# ll aa
```

```
-rw-r--r-- 1 root root 11 1月 31 15:44 aa
```

创建 aa 文件的符号链接 cc，创建完成后查看 aa 和 cc 文件的链接数的变化：

```
[root@RHEL4 /]# ln -s aa cc
```

```
[root@RHEL4 /]# ll aa cc
```

```
-rw-r--r-- 1 root root 11 1月 31 15:44 aa
```

```
lrwxrwxrwx 1 root root 2 1月 31 16:02 cc -> aa
```

可以看出 cc 文件是指向 aa 文件的一个符号链接。而指向存储 aa 文件内容的那块硬盘

空间的文件仍然只有 aa 一个文件，cc 文件只不过是指向了 aa 文件名而已。所以 aa 文件的链接数仍为 1。

在利用 cat 命令查看 cc 文件的内容时，cat 命令在寻找 cc 的内容时，发现 cc 是一个符号链接文件，就根据 cc 记录的文件名找到 aa 文件，然后将 aa 文件的内容显示出来。

此时如果删除了 cc 文件，对 aa 文件无任何影响，但如果删除了 aa 文件，那么 cc 文件就因无法找到 aa 文件而毫无用处了。

说明：可以对文件或目录建立软链接。

7. gzip 和 gunzip 命令

gzip 命令用于对文件进行压缩，生成的压缩文件以 “.gz” 结尾，而 gunzip 命令是对以 “.gz” 结尾的文件进行解压缩。该命令的语法为：

```
gzip -v 文件名
```

```
gunzip -v 文件名
```

-v 参数选项表示显示被压缩文件的压缩比或解压时的信息。

例如：

```
[root@RHEL4 /]# gzip -v httpd.conf
```

```
httpd.conf:      65.0% -- replaced with httpd.conf.gz
```

```
[root@RHEL4 /]# gunzip -v httpd.conf.gz
```

```
httpd.conf.gz:   65.0% -- replaced with httpd.conf
```

8. tar 命令

tar 是用于文件打包的命令行工具，tar 命令可以把一系列的文件归档到一个大文件中，也可以把档案文件解开以恢复数据。总的来说，tar 命令主要用于打包和解包。tar 命令是 Linux 系统中常用的备份工具之一。该命令的语法为：

```
tar [参数] 档案文件 文件列表
```

tar 命令的常用参数选项有：

- -c：生成档案文件。
- -v：列出归档解档的详细过程。
- -f：指定档案文件名称。
- -r：将文件追加到档案文件末尾。

- -z: 以 gzip 格式压缩或解压缩文件。
- -j: 以 bzip2 格式压缩或解压缩文件。
- -d: 比较档案与当前目录中的文件。
- -x: 解开档案文件。

例如:

```
[root@RHEL4 /]# tar -cvf yy.tar aa tt //将当前目录下的 aa 和 tt 文件归档为 yy.tar
```

```
[root@RHEL4 /]# tar -xvf yy.tar //从 yy.tar 档案文件中恢复数据
```

```
[root@RHEL4 /]# tar -czvf yy.tar.gz aa tt //将当前目录下的 aa 和 tt 文件归档并压缩为 yy.tar.gz
```

```
[root@RHEL4 /]# tar -xzvf yy.tar.gz //将 yy.tar.gz 文件解压缩并恢复数据
```

9. rpm 命令

rpm 命令主要用于对 RPM 软件包进行管理。RPM 包是 Linux 的各种发行版本中应用最为广泛的软件包格式之一。学会使用 rpm 命令对 RPM 软件包进行管理至关重要。该命令的语法为:

```
rpm [参数] 软件包名
```

rpm 命令的常用参数选项有:

- -qa: 查询系统中安装的所有软件包。
- -q: 查询指定的软件包在系统中是否安装。
- -qi: 查询系统中已安装软件包的描述信息。
- -ql: 查询系统中已安装软件包里所包含的文件列表。
- -qf: 查询系统中指定文件所属的软件包。
- -qp: 查询 RPM 包文件中的信息,通常用于在未安装软件包之前了解软件包中的信息。
- -i: 用于安装指定的 RPM 软件包。
- -v: 显示较详细的信息。
- -h: 以“#”显示进度。
- -e: 删除已安装的 RPM 软件包。
- -U: 升级指定的 RPM 软件包。软件包的版本必须比当前系统中安装的软件包的版本高才能正确升级。如果当前系统中并未安装指定的软件包,则直接安装。
- -F: 更新软件包。

例如:

```
[root@RHEL4 /]#rpm -qa|more //显示系统安装的所有软件包列表
```

```
[root@RHEL4 /]#rpm -q httpd //查询系统是否安装了 httpd 软件包
```

```
[root@RHEL4 /]#rpm -qi httpd //查询系统已安装的 httpd 软件包的描述信息
```

```
[root@RHEL4 /]#rpm -ql httpd //查询系统已安装的 httpd 软件包里所包含的文件列表
```

```
[root@RHEL4 /]#rpm -qf /etc/passwd //查询 passwd 文件所属的软件包
```

```
[root@RHEL4 RPMS]# rpm -ivh httpd-2.0.52-9.ent.i386.rpm //安装软件包，并以“#”显示  
安装进度和安装的详细信息
```

```
[root@RHEL4 RPMS]# rpm -Uvh httpd-2.0.52-9.ent.i386.rpm //升级软件包
```

```
[root@RHEL4 RPMS]#rpm -e httpd //卸载 httpd 软件包
```

10. whereis 命令

whereis 命令用来寻找命令的可执行文件所在的位置。该命令的语法为：

```
whereis [参数] 命令名称
```

whereis 命令的常用参数选项有：

- -b: 只查找二进制文件。
- -m: 只查找命令的联机帮助手册部分。
- -s: 只查找源代码文件。

例如：

```
//查找命令 rpm 的位置
```

```
[root@RHEL4 ~]# whereis rpm
```

```
rpm: /bin/rpm /etc/rpm /usr/lib/rpm /usr/include/rpm /usr/share/man/man8/rpm.8.gz
```

11. whatis 命令

whatis 命令用于获取命令简介。它从某个程序的使用手册中抽出一行简单的介绍性文件，帮助用户迅速了解这个程序的具体功能。该命令的语法为：

```
whatis 命令名称
```

例如：

```
[root@RHEL4 ~]# whatis ls
```

```
ls (1) - list directory contents
```

12. find 命令

find 命令用于文件查找。它的功能非常强大。该命令的语法为：

```
find [路径] [匹配表达式]
```

find 命令的匹配表达式主要有如下几种类型：

- **-name filename**：查找指定名称的文件。
- **-user username**：查找属于指定用户的文件。
- **-group grpname**：查找属于指定组的文件。
- **-print**：显示查找结果。
- **-size n**：查找大小为 **n** 块的文件，一块为 512B。符号 “+n” 表示查找大小大于 **n** 块的文件；符号 “-n” 表示查找大小小于 **n** 块的文件；符号 “nc” 表示查找大小为 **n** 个字符的文件。
- **-inum n**：查找索引节点号为 **n** 的文件。
- **-type**：查找指定类型的文件。文件类型有：**b**（块设备文件）、**c**（字符设备文件）、**d**（目录）、**p**（管道文件）、**l**（符号链接文件）、**f**（普通文件）。
- **-atime n**：查找 **n** 天前被访问过的文件。“+n” 表示超过 **n** 天前被访问的文件；“-n” 表示未超过 **n** 天前被访问的文件；
- **-mtime n**：类似于 **atime**，但检查的是文件内容被修改的时间。

- `-ctime n`: 类似于 `atime`, 但检查的是文件索引节点被改变的时间。
- `-perm mode`: 查找与给定权限匹配的文件, 必须以八进制的形式给出访问权限。
- `-newer file`: 查找比指定文件新的文件, 即最后修改时间离现在较近。
- `-exec command {} \;`: 对匹配指定条件的文件执行 `command` 命令。
- `-ok command {} \;`: 与 `exec` 相同, 但执行 `command` 命令时请求用户确认。

例如:

```
[root@RHEL4 ~]# find . -type f -exec ls -l {} \;
```

//在当前目录下查找普通文件, 并以长格形式显示

```
[root@RHEL4 ~]# find /logs -type f -mtime 5 -exec rm {} \;
```

//在/logs 目录中查找修改时间为 5 天以前的普通文件, 并删除。

```
[root@RHEL4 ~]# find /etc -name "*.conf"
```

//在/etc/目录下查找文件名以 “.conf” 结尾的文件

```
[root@RHEL4 ~]# find . -type f -perm 755 -exec ls {} \;
```

//在当前目录下查找权限为 755 的普通文件, 并显示。

注意: 由于 `find` 命令在执行过程中将消耗大量资源, 建议以后台方式运行。

13. `grep` 命令

`grep` 命令用于查找文件中包含有指定字符串的行。该命令的语法为:

```
grep [参数] 要查找的字符串 文件名
```

`grep` 命令的常用参数选项有:

- `-v`: 列出不匹配的行。
- `-c`: 对匹配的行计数。
- `-l`: 只显示包含匹配模式的文件名。
- `-h`: 抑制包含匹配模式的文件名的显示。
- `-n`: 每个匹配行只按照相对的行号显示。
- `-i`: 对匹配模式不区分大小写。

在 `grep` 命令中, 字符 “^” 表示行的开始, 字符 “\$” 表示行的结尾。如果要查找的字符串中带有空格, 可以用单引号或双引号括起来。

例如：

```
[root@RHEL4 ~]# grep -2 user1 /etc/passwd
```

//在文件 passwd 中查找包含字符串 “user1” 的行，如果找到，显示该行及该行前后各 2 行的内容。

```
[root@RHEL4 ~]# grep "^user1$" /etc/passwd
```

//在 passwd 文件中搜索只包含 “user1” 五个字符的行。

提示：grep 和 find 命令的差别在于 grep 是在文件中搜索满足条件的行，而 find 是在指定目录下根据文件的相关信息查找满足指定条件的文件。

2.3 系统信息类命令

系统信息类命令是对系统的各种信息进行显示和设置的命令。

1. dmesg 命令

dmesg 命令用实例名和物理名称来标识连到系统上的设备。dmesg 命令也显示系统诊断信息、操作系统版本号、物理内存大小以及其他信息。例如：

```
[root@RHEL4 ~]#dmesg|more
```

提示：系统启动时，屏幕上会显示系统 CPU、内存、网卡等硬件信息。但通常显示的比较快，如果用户没有来得及看清，可以在系统启动后用 dmesg 命令查看。

2. df 命令

df 命令主要用来查看文件系统的各个分区的占用情况。例如：

```
[root@RHEL4 ~]# df
```

Filesystem	1K-块	已用	可用	已用%	挂载点
/dev/sda3	5842664	2778608	2767256	51%	/
/dev/sda1	93307	8564	79926	10%	/boot
none	63104	0	63104	0%	/dev/shm
/dev/hdc	641798	641798	0	100%	/media/cdrom

该命令列出了系统上所有已挂载的分区的大小、已占用的空间、可用空间以及占用率。空间大小的单位为 K。使用选项-h，将使输出的结果具有更好的可读性，例如：

```
[root@RHEL4 ~]# df -h
```

Filesystem	容量	已用	可用	已用%	挂载点
/dev/sda3	5.6G	3.7G	3.7G	51%	/
/dev/sda1	92M	8.4M	79M	10%	/boot
none	62M	0	62M	0%	/dev/shm
/dev/hdc	627M	627M	0	100%	/media/cdrom

3. du 命令

du 命令主要用来查看某个目录中的各级子目录所使用的硬盘空间数。基本用法是在命令后跟目录名，如果不跟目录名，则缺省为当前目录。例如：

```
[root@RHEL4 dir1]# du /dir1
```

```
/dir1/test/subdir2
```

```
4      /dir1/test/subdir1
```

```
20     /dir1/test
```

```
24     /dir1
```

该命令显示出当前目录下各级子目录所占用的硬盘空间数。

有些情况下，用户可能只想查看某个目录总的已使用空间，则可以使用-s 选项。

4. free 命令

free 命令主要用来查看系统内存，虚拟内存的大小及占用情况。例如：

```
[root@RHEL4 dir1]# free
```

	total	used	free	shared	buffers
cached					
Mem:	126212	124960	1252	0	16408
34028					
-/+ buffers/cache:	74524	51688			
Swap:	257032	25796	231236		

5. date 命令

date 命令可以用来查看系统当前的日期和时间。例如：

```
[root@RHEL4 dir1]# date
```

```
四 1月 31 18:55:16 CST 2008
```

date 命令还可以用来设置当前日期和时间，例如：

```
[root@RHEL4 dir1]# date -d 08/08/2008
```

```
五 8月 8 00:00:00 CST 2008
```

注意：只有 root 用户才可以改变系统的日期和时间。

6. cal 命令

cal 命令用于显示指定月份或年份的日历，可以带两个参数，其中年、月份用数字表示；只有一个参数时表示年份，年份的范围为 1~9999；不带任何参数的 cal 命令显示当前月份的日历。例如：

```
[root@RHEL4 dir1]# cal 8 2008
```

八月 2008

日	一	二	三	四	五	六
---	---	---	---	---	---	---

					1	2
--	--	--	--	--	---	---

3	4	5	6	7	8	9
---	---	---	---	---	---	---

10	11	12	13	14	15	16
----	----	----	----	----	----	----

17	18	19	20	21	22	23
----	----	----	----	----	----	----

24	25	26	27	28	29	30
----	----	----	----	----	----	----

31						
----	--	--	--	--	--	--

7. clock 命令

clock 命令用于从计算机的硬件获得日期和时间。例如：

```
[root@RHEL4 dir1]# clock
```

2008 年 01 月 31 日 星期四 18 时 51 分 34 秒 -0.272325 seconds

2.4 进程管理类命令

进程管理类命令是对进程进行各种显示和设置的命令。

1. ps 命令

ps 命令主要用于查看系统的进程。该命令的语法为：

ps [参数]

ps 命令的常用参数选项有：

- -a: 显示当前控制终端的进程（包含其他用户的）。
- -u: 显示进程的用户名和启动时间等信息。
- -w: 宽行输出，不截取输出中的命令行。
- -l: 按长格形式显示输出。
- -x: 显示没有控制终端的进程。
- -e: 显示所有的进程。
- -t n: 显示第 n 个终端的进程。

例如：

```
[root@RHEL4 dir1]# ps -au
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	2459	0.0	0.2	1956	348	tty2	Ss+	09:00	0:00	/sbin/mingetty tty2
root	2460	0.0	0.2	2260	348	tty3	Ss+	09:00	0:00	/sbin/mingetty tty3
root	2461	0.0	0.2	3420	348	tty4	Ss+	09:00	0:00	/sbin/mingetty tty4
root	2462	0.0	0.2	3428	348	tty5	Ss+	09:00	0:00	/sbin/mingetty tty5
root	2463	0.0	0.2	2028	348	tty6	Ss+	09:00	0:00	/sbin/mingetty tty6
root	2895	0.0	0.9	6472	1180	tty1	Ss	09:09	0:00	bash

提示：ps 通常和重定向、管道等命令一起使用，用于查找出所需的进程。

2. kill 命令

前台进程在运行时，可以用【ctrl+c】键来终止它，但后台进程无法使用这种方法终止，此时可以使用 kill 命令向进程发送强制终止信号，以达到目的。例如：

```
[root@RHEL4 dir1]# kill -l
```

```
1)  SIGHUP      2)  SIGINT      3)  SIGQUIT     4)  SIGILL

5)  SIGTRAP     6)  SIGABRT    7)  SIGBUS      8)  SIGFPE

9)  SIGKILL     10) SIGUSR1    11) SIGSEGV     12) SIGUSR2

13) SIGPIPE     14) SIGALRM    15) SIGTERM     17) SIGCHLD

18) SIGCONT     19) SIGSTOP    20) SIGTSTP     21) SIGTTIN

22) SIGTTOU     23) SIGURG     24) SIGXCPU     25) SIGXFSZ

26) SIGVTALRM   27) SIGPROF    28) SIGWINCH    29) SIGIO

30) SIGPWR      31) SIGSYS     34) SIGRTMIN    35) SIGRTMIN+1

(略)
```

上述命令用于显示 kill 命令所能够发送的信号种类。每个信号都有一个数值对应，例如“SIGKILL”信号的值为 9。

kill 命令的格式为：

```
kill [参数] 进程 1 进程 2 .....
```

参数选项-s 一般跟信号的类型。

例如：

```
[root@RHEL4 dir1]# ps
```

PID	TTY	TIME	CMD
1448	pts/1	00:00:00	bash
2394	pts/1	00:00:00	ps

```
[root@RHEL4 dir1]# kill -s SIGKILL 2394 或者//kill -9 2394
```

//上述命令用于结束 ps 进程

3. killall 命令

和 kill 命令相似，killall 命令可以根据进程名发送信号。例如：

```
[root@RHEL4 dir1]# killall -9 httpd
```

4. nice 命令

Linux 系统有两个和进程有关的优先级。用“ps -l”命令可以看到两个域：PRI 和 NI。PRI 是进程实际的优先级，它是由操作系统动态计算的，这个优先级的计算和 NI 值有关。NI 值可以被用户更改，NI 值越高，优先级越低。一般用户只能加大 NI 值，只有超级用户才可以减小 NI 值。NI 值被改变后，会影响 PRI。优先级高的进程被优先运行，缺省时进程的 NI 值为 0。nice 命令的用法如下：

```
nice -n 程序名 以指定的优先级运行程序
```

其中，n 表示 NI 值，正值代表 NI 值增加，负值代表 NI 值减小。
例如：

```
[root@RHEL4 dir1]# nice --1 ps -l
```

5. renice 命令

renice 命令是根据进程的进程号来改变进程的优先级的。renice 的用法如下：


```
renice n 进程号
```

其中，n 为修改后的 NI 值。

例如：

```
[root@RHEL4 dir1]# ps -l
```

	F	S	UID	PID	PPID	C	PRI	NI	ADDR	SZ	WCHAN	TTY	TIME	CMD
	4	S	0	1448	1446	0	75	0	-	-	1501	wait	pts/1	00:00:01
bash														
	4	R	0	2451	1448	0	76	0	-	715	-	pts/1	00:00:00	ps

```
[root@RHEL4 dir1]# renice -6 2451
```

6. top 命令

和 ps 命令不同，top 命令可以实时监控进程的状况。top 屏幕自动每 5 秒钟刷新一次，也可以用“top -d 20”，使得 top 屏幕每 20 秒钟刷新一次。top 屏幕的部分内容：

top - 19:47:03 up 10:50, 3 users, load average: 0.10, 0.07, 0.02

Tasks: 90 total, 1 running, 89 sleeping, 0 stopped, 0 zombie

Cpu (s) : 1.0% us, 2.1% sy, 0.0% ni, 95.8% id, 0.0% wa, 0.0% hi, 1.0% si

Mem: 126212k total, 124520k used, 1692k free, 10116k buffers

Swap: 257032k total, 25796k used, 231236k free, 34312k cached

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2946	root	14	-1	39812	12m	3504	S	1.3	9.8	14:25.46	X
3067	root	25	10	39744	14m	9172	S	1.0	11.8	10:58.34	
rhn-applet-gui											
2449	root	16	0	6156	3328	1460	S	0.3	3.6	0:20.26	hald
3086	root	15	0	23412	7576	6252	S	0.3	6.0	0:18.88	
mixer_applet2											
1446	root	16	0	8728	2508	2064	S	0.3	2.0	0:10.04	sshd
2455	root	16	0	2908	948	756	R	0.3	0.8	0:00.06	top
1	root	16	0	2004	560	480	S	0.0	0.4	0:02.01	init

top 命令前五行的含义如下：

第一行：正常运行时间行。显示系统当前时间，系统已经正常运行的时间，系统当前用户数等。

第二行：进程统计数。显示当前的进程总数，睡眠的进程数，正在运行的进程数，暂停的进程数，僵死的进程数。

第三行：CPU 统计行。包括用户进程，系统进程，修改过 NI 值的进程，空闲进程各自使用 CPU 的百分比。

第四行：内存统计行。包括内存总量，已用内存、空闲内存、共享内存、缓冲区的内存总量。

第五行：交换分区和缓冲分区统计行。包括交换分区总量、已使用的交换分区、空闲交换分区、高速缓冲区总量。

在 top 屏幕下，用 “q” 键可以退出，用 “h” 键可以显示 top 下的帮助信息。

7. bg、jobs、fg 命令

bg 命令用于把进程放到后台运行，例如：

```
[root@RHEL4 dir1]# bg find
```

jobs 命令用于查看在后台运行的进程，例如：

```
[root@RHEL4 dir1]# find / -name aaa &
```

```
[1] 2469
```

```
[root@RHEL4 dir1]# jobs
```

```
[1]+  Running                  find / -name aaa &
```

fg 命令用于把从后台运行的进程调到前台，例如：

```
[root@RHEL4 dir1]# fg find
```

2.5 其他常用命令

除了上面介绍的命令外，还有一些命令也经常用到。

1. clear 命令

clear 命令用于清除字符终端屏幕内容。

2. uname 命令

uname 命令用于显示系统信息。例如：

```
root@RHEL4 dir1]# uname -a
```

```
Linux RHEL4 3.6.9-5.EL #1 Wed Jan 5 19:22:18 EST 2005 i686 i686 i386 GNU/Linux
```

3. man 命令

man 命令用于列出命令的帮助手册。例如：

```
[root@RHEL4 dir1]# man ls
```

典型的 man 手册包含以下几部分：

NAME: 命令的名字。

SYNOPSIS: 名字的概要，简单说明命令的使用方法。

DESCRIPTION: 详细描述命令的使用，如各种参数选项的作用。

SEE ALSO: 列出可能要查看的其他相关的手册页条目。

AUTHOR, COPYRIGHT: 作者和版权等信息。

4. shutdown 命令

shutdown 命令用于在指定时间关闭系统。该命令的语法为：

```
shutdown [参数] 时间 [警告信息]
```

shutdown 命令常用的参数选项有：

- -r: 系统关闭后重新启动。
- -h: 关闭系统。

时间可以是以下几种形式：

- now: 表示立即。
- hh:mm: 指定绝对时间，hh 表示小时，mm 表示分钟。
- +m: 表示 m 分钟以后。

例如：

```
[root@RHEL4 dir1]# shutdown -h now //关闭系统
```

5. halt 命令

halt 命令表示立即停止系统，但该命令不自动关闭电源，需要人工关闭电源。

6. reboot 命令

reboot 命令用于重新启动系统，相当于“shutdown -r now”。

7. poweroff 命令

poweroff 命令用于立即停止系统，并关闭电源，相当于 “shutdown -h now”。

8. alias 命令

alias 命令用于创建命令的别名。该命令的语法为：

```
alias 命令别名 = “命令行”
```

例如：

```
[root@RHEL4 dir1]# alias httpd="vi /etc/httpd/conf/httpd.conf"
```

```
//定义 httpd 为命令"vi /etc/httpd/conf/httpd.conf"的别名
```

alias 命令不带任何参数时，将列出系统已定义的别名。

9. unalias 命令

unalias 命令用于取消别名的定义。例如：

```
[root@RHEL4 dir1]# unalias httpd
```

10. history 命令

history 命令用于显示用户最近执行的命令。可以保留的历史命令数和环境变量 HISTSIZE 有关。只要在编号前加 “!”，就可以重新运行 history 中显示出的命令行。例如：

```
[root@RHEL4 dir1]# !1239
```

表示重新运行第 1239 个历史命令。

2.6 练习题

一、选择题

- _____命令能用来查找在文件 TESTFILE 中包含四个字符的行？
A. grep'????'TESTFILE B. grep'....'TESTFILE
C. grep'^????\$'TESTFILE D. grep'^....\$'TESTFILE
- _____命令用来显示/home 及其子目录下的文件名。
A. ls -a /home B. ls -R /home C. ls -l /home D. ls -d /home
- 如果忘记了 ls 命令的用法，可以采用_____命令获得帮助

- A. ? ls B. help ls C. man ls D. get ls
4. 查看系统当中所有进程的命令是_____。
- A. ps all B. ps aix C. ps auf D. ps aux
5. Linux 中有多个查看文件的命令, 如果希望在查看文件内容过程中用光标可以上下移动来查看文件内容, 则符合要求的那一个命令是_____。
- A. cat B. more C. less D. head
6. _____命令可以了解您在当前目录下还有多大空间。
- A. Use df B. Use du / C. Use du . D. Use df .
7. 假如需要找出 /etc/my.conf 文件属于哪个包 (package), 可以执行_____命令。
- A. rpm -q /etc/my.conf B. rpm -requires /etc/my.conf
- C. rpm -qf /etc/my.conf D. rpm -q | grep /etc/my.conf
8. 在应用程序启动时, _____命令设置进程的优先级。
- A. priority B. nice C. top D. setpri
9. _____命令可以把 f1.txt 复制为 f2.txt?
- A. cp f1.txt | f2.txt B. cat f1.txt | f2.txt
- C. cat f1.txt > f2.txt D. copy f1.txt | f2.txt
10. 使用_____命令可以查看 Linux 的启动信息。
- A. mesg -d B. dmesg C. cat /etc/mesg D. cat /var/mesg

二、简答题

1. more 和 less 命令有何区别?
2. Linux 系统下对磁盘的命名原则是什么?
3. 在网上下载一个 Linux 下的应用软件, 介绍其用途和基本使用方法。

实训 Linux 常用命令

一、实训目的

- (1) 掌握 Linux 各类命令的使用方法;
- (2) 熟悉 Linux 操作环境。

二、实训内容

练习使用 Linux 常用命令, 达到熟练应用的目的。

三、实训环境

- (1) 一台已经安装好 Linux 操作系统的主机, 并且已经配置好基本的 TCP/IP 参数, 能够通过网络连接局域网中或远程的主机。
- (2) 一台 Linux 服务器, 能够提供 FTP、Telnet 和 SSH 连接。

四、实训练习

(1) 文件和目录类命令:

- 启动计算机, 利用 root 用户登录到系统, 进入字符提示界面。
- 用 pwd 命令查看当前所在的目录。
- 用 ls 命令列出此目录下的文件和目录。
- 用 -a 选项列出此目录下包括隐藏文件在内的所有文件和目录。

- 用 `man` 命令查看 `ls` 命令的使用手册。
- 在当前目录下，创建测试目录 `test`。
- 利用 `ls` 命令列出文件和目录，确认 `test` 目录创建成功。
- 进入 `test` 目录，利用 `pwd` 查看当前工作目录。
- 利用 `touch` 命令，在当前目录创建一个新的空文件 `newfile`。
- 利用 `cp` 命令复制系统文件 `/etc/profile` 到当前目录下。
- 复制文件 `profile` 到一个新文件 `profile.bak`，作为备份。
- 用 `ll` 命令以长格形式列出当前目录下的所有文件，注意比较每个文件的长度和创建时间的不同。
- 用 `less` 命令分屏查看文件 `profile` 的内容，注意练习 `less` 命令的各个子命令，例如 `b`、`p`、`q` 等并对 `then` 关键字查找。
- 用 `grep` 命令在 `profile` 文件中对关键字 `then` 进行查询，并与上面的结果比较。
- 给文件 `profile` 创建一个软链接 `lnsprofile` 和一个硬链接 `lnhprofile`。
- 长格形式显示文件 `profile`、`lnsprofile` 和 `lnhprofile` 的详细信息。注意比较 3 个文件链接数的不同。
- 删除文件 `profile`，用长格形式显示文件 `lnsprofile` 和 `lnhprofile` 的详细信息，比较文件 `lnhprofile` 的链接数的变化。
- 用 `less` 命令查看文件 `lnsprofile` 的内容，看看有什么结果。
- 用 `less` 命令查看文件 `lnhprofile` 的内容，看看有什么结果。
- 删除文件 `lnsprofile`，显示当前目录下的文件列表，回到上层目录。
- 用 `tar` 命令把目录 `test` 打包。
- 用 `gzip` 命令把打好的包进行压缩。
- 把文件 `test.tar.gz` 改名为 `backup.tar.gz`。
- 显示当前目录下的文件和目录列表，确认重命名成功。
- 把文件 `backup.tar.gz` 移动到 `test` 目录下。
- 显示当前目录下的文件和目录列表，确认移动成功。
- 进入 `test` 目录，显示目录中的文件列表。
- 把文件 `test.tar.gz` 解包。
- 显示当前目录下的文件和目录列表，复制 `test` 目录为 `testbak` 目录作为备份。
- 查找 `root` 用户自己的主目录下的所有名为 `newfile` 的文件。
- 删除 `test` 子目录下的所有文件。
- 利用 `rmdir` 命令删除空子目录 `test`。
- 回到上层目录，利用 `rm` 命令删除目录 `test` 和其下所有文件。

(2) 系统信息类命令：

- 利用 `date` 命令显示系统当前时间，并修改系统的当前时间。
- 显示当前登录到系统的用户状态。
- 利用 `free` 命令显示内存的使用情况。
- 利用 `df` 命令显示系统的硬盘分区及使用状况。
- 显示当前目录下的各级子目录的硬盘占用情况。

(3) 进程管理类命令：

- 使用 `ps` 命令查看和控制进程：
 - 显示本用户的进程。
 - 显示所有用户的进程。
 - 在后台运行 `cat` 命令。

- 查看进程 cat。
- 杀死进程 cat。
- 再次查看查看进程 cat，看看是否被杀死。
- 使用 top 命令查看和控制进程：
 - 用 top 命令动态显示当前的进程。
 - 只显示用户 user01 的进程（利用 u 键）。
 - 利用 k 键，杀死指定进程号的进程。
- 挂起和恢复进程：
 - 执行命令 cat。
 - 按 Ctrl+z 键，挂起进程 cat。
 - 输入 jobs 命令，查看作业。
 - 输入 bg，把 cat 切换到后台执行。
 - 输入 fg，把 cat 切换到前台执行。
 - 按 Ctrl+c 键，结束进程 cat。
- find 命令的使用：
 - 在/var/lib 目录下查找所有文件其所有者是 games 用户的文件。
 - 在/var 目录下查找所有文件其所有者是 root 用户的文件。
 - 查找所有文件其所有者不是 root，bin 和 student 用户并用长格式显示。
 - 查找/usr/bin 目录下所有大小超过一百万 byte 的文件并用长格式显示。
 - 查找/tmp 目录下属于 student 的所有普通文件，这些文件的修改时间为 120 分钟以前，查询结果用长格式显示。
 - 对于查到的上述文件，用-ok 选项删除。

（4）rpm 软件包的管理：

- 查询系统是否安装了软件包 squid。
- 如果没有安装，则挂载 Linux 安装光盘，安装 squid 软件包。
- 卸载刚刚安装的软件包。
- 软件包的升级。
- 软件包的更新。

（5）tar 命令的使用：

系统上的主硬盘在使用的时候有可怕的噪音，但是它上面有有价值的文件。自从系统在两年半以前备份过，你决定手动备份少数几个最紧要的文件。/tmp 目录里储存在不同的硬盘的分区上快坏的分区，这样你想临时的把文件备份到那里。

- 在/home 目录里，用 find 命令定位文件所有者是 student 的文件。然后将其压缩。
- 保存/etc 目录下的文件到/tmp 目录下。
- 列出两个文件的大小。
- 使用 gzip 压缩文档。

五、实训报告

完成实训报告。