



第二章 shell及常用命令



- 本章内容
 - 2.1Linux终端使用基础
 - 2.2文件与目录的基本概念
 - 2.3目录和文件的基本操作
 - 2.4备份与压缩命令
 - 2.5其他常用命令
 - 2.6联机帮助命令



什么是Linux终端

- **Linux**终端也称为虚拟控制台，一台计算机的输入输出设备就是一个物理的控制台。
- 如果在一台计算机上用软件的方法实现了多个互不干扰独立工作的控制台界面，就是实现了多个虚拟控制台。
- **Linux**终端的工作方式是字符命令行方式，用户通过键盘输入命令进行操作，可以通过**Linux**终端对系统进行控制。



什么是Linux终端（续）

- Linux提供6个虚拟终端，X Window在第7个终端上。
- 虚拟控制台的选择可以通过按下<Alt>键和功能键<Fn>来实现。
- 在X Window图形操作界面中按<Ctrl+ Alt+ Fn>就可以进入控制台字符操作界面。
- 虚拟控制台使得Linux成为一个真正的多用户操作系统。



shell的基本形式

- Shell是一个命令解释器，它可以用来启动、挂起、停止程序，还允许用户编写由shell命令组成的程序。
- shell的种类：
 - bash: 是GNU的Bourne-Again shell，是GNU操作系统上默认的shell。
 - sh: 对bash的符号链接。
 - zsh: UNIX的命令解释器。
 - nologin: 并不是一种shell，作为被禁用用户shell的替代物，用来显示账号不可用信息。



shell的基本形式（续）

- shell命令的基本格式

命令名 [选项] <参数1> <参数2> ...

- 使用命令的基本方法

- Linux命令严格区分大小写

- 使用分号（“；”）将多个命令隔开，在一行中可以输入多个命令

- 命令自动补齐与历史记录

- <Tab>键

- Ctrl+r

- history



shell的基本形式（续）

- Shell提示符：[登录用户@主机名~]提示符
 - root用户登陆成功，出现“#”提示符
 - 普通用户登陆成功，出现“\$”提示符
 - 符号“~”表示当前目录是该用户的主目录
- 输入输出重定向
 - 输入重定向符“<”
 - 输出重定向符“>”和“>>”
- 管道
 - 在Linux系统中，管道是一种先进先出的单向数据通道，符号为“|”，管道可以把前一个命令的输出传递给下一个命令作为输入。
 - 格式：Command1 | Command2 | Command3



文件与文件类型

- 文件是**Linux**用来存储信息的基本结构，它是被命名的存储在某种介质上的一组信息的集合。
- **Linux**系统中有三种基本的文件类型
 - 普通文件：分为文本文件和二进制文件；
 - 目录文件：目录文件存储了一组相关文件的位置、大小等与文件有关的信息；
 - 设备文件：**Linux**系统把每一个I/O设备都看成一个文件，与普通文件一样处理，这样可以使文件与设备的操作尽可能统一。设备文件又可分为块设备文件和字符设备文件。



目录

- **Linux**系统以目录的方式来组织和管理系统中的所有文件。
- 将所有文件的说明信息采用树型结构组织起来整个文件系统有一个“根”（**root**），然后在根上分“杈”（**directory**），任何一个分杈上都可以再分杈，杈上也可以长出“叶子”。“根”和“杈”称为“目录”或“文件夹”。而“叶子”则是一个个的文件。
- **Linux**系统通过目录将系统中所有的文件分级、分层组织在一起，形成了**Linux**文件系统的树型层次结构。以根目录“/”为起点，所有其他的目录都由根目录派生而来。
- 特殊目录：“.”代表该目录自己，“..”代表该目录的父目录，对于根目录，“.”和“..”都代表其自己。



工作目录、用户主目录与路径

- 工作目录：用户登录到Linux系统后，每时每刻都处在某个目录之中，此目录被称为“工作目录”或“当前目录”。
- 用户主目录（**Home Directory**）：是系统管理员在增加用户时为该用户建立起来的目录，每个用户都有自己的主目录。使用符号~表示。



工作目录、用户主目录与路径（续）

- 路径是指从树型目录结构中的某个目录到某个文件的一条道路。此路径的主要构成是目录名称，中间用“/”分开。
 - 绝对路径是指从“根”开始的路径，也称为完全路径；
 - 相对路径是指从用户工作目录开始的路径。
- 通配符
 - 通配符“*”；
 - 通配符“?”；
 - 字符组模式：通配符“[”、“]”、“-”用于构成字符组模式；
 - 转义字符“\”。



文件查看和连接命令cat

- 文件查看和连接命令**cat**

`cat [选项] <file1> ...`

- 主要选项:

- `-n`或`--number`

- `-b`或`--number-nonblank`

- `-s`或`--squeeze-blank`

- `-v`或`--show-nonprinting`

- 例如:

`cat -n textfile1`

`cat -b textfile1 textfile2`



分屏显示命令more

- 分屏显示命令**more**

more [选项] <file>...

- 主要选项:

- <-num>, -d,-l,-f,-p,-c,-s,-u

- +/<string>

- +<num>

- 例如:

more -s textfile

more +20 textfile



按页显示命令less

- 按页显示命令**less**

less [选项] <filename>

- 主要选项:

- **-i**

- **-p<string>**

- **-<显示列数>**

- **-s**

- 例如:

less -s textfile



复制命令cp

- 复制命令**cp**

cp [选项] <source> <dest>

或者 cp [选项] <source>... <directory>

- 主要选项

➤-r

➤-f

➤-a

- 例如:

cp aaa bbb

cp *.c finished



删除命令 rm

- 删除命令rm

rm [选项] <name>...

- 主要选项

➤ -i

➤ -f

➤ -r

- 例如

rm -i *.c

rm -rf myfile



移动或重命名命令mv

- 移动或重命名命令mv

mv [选项] <source> <dest>

或者 mv [选项] <source>... <directory>

- 主要选项

➤-i

➤-f

- 例如

mv -i *.c myproject



创建目录命令 **mkdir**

- 创建目录命令**mkdir**
 mkdir [-p] <dirName>...
- 主要选项
 - **-p**
- 例如
 mkdir AAA
 mkdir -p BBB/CCC



删除空目录命令 **rmdir**

- 删除删除空目录命令**rmdir**

rmdir [-p] <dirName>

- 主要选项

➤ **-p**

- 例如

rmdir AAA



切换工作目录和显示当前路径命令

- 切换工作目录命令**cd**
 `cd <dirName>`
- 例如
 `cd /usr/bin`
 `cd ~`
 `cd ../..`
- 显示当前路径命令**pwd**
 `pwd`



查看目录内容命令ls

- 查看目录命令ls

ls [选项] [<name>...]

- 主要选项

➤ -a, -l, -r, -t

➤ -A, -F, -R

- 例如

ls -lt s*

ls -lR /bin

ls -AF



查找文件或者目录命令 **find** (I)

- 查找文件或者目录命令**find**
 `find [path...] [expression]`
- **expression**中可使用的需要选项
 - `-amin<n>`
 - `-anewer<file>`
 - `-atime<n>`
 - `-cmin<n>`
 - `-cnewer<file>`
 - `-ctime<n>`



查找文件或者目录命令 find (II)

- -empty
- -gid<n> or -group<name>
- -ipath<p>, -path<p>
- -name<name>, -iname<name>
- -size<n单位>
- -type<c>
- -pid<n>
- exp1 exp2或者exp1 -a exp2或者exp1 -and exp2
- !expr或者-not expr
- exp1 -o exp2或者exp1 -or exp2



查找文件或者目录命令 **find** (III)

- 例如

```
find / -name test
```

```
find . -type f
```

```
find . -cmin -20
```

```
find /usr/ -size 50K
```




文件定位命令 **locate** (I)

- 文件定位命令**locate**

locate [选项] <search string>

- 主要选项

- -u

- -U<dir>

- -e<dir1, dir2,...>

- -l<level>

- -i

- -q



文件定位命令 **locate** (II)

➤ **-<n>**

➤ **-r<regexp>**

- 例如

locate test*

locate -n 100 a.out



文件内容检索命令grep

- 命令格式为:

grep [选项] <string> <file>...

其中，**string**是准备检索的字符串或模式，**file**是准备从中检索的文件。

- 常用选项

➤ **-i** : 在进行比较时忽略大小写

➤ **-n**: 在输出结果中给出行号

- 例如

grep Smith phonebook



链接命令ln

- 命令格式为:

ln [选项] <source> <dest>

- 主要选项

➤ -f, -d, -i, -n, -s, -v, -b

- 例如

ln -s yy zz

ln yy xx



创建文件、改变文件或目录时间的命令touch (I)

- 改变文件或目录时间的命令touch

touch [选项] <file1> [file2 ...]

- 主要选项

➤ -a, -m, -c

➤ -r<参考文件或目录>

➤ -d<datestring>

➤ -t<stamp>

➤ --no-create



创建文件、改变文件或目录时间的命令touch（II）

- 例如

`touch file`

`touch -t 0005061803 file`

`touch -d "6:03pm 05/06/2000" file`

`touch -r rfile file`



文件比较命令diff和diff3

- diff命令可以比较两个文件的异同:
diff file1 file2
- diff3命令可以比较三个文件的异同:
diff3 file1 file2 file3



文件排序命令sort

- 命令格式为:

`sort [选项] <file>`

- 主要选项

➤ **-f**: 表示忽略字符的大小写

➤ **-k**: 表示按关键字或字段的位置排序

➤ **-r**: 按照反序排序

➤ **-n**: 表示按照字符串的数字值而不是文字进行排序

- 例如

```
ls -l /var/log/message* |sort -k5 -rn
```




备份命令tar (I)

- 命令格式为:
 tar <主选项> [辅选项] <文件或者目录>
- 主要选项
 - -c, -r, -t, -u, -x
- 辅助选项
 - -f, -k, -m, -v, -w, -z



备份命令tar (II)

- 例如

```
tar -cvf usr.tar /home
```

```
tar -czvf usr.tar.gz /home
```

```
tar -xzvf usr.tar.gz
```

```
tar -tvf usr.tar | more
```



压缩和解压命令gzip

- 命令格式为:
 gzip [选项] <文件名>
- 主要选项
 - **-c**
 - **-d 或 --decompress**
 - **-l, -r, -t, -v**
- 例如
 - cd /home**
 - gzip ***
 - gzip -dv ***
 - gzip -l ***
 - gzip usr.tar**



解压命令unzip

- 命令格式为:
 unzip [选项] <压缩文件>
- 主要选项
 - -d<目录>
 - -n, -o, -j, -v, -t
- 例如
 - unzip text.zip
 - unzip -n -d /tmp text.zip
 - unzip -v text.zip



显示日历命令cal

- 命令格式为:
`cal [选项] [[月] 年]`
- 主要选项
 - -1, -3, -s, -m, -j, -y
- 例如
`cal 2 2005`



日期时间命令date

- 日期时间命令**date**

- 显示日期和时间的命令格式为:

- date [选项] [+FormatString]**

- 设置日期和时间的命令格式为:

- date <SetString>**

- 主要选项

- -d<datestr> 或者 --date=<datestr>

- -s<datestr> 或者 --set=<datestr>

- 例如

- date "+Date is=>%x,Time is=>%X"**

- date -s 14:36:00**

- date -s 061028**



其他常用命令

- 显示文字命令 **echo**
echo [-n] <字符串>
- 清除屏幕命令
clear



软件包管理命令rpm (I)

rpm原来是Red Hat Linux发行版专门用来管理Linux各项套件的程序。

- 安装软件

```
rpm -i ( 或者 --install) [安装选项] <file1.rpm> ...  
      <fileN.rpm>
```

- 删除

```
rpm -e (或者--erase) [删除选项] pkg1 ... pkgN
```




软件包管理命令rpm (II)

- 升级

rpm -U (或者 --upgrade) [升级选项] file1.rpm ...
fileN.rpm

- 查询

rpm -q (或者 --query) [查询选项] pkg1 ... pkgN

- 校验已安装的软件包

rpm -V (或者 --verify) [校验选项] pkg1 ... pkgN



Yum命令

yum是Yellow dog Updater Modified的简称，它可以方便地添加/删除/更新RPM包，能自动解决包的依赖性问题。

- 用yum安装软件包

 - yum install

- 用yum删除软件包

 - yum remove

- 安装所有更新软件

 - yum update



man 命令

- **man**可以显示命令的帮助文档，大多数命令都有一个帮助文档且每个文档又分为不同的“章”
- **man [<章>] <命令>**
- “章”分为以下几种：
 - 1.用户可以操作的文件或者可执行文件
 - 2.系统核心可调用的函数与工具等
 - 3.一些常用的函数和函数库
 - 4.设备说明文件
 - 5.设置文件或是某些文件的格式
 - 6.游戏
 - 7.惯例与协议等说明内容
 - 8.系统管理员可用的管理命令
 - 9.**linux**内核**API**



- 浏览man页命令
 - 空格键：往下翻页
 - PgUp：往上翻页
 - PgDn：往下翻页
 - Home：到最前面
 - End：到最后面
 - /<word>：搜索文本
 - n/N：转到前一项或后一项匹配
 - q：退出



- info

- Info <命令>

- info 帮助的结构和网页相似

- Help

- help [命令]

- 显示shell命令帮助的信息（内部命令）

- <命令> --help

- 显示shell命令帮助的信息（外部命令）

- /usr/share/doc/

- 大多数软件包的文档位于/usr/share/doc/目录中