

第二部分

学习Linux操作系统的基本命令

第4学时 阅读与浏览命令

第5学时 操作与搜索命令

第6学时 使用shell

第7学时 使用X窗口系统

第8学时 探索K Desktop Environment桌面环境

第4学时 阅读与浏览命令

本学时教程介绍用于浏览、搜索和阅读文件系统中的文件和子目录及 Linux分区上的子目录结构的基本的Linux操作系统命令。学习完本学时教程之后，应该知道如何获得某个命令的帮助信息、如何找到Linux操作系统中的位置还有如何在硬盘上找到文件。本学时教程从搜索和浏览子目录的命令开始，接着讲解如何阅读子目录和文件。但是第一步，要先学会如何使用man命令帮助自己。

4.1 使用man命令获取帮助

关于Linux操作系统首先要知道的事情之一是可以非常容易获得帮助。象大多数 UNIX运行版本一样，Linux操作系统发行版本也为几乎每个程序、工具、命令或系统编程调用编制了使用手册页。可以得到几乎所有命令的有关信息，包括 man命令本身。举例来说，输入下列命令就可以阅读man命令的使用手册页：

```
# man man
```

使用手册页就像一页 UNIX的命令汇总。每页使用手册页文件的名称以一个个位数字作为文件后缀，存放在 /usr/man的一个子目录下。许多 OpenLinux操作系统命令的使用手册页都在最初的安装过程中或者在单独安装某个命令程序的时候拷贝到硬盘。比如，man命令的使用手册页文件被命名为：

```
/usr/man/man1/man.1.gz
```



man命令的使用手册页的名称是 man.1，但后缀.gz表示man.1文件已经使用gzip命令压缩过了。man命令在显示使用手册页内容之前会自动解压缩这个文件。如果希望了解更多关于压缩文件方面的资料，请阅读第5学时教程“操作和搜索命令”中的“压缩和解压缩文件”一节。

在子目录/usr/man/man1中存放有超过一千个使用手册页的文件。但是如果想知道使用手册页的格式和存放的位置，在man.1中是找不到的，它们存放在/usr/man/man7的man.7文件中。表4-1给出了每个使用手册组成部分的位置和内容。

表4-1 Linux操作系统的使用手册页的组成部分

目 录	内 容
/usr/man/man1	命令 在shell中执行的命令
/usr/man/man2	系统调用 关于核心函数的文档
/usr/man/man3	库调用 libc函数的使用手册页
/usr/man/man4	特殊文件 关于/dev目录中的文件的信息
/usr/man/man5	文件格式 /etc/passwd和其他文件的详细格式
/usr/man/man6	游戏
/usr/man/man7	宏命令包 对Linux文件系统、使用手册页等的说明
/usr/man/man8	系统管理 根操作员操作的使用手册页
/usr/man/man9	核心例程 关于Linux操作系统内核源例程或者内核模块技术指标的文档

每页使用手册页都传统地分成几个组成段；各种命令的文档也分成几个组成段。表 4-2给出了使用手册页的组织格式。

表4-2 使用手册页的组织格式

组成段	说明
Name	命令的名称及简单说明
Synopsis	如何使用这个命令及命令行参数
Description	对这个程序命令及其参数的解释
Files	这个命令用到的文件清单和它们存放的位置
See Also	有相互联系的使用手册页的清单
Diagnostics	特殊输出情况的说明
Bugs	编程漏洞
Author	程序的主要编写者和其他维护人员

在线使用手册页包含特殊的用于 nroff文本排版程序的排版控制符号，使用特殊的宏命令(文档参见 man.7文件)来控制使用手册页的显示和打印。关于 nroff文本排版程序更多的资料请阅读第15学时“排版打印文档”。nroff格式对其他程序如 makewhatis, whatis和apropos等(这些命令将在本学时教程后面介绍)也极为重要。这些程序要求使用手册页有统一的格式，这样在执行查询操作时信息就可以正确地显示，建立与 OpenLinux操作系统相关的程序数据库的时候数据就可以正确地存储。

还可以在子目录 /usr/doc下找到更多的关于 Linux操作系统的命令和其他主题的详细资料，这个子目录中有那些常见问题答疑 FAQs文件和操作指导 HOWTOs文件。另外，发行GNU软件包的自由软件基金会 (Free Software Foundation, 简称FSF)也把它许多文件存储为一种特殊的超文本信息格式。本书中讨论的许多命令都是 GNU程序。



有许多Linux操作系统使用的软件是来源于自由软件基金会的，它是由emacs编辑器程序(参见第14学时教程“文字处理程序”)的编写者Richard Stallman创立的。FSF组织按照GNU的通用出版许可证(General Public License, 简称GPL)的规定发行它自己的软件。Linux操作系统和GNU软件成功与流行的部分原因就是得益于GPL的规定。如果了解更多关于Linux操作系统中使用的GNU软件程序、FSF、或者GNU的GPL的资料，可以试试info命令，它是一个GNU超本文档的阅读器程序，可以在子目录/usr/info中找到。可以象下面这样使用man命令：

```
man GPL
```

man命令对使用手册页进行检索的时候，其依据主要是根据在子目录 /etc中man.conf文件中详细的原则。这些原则规定了查找使用手册页的缺省子目录。查找这些使用手册页的缺省位置有：

```
MANPATH /usr/man
MANPATH /usr/TeX/man
MANPATH /usr/local/man
MANPATH /usr/X11R6/man
MANPATH /usr/openwin/man
```



在X窗口系统中有一个man命令的图形化界面的版本，叫做xman。不仅可以使用xman阅读使用手册页，还可以看到存放这些使用手册页的子目录的路径名。如果使用的是X11下的KDE桌面环境(在第8学时教程“探索K Desktop Environment桌面环境”中介绍)，将会发现KDE的使用手册页是超文本标记语言格式，即HTML格式的。

4.2 浏览及搜索文件系统

本小节介绍基本的浏览命令并演示在Linux文件系统中移动、查找文件以及如何在Linux操作系统中建立文件信息数据库供诸如whatis之类的命令使用。还会学到具有同样功能的其他方法和程序以及如何加速搜索，尽快找到文件。

4.2.1 使用cd命令进入不同的目录

新术语 cd(切换子目录)命令是在Linux文件系统的不同部分之间移动的基本工具。当登录进入OpenLinux操作系统之后，你处在用户子目录中。这个子目录有一个名字，也就是“路径名”，它是由/home/开头，后面跟着你的用户名。

输入cd命令，后面跟着一个路径名，就可以直接进入另外个子目录中去。举例来说，下面的命令把你带到/usr/bin子目录中去：

```
# cd /usr/bin
```

在/usr/bin子目录中时，可以用下一个命令进入/usr子目录：

```
# cd ..
```

在/usr/bin子目录中你还可以使用下面的命令直接进入根目录，即“/”目录：

```
# cd ../../
```

最后，你总能够用下面的命令回到自己的用户子目录(也就是保存自己文件的子目录)：

```
# cd 或者
```

```
# cd ~
```



如果想使用man命令查看cd命令的使用手册页，会什么也找不到。为什么呢？因为cd命令是内建在shell中的，而shell在敲过回车键之后就解释执行命令行。第6学时教程“使用shell”的bash一节中有更多的详细资料。

4.2.2 使用pwd命令查看自己所在的位置

pwd(显示工作子目录)命令告诉你自己所在的位置，并显示当前工作子目录。例如，如果执行命令：

```
# cd /usr/bin
```

接着输入：

```
# pwd
```

会看到：

```
/usr/bin
```

pwd命令虽然有一个使用手册页，但是实际上每次使用 pwd命令的时候使用的都是一个内建在shell中的pwd命令。怎样来区别它们呢？如果试用下面的方式调用 pwd命令，就会看到它只给出当前的工作子目录：

```
# pwd --help
```

反之，试着调用子目录/bin中的pwd命令，如下所示：

```
# /bin/pwd --help
```

这将执行子目录/bin中的pwd命令而不是内建在 shell中的pwd命令。会看到一个简短的pwd命令的帮助文件而不是当前工作子目录。如果确实搞不清自己到底在什么位置，可以使用子目录/bin中的pwd命令。

4.2.3 使用find命令在子目录中搜索匹配的文件

find命令是一个功能强大的操作命令，可以使用它在硬盘驱动器上查找文件。只用一条很简单的find命令就可以在硬盘驱动器上很容易地查找文件。比如说，如果想查找子目录 /usr中的spell命令，可以采用下面的方法：

```
# find /usr -name spell -print
```

还可以用find命令按照日期查找文件，或者可以指定一个日期范围。例如，如果想查找子目录/usr/bin中最近一百天没有使用过的程序，可以采用下面的命令：

```
# find /usr/bin -type f -atime +100 -print
```

-atime参数后面跟着代表日期的一个数字。如果想在子目录 /usr/bin中查找建立不到一天的新文件或者二十四小时内修改过的文件，可以使用下面的参数：

```
# find /usr/bin -type f -mtime -1 -print
```

-mtime参数后面跟一个表示天数的数字（本例中是一天或者更短的时间），用来查找修改过的文件。find命令的搜索字符串中还接受通配符（如*或?），将在第5学时教程学到这些。举个简单的例子，可以用find命令下面的格式来显示子目录/usr中所有的PostScript格式的文件：

```
# find /usr -name '*.ps' -print
```

将在第16学时教程“图形工具程序”的“了解 Linux操作系统的图形文件格式”小节中讨论PostScript格式。

-xdev是find命令另外一个很便于使用的参数。我们前面所举的查询操作的例子都局限在子目录/usr中。那么如果想从根目录(/)中开始查找会发生什么事情呢？-xdev参数把查询操作限制在当前的文件系统之中，本例中就是 Linux文件系统。如果没有在当前的文件系统中使用-xdev参数，find命令就会不停地检索已挂装的所有的 CD-ROM光盘、DOS和Windows分区，这样就可能会找到一些并不感兴趣的文件、减慢搜索过程、输出杂乱的检索结果。

例如，可以用-xdev参数限制 find命令查找一个 Windows分区。如果希望检索子目录/mny/dos下已经安装Windows中的所有以.sys结尾的文件，可以按照下面的方法使用-xdev参数：

```
# find /mnt/dos -name *.sys -print -xdev
```

find命令还有许多的不同的参数可以使用。



可以在搜索与删除任务中使用 find命令，这样就可以从计算机中删除选定的文件。请阅读第 21 学时教程“系统监管基础”中“节省硬盘空间”一节中的详细内容。

find命令能够非常迅速地检索硬盘驱动器和文件系统，但是另外还有一些方法也可以用来

快速地查找文件，特别是可执行程序。请继续读下去吧。

4.2.4 使用whereis命令查找文件

whereis命令可以迅速地找到文件，而且它还可以提供这个文件的二进制可执行文件、源代码文件和使用手册页存放的位置。例如，下面的命令给出 find命令是放在子目录 /usr/bin中的；而它的使用手册页是放在子目录 /usr/man/man1中的：

```
# whereis find
find : /usr/bin/find /usr/man/man1/find.1.gz
```

可以采用下面的方法让 whereis命令只查找这个程序的二进制可执行文件：

```
# whereis -b find
find : /usr/bin/find
```

这在只想查找某个程序的存储位置而不需要知道它的文档存储位置时特别方便。如果 whereis找不到你让它查找的东西，就会只看见返回一个空字符串，例如：

```
# whereis foo
foo :
```

whereis命令找不到某个文件的部分原因可能是这个文件没有存在于任何 whereis命令搜索的子目录中。whereis命令检索的子目录是固定编写在它的程序中的。虽然这看起来有点象是个缺陷，但把搜索限制在固定的子目录如 /usr/man、/usr/bin或 /usr/sbin中可以加快文件查找的进度。

虽然 whereis 命令在查找程序及其使用手册页时就已经比使用 find命令快了，但是你还可以使用一个更快的搜索功能，那就是将在下一小节讨论的 locate命令。

4.2.5 使用locate命令定位文件

加快文件搜索的方法之一是不去搜索文件子目录！可以使用像 locate命令这样的程序来做到这一点。locate命令使用的是一个文件名数据库，而检索区区一个文件当然要比搜索整个硬盘驱动器要节省时间。使用 locate命令查找文件要比使用 find命令快得多，其原因是 locate命令直接去检索这个数据库文件，从中找到匹配的文件名并显示它的查询结果。

locate命令很容易使用。比如说，如果你想查找你的系统上的所有 PostScript格式的文件，可以使用下面的命令：

```
# locate *.ps
```

文件查询结果几乎是立刻就出现在屏幕上了。我想你会认为 locate命令比 find命令还要容易使用一些。但是还是需要注意它们之间的区别： find命令在Linux操作系统安装完成后可以立刻开始进行文件的查询；而 locate命令则需要先为计算机上所有的文件建立一个数据库。这一点也不必操心，因为建立数据库的工作几乎是完全自动化的。

安装好 OpenLinux操作系统之后，locate命令的数据库文件会在每天凌晨的五点五十分自动更新。如果想为 locate命令的数据库文件立刻生成一个最新的版本，可以使用 updatedb命令。以根操作员的身份登录进入系统，或者使用 su命令(参见第21学时教程“系统监管基础”)取得根操作员的权限之后，在提示符下敲入：

```
# updatedb
```

updatedb命令的操作大概需要一分钟左右才能完成。而当它完成的时候，就会在子目录 /var/state中生成locate命令的数据库文件locatedb。对大约900兆左右的文件容量来说，locatedb文件大约会有70万个字符。使用locate命令唯一的不足是随着时间推移，它的数据库文件会因为

你的计算机系统中添加或者删除文件而跟不上变化。好在OpenLinux操作系统能够自动更新locate命令的数据库文件，具体实现方法请阅读第24学时教程“使用任务计划实现系统管理自动化”。

4.2.6 使用whatis和apropos命令获取命令简介

第一次探索OpenLinux操作系统的时候，可能会遇上一些不太了解其功能作用的程序。虽说大部分的Linux程序在编写的时候都为它的命令行参数“？”、“-help”、或者“--help”多少准备了一点帮助信息，但是如果不了解某个程序的具体作用，一般是不会去运行这个程序的。

whatis命令能够从某个程序的使用手册页中抽出一行简单的介绍性文字，帮助你迅速了解这个程序的具体功能。例如，如果希望了解who命令有什么作用，可以输入下面的内容：

```
# whatis who
```

显示器清屏，然后whatis命令显示下面这样的一行文字：

```
who (1) - show who is logged on
```

按下字母“Q”键退出whatis命令。类似于locate命令，需要使用makewhatis命令——可以在子目录/usr/sbin中找到这个命令——来生成一个命令的简介数据库。如果你想这么做，请以根操作员的身份登录系统并输入：

```
# makewhatis
```

makewhatis命令就象updatedb命令一样也需要几分钟来建立whatis命令使用的数据库文件，这个数据库文件的名字也叫whatis，存放在子目录/usr/man/man1中。makewhatis命令有几个参数，但它没有使用手册页。如果想看看它的快速简介，可以使用下面的方法：

```
# makewhat -?
```

另外，也需要定期更新whatis数据库文件以便跟踪记录那些新安装的程序。第4学时教程“使用任务计划实现系统管理自动化”中介绍了怎样才能让OpenLinux操作系统自动进行这类操作的方法。

现在已经学会了如何使用whereis和whatis命令帮助你来查找程序和了解某个程序的具体功能。可是如果打算执行某个操作，却又想不起来哪个程序能够完成这个操作的时候，该怎么办？在这种情况下，就要求助于apropos命令了。

例如，如果你记不起哪个命令是用来搜索文件的，只需要敲入：

```
# apropos search
```

```
apropos (1) - search the whatis database for strings
badblocks (8) - search a device for bad blocks
bsearch (3) - binary search of a sorted array.
conflict (8) - search for alias/password conflicts
find (1) - search for files in a directory hierarchy
hcreate, hdestroy, hsearch (3) - hash table management
lfind, lsearch (3) - linear search of an array.
lkbib (1) - search bibliographic databases
lookbib (1) - search bibliographic databases
lsearch (n) - See if a list contains a particular element
manpath (1) - determine user's search path for man pages
strpbrk (3) - search a string for any of a set of characters
strspn, strcspn (3) - search a string for a set of characters
tsearch, tfind, tdelete, twalk (3) - manage a binary tree
whatis (1) - search the whatis database for complete words.
zgrep (1) - search possibly compressed files for
➤ a regular expression
zipgrep (1) - search possibly zip files for
➤ a regular expression
```

可以看到一个从whatis数据库文件中挑选出来的程序清单。阅读完这个清单之后，请按

“Q”键退出apropos命令。apropos命令在这个数据库文件中按照给出的关键字进行搜索。如果能保证whatis数据库文件及时更新，就可以使用apropos命令来帮助找到需要的程序。



也可以使用man命令的-K参数来实现apropos命令同样的功能，但其搜索过程会很慢，而且搜索结果给出的是每个含有匹配字符串的使用手册页。例如，如果想检索有那些程序可以处理PostScript格式的数据，可以试用如下的命令

```
# man -K PostScript
```

按“Q”键退出之前会得到如下所示的输出结果：

```
/usr/man/man7/unicode.7? [ynq] n  
/usr/man/man7/suffixes.7? [ynq] n  
/usr/man/man7/groff_char.7? [ynq] n  
/usr/man/man1/convert.1x? [ynq] n  
/usr/man/man1/xv.1? [ynq] n  
/usr/man/man1/xdvi.1? [ynq] n  
/usr/man/man1/dvips.1? [ynq] n  
/usr/man/man1/afm2tfm.1? [ynq] n  
/usr/man/man1/ps2pk.1? [ynq] n  
/usr/man/man1/ps2frag.1? [ynq] q
```

4.3 阅读子目录和文件

现在已经学会了如何进入不同的子目录、查找文件以及如何进一步查找与程序有关的资料。下面将介绍在Linux操作系统中可以使用的其他的基本命令。本小节将介绍怎样列出一个子目录中的内容清单、如何对硬盘驱动器上的文件进行归类以及如何阅读文件的内容等等。为了帮助你入门，我们从学习这些命令的基本格式开始。

4.3.1 使用ls命令列出子目录的内容清单

ls(列出子目录内容清单)命令会是你最经常使用的程序命令之一。使用它最简单的格式，ls命令可以列出当前子目录下几乎所有的文件。虽然这个命令本身只有两个字母，但是它的命令行参数可能比其他任何程序都多(最新统计超过75个)！



为什么会有这么多的Linux命令？为什么这些命令的名称又这么短呢？如果对Linux操作系统不熟悉，很快会发现有许多许多的程序有着很短的，由每一种命令的首字母组合构成的如cd、pwd、df、du、ps、ls、as、ar、at、w等形式的命令名。不要害怕！相反，应该感谢Linux操作系统继承了UNIX中的这么一大笔程序工具的财富。UNIX的哲学是一个好的程序是一个能够非常好地完成一种操作的小代码段；而操作系统的威力则来源于它能够把来自几个程序的输出完美地结合到一起。在选择命令名称的时候，很多情况是根据减少字符输入的原则选定的。一两个字符就可以表达清楚的事为什么需要敲入一大堆程序名字呢？另外，许多早期的UNIX用户是通过非常慢的(波特率只有110)终端线与远端UNIX主机通讯的，所以短小精炼的命令程序可以传输得更加快一些。在学习第6学时教程的时候，会更深地体会到这一点。

ls命令的基本格式列出当前子目录中的文件，如下所示：

```
[ home/bball ] $ ls
News          axhome        nsmail        search
author.msg    documents     reading       vultures.msg
auto          mail          research
```

上面的例子中列出了用户子目录中的文件。

也可以使用 -m 参数把文件用逗号分隔显示在一行上：

```
# ls -m
News, author.msg, auto, axhome, documents, mail, nsmail, reading,
➔ research, search, vultures.msg
```

如果不喜欢ls命令输出的这种排列方式，也可以用 -x 参数按水平对齐的方式而不是缺省的垂直对齐方式列出文件，如下所示：

```
# ls -x
News  author.msg  auto          axhome  documents
mail  nsmail      reading       research search
vultures.msg
```

但是所有列出来这些都是文件吗？其中有没有子目录呢？区分它们的一个办法就是使用 -F 参数：

```
# ls -F
News/          axhome/       nsmail/       search*
author.msg     documents/    reading/      vultures.msg
auto/          mail/         research/
```

正如所看到的，-F 参数使得ls命令可以在显示子目录的时候在它的文件名之后加上一个斜线(“/”)字符。而search文件后面的星号(“*”)字符表示这是一个可执行程序。但是这些就是这个子目录里所有的文件吗？如果希望看到所有的文件，可以将 -a 参数和 -F 参数一起使用，如下所示：

```
# ls -aF
./          .dt/          .neditdb      auto/
../         .dtprofile*  .netscape/   axhome/
.Xauthority .festival_history .newsrc       documents/
.Xdefaults  .forward     .oldnewsrc    mail/
.addressbook .fvwm2rc95* .pinerc       nsmail/
.addressbook.lu .index/     .procmail/    reading/
.bash_history .mailcap    .procmailrc   research/
.bash_logout .mailrc     .tin/         search*
.bash_profile .mime.types .xinitrc*     vultures.msg
.bashrc      .ncftp/     News/
.desksetdefaults .nedit      author.msg
```

正如你所看到的，除非明确地指出显示所有的文件，那些以句号(“.”)开头的文件或者子目录是无法通过ls命令显示出来的。使用 -F 参数是在文件列表清单中查看文件和子目录的一种方法，但是如果有一台彩色显示器，或者使用的是彩色状态下的 X11 的话，就可以让ls命令以不同的颜色来显示文件、子目录、或者可执行文件。如果想这么做，可以像下面这样使用 --color 参数：

```
# ls --color
```

如果在 X11 环境中使用的是 xterm 终端程序，那么子目录将会是蓝色的，程序将会是绿色的，普通文件则会是黑色的。还可以自己设定不同类型的文件使用不同的颜色。

举例来说，如果希望把所有的图形文件如 PCX 或者 GIF 格式的图形显示为红色，并且把所

有以.txt结尾的文本文件显示为淡紫色，需要先建立一个名为 LS_COLORS的shell环境变量——也就是系统或者用户定义的值，定义这些颜色并把这个变量通过下面的办法通知操作系统：

```
# LS_COLORS = " *pcx = 31 : *gif = 31 : *txt = 35 " ; export LS_COLORS
```



在ls命令的使用手册页中给出了缺省的颜色设置值。可以把自己定义的设置值放入.bashrc文件中，这样在下次登录到OpenLinux操作系统中的时候就可以使用新的颜色设置了。作为根操作员，还可以为所有的用户设置系统全局范围的缺省颜色。请阅读第6学时教程去了解如何在OpenLinux操作系统中定义LS_COLORS及其他环境变量供所有的用户使用。

1. 列出文件的全部信息

想了解更多关于文件的信息吗？可以使用ls命令的-l参数来列出文件的全部信息，如下所示：

```
# ls -l
total 2086
-rw-r--r--  1 bball  users      256302 Dec 30 15:42 162317art.zip
drwx-----  6 bball  users        1024 Dec 17 22:35 Desktop
drwxr-xr-x 24 bball  users        1024 Dec  2 15:09 Office40
drwxr-xr-x 22 bball  users        1024 Dec  2 11:23 Office50
-rw-r--r--  1 bball  users      29360 Dec 30 14:38 Sheet.as
-rw-r--r--  1 bball  users        705 Dec 30 15:15 Sheet.wk3
-rw-r--r--  1 bball  users      2360 Dec 30 15:15 Sheet.xl4
-rw-r--r--  1 bball  users      5632 Dec 30 15:20 Sheet.xl5
-rw-r--r--  1 bball  users      2958 Dec 17 20:37 XWp
drwxr-xr-x  5 bball  users        1024 Mar 24 1998 aim
drwxr-xr-x  4 bball  users        1024 Dec 30 15:15 axhome
drwxr-xr-x  2 bball  users        1024 Dec 23 23:15 documents
drwxr-xr-x  2 bball  users        1024 Dec 11 11:06 graphics
drwxr-xr-x  6 bball  users        1024 Dec  1 11:19 lg
drwxrwxrwt  2 bball  users        1024 Dec  9 16:34 mail
drwx-----  2 bball  users        1024 Nov  8 10:56 nsmail
drwxr-xr-x  3 bball  users        1024 Nov 10 16:22 rvplayer5.0
drwxr-xr-x  5 bball  users        1024 Nov  4 10:49 so50
drwxr-xr-x  2 bball  users       3072 Dec 30 15:43 tyl2
-rwxr-xr-x  1 bball  users      12004 Dec 30 12:37 worth.as
-rw-r--r--  1 bball  users      1533 Dec 30 13:09 worth.csv
-rw-r--r--  1 bball  users    227985 Dec 30 13:08 worth.dif
-rw-r--r--  1 bball  users     4905 Dec 30 13:12 worth.sc
-rw-r--r--  1 bball  users     6110 Dec 30 13:09 worth.slk
-rw-r--r--  1 bball  users     1533 Dec 30 13:09 worth.txt
```

正如你所看到的，总共有八个不同的信息栏。第1栏是文件的权限标志，将在第22学时教程“管理文件和文件系统”中讨论。这些标志通常表明了某个文件的类型以及其他的用户是否能够对这个文件进行读、写（修改或者删除）、或者运行等操作。

第2栏给出的是这个文件的链接个数，将在第5学时教程中讨论。第3栏是文件所有者的用户名，第4栏是这个用户所在的用户组组名。所有者和用户组的概念将在第22学时教程中讨论。第5栏给出文件的长度。接下来两栏是这个文件或者子目录创建或者最后一次被修改时的日期和时间。而最后一栏很明显就是这个文件的文件名。

2. 指定其他的子目录

也可以在ls命令的命令行上指定其他的子目录或者路径名，这个方法可以用来查看其他子

目录中的内容。举例来说，如果想查看子目录 /usr/bin 中的全部的文件(看到的可能远不止下面列出的这些)，可以使用下面的方法：

```
# ls /usr/bin
```

```
arch          dd            gzip          netstat       stty
ash           df            hostname      nisdomainname su
awk           dmesg        kill          ping          sync
basename     dnsdomainname ksh           ps            tar
bash         doexec       ln            pwd           tcsh
bsh          domainname   login         red           touch
cat          echo         ls            rm            true
chgrp       ed           mail          rmdir        umount
chmod       egrep        mkdir         rpm           uname
chown      false        mknod        sed           ypdomainname
cp          fgrep        more          setserial    zcat
cpio        gawk         mount         sh
csh         grep         mt            sleep
date        gunzip       mv            sort
```

新术语 规则表达式 一种特殊的字符串描述格式语言。ls命令支持使用通配符和规则表达式，也就是说可以使用与那些你在学习 find和locate命令时看到的例子中相类似、但更加复杂的参数形式。例如，如果只想列出当前子目录中的文本文件，可以使用下面的方法：

```
# ls *.txt
```

最后，如果想看到计算机系统中全部的文件，可以使用ls命令的-R参数，它将逐层进入所有的子目录并把其中的内容全部显示出来。虽然你可以使用这个方法检索文件和归类系统中的文件，但我得告诉你显示的文件可能需要好几分钟。而它的输出结果可能还包括那些不想列出的文件或者是其他操作系统的文件系统如DOS或Windows上的文件，特别是在使用了下面的命令的时候：

```
# ls -R /
```

一个比较好的办法是把-d参数和-R参数一起使用，这样可以只列出有限的子目录层次下的文件。例如，下面的命令将沿着根目录(即“/”目录)列出三层子目录中的内容：

```
# ls -Rd /*/*/*
```

另外，还有一个更好的获得你系统上的目录结构图的办法——tree命令——我们马上就要讲到。



是否喜欢ls -aF显示子目录的方式？是否喜欢ls命令一直使用彩色？
如果希望ls命令能够一直保持这类特色，请阅读第6学时教程。

4.3.2 使用dir和vdir命令列出子目录的内容清单

如果不喜欢在列出子目录内容清单的时候使用ls命令那么多的参数，可以使用dir或vdir命令。与ls命令相比，这两个命令大概只有45个命令行参数，但在功能上它们毫不逊色。它们在使用上与ls命令类似，但是使用了一些缺省的参数。

使用dir命令的时候就像普通的ls命令那样把文件按列对齐输出，如下所示：

```
# dir
01_hyph.dat  49_thes.dat  bin          help          store
01_spell.dat Desktop      config       install.ini   template
01_thes.dat  LICENSE      database     instdb.ins    wordbook
```

```

44_hyph.dat  README          download      kino          xp3
44_spell.dat addin           explorer     l_office.xpm
44_thes.dat  autotext       filter       lib
49_hyph.dat  backup         fonts        s_office.xpm
49_spell.dat basic          gallery      sofficerc

```

使用vdir命令的时候就像ls命令的-l参数，它缺省地列出文件的全部信息，如下所示：

```

# vdir
total 3659
-rw-rw-rw-  1 bball  users      65605 Oct 27 10:00 01_hyph.dat
-rw-rw-rw-  1 bball  users     313015 Oct 27 10:00 01_spell.dat
-rw-rw-rw-  1 bball  users     375808 Oct 27 10:00 01_thes.dat
-rw-r--r--  1 bball  users      65605 Oct 27 10:00 44_hyph.dat
-rw-r--r--  1 bball  users     313015 Oct 27 10:00 44_spell.dat
-rw-r--r--  1 bball  users     380928 Oct 27 10:00 44_thes.dat
-rw-rw-rw-  1 bball  users      51765 Oct 27 10:00 49_hyph.dat
-rw-rw-rw-  1 bball  users     602497 Oct 27 10:00 49_spell.dat
-rw-rw-rw-  1 bball  users    1191936 Oct 27 10:00 49_thes.dat
drwxr-xr-x  2 bball  users       1024 Dec  2 11:21 Desktop
-rw-r--r--  1 bball  users     11631 Oct 27 10:00 LICENSE
-rw-r--r--  1 bball  users     8435 Oct 27 10:00 README
drwxr-xr-x  3 bball  users       1024 Dec  2 11:21 addin
drwxr-xr-x  2 bball  users       1024 Dec  2 11:21 autotext
drwxr-xr-x  2 bball  users       1024 Dec 30 15:41 backup
drwxr-xr-x  2 bball  users       1024 Dec 17 11:43 basic
drwxr-xr-x  2 bball  users       2048 Dec  2 11:22 bin
drwxr-xr-x 11 bball  users       3072 Jan  4 09:52 config
drwxr-xr-x  2 bball  users       1024 Dec  2 11:30 database
drwxr-xr-x  2 bball  users       1024 Dec  2 11:20 download
drwxr-xr-x  5 bball  users       1024 Dec  2 11:28 explorer
drwxr-xr-x  2 bball  users       1024 Dec  2 11:22 filter
drwxr-xr-x  4 bball  users       1024 Dec  2 11:20 fonts
drwxr-xr-x 14 bball  users       2048 Dec  2 11:22 gallery
drwxr-xr-x  4 bball  users       1024 Dec  2 11:21 help
-rwxr-xr-x  1 bball  users     15569 Dec  2 11:22 install.ini
-rw-r--r--  1 bball  users    274209 Dec 17 11:20 instdb.ins
drwxr-xr-x  2 bball  users       1024 Dec  2 11:21 kino
-rw-r--r--  1 bball  users     5199 Oct 27 10:00 l_office.xpm
drwxr-xr-x  2 bball  users       3072 Dec  2 11:21 lib
-rw-r--r--  1 bball  users       812 Oct 27 10:00 s_office.xpm
-rwxr-xr-x  1 bball  users     8406 Jan  4 09:52 sofficerc
drwxr-xr-x  3 bball  users       1024 Dec  2 11:38 store
drwxr-xr-x 13 bball  users       1024 Dec 17 11:43 template
drwxr-xr-x  2 bball  users       1024 Dec  2 12:08 wordbook
drwxr-xr-x  5 bball  users       1024 Dec  2 11:20 xp3

```

虽然找不到dir和vdir命令它们各自的使用手册页（在ls的使用手册页里提到了它们），但是可以使用--help参数得到它们各自的帮助信息。

4.3.3 使用cat命令列出及合并文件

新术语 cat(合并文件)命令用来把文件内容送到你的屏幕上去显示。你也可以使用这个命令通过一个叫做输出“重定向”的过程把文件内容送到其他的文件中。本小节只介绍这个命令的一些基本用法。请阅读第6学时教程中对标准输入、标准输出和重定向等概念的详细讲解。虽然cat命令在阅读短文件的内容的时候很有用，但它更经常地被用来进行文件的合并、建立、覆盖或者添加内容等操作。如果想使用cat命令查看一个短文件，可以输入如下的命令：

```

# cat test.txt
This text file was created by the cat command .

```

```
Cat could be the world's simplest text editor .
If you read this book , you ' ll learn how to use cat .
This is the last line of text in this file .
```

cat命令也有一些参数。如果想在阅读文件的时候加上行号，比方想要注明某个特殊的段落，可以使用-n参数：

```
# cat-n test.txt
 1 This text file was created by the cat command .
 2 Cat could be the world's simplest text editor .
 3 If you read this book , you ' ll learn how to use cat .
 4 This is the last line of text in this file .
```

还可以使用cat命令一次查看多个文件，因为cat命令也接受通配符：

```
# cat -n test*
 1 This text file was created by the cat command .
 2 Cat could be the world's simplest text editor .
 3 If you read this book , you ' ll learn how to use cat .
 4 This is the last line of text in this file .
 5 This is the first line of test2.txt .
 6 This file was also created by cat .
 7 This is the last line of test2.txt .
```

正如所看到的，cat命令在它的输出中包含了另外一个文件，并为整个输出结果的每一行(不是每个文件)加上了行号。请注意还可以使用下面的方法来查看这两个文件：

```
# cat test.txt test2.txt
```

这个命令的输出结果与使用通配符的时候看到的结果是完全一样的。

但是同时查看几个文件只是cat命令的一种用法。还可以使用cat命令和重定向操作符(“>”)来合并文件。例如，如果要把文件test.txt和文件test2.txt合并到另外一个叫做test3.txt的文件中去，可以使用下面的方法：

```
# cat test* > test3.txt
```

可以用下面的方法检查操作结果：

```
# ls -l test*
-r w - r - - r - - 1 bball users    190 Jan 4 16 : 06  test.txt
-r w - r - - r - - 1 bball users    108 Jan 4 16 : 07  test2.txt
-r w - r - - r - - 1 bball users    298 Jan 4 16 : 07  test3.txt
```

但是如果只是想把test.txt和test2.txt文件合并，但是并不想再生成另外一个更大的文件的时候又该怎么办呢？如果是这种情况，首先需要决定是把test.txt的内容加到文件test2.txt中去，还是把test2.txt的内容加入test.txt中去。然后，使用cat命令和重定向符“>>”，敲入下面的内容：

```
# cat test.txt >> test2.txt
```

上面的命令把文件test.txt的内容添加到文件test2.txt的后面去了。可以再使用cat命令检查一下操作结果：

```
# cat test2.txt
This is the first line of test2.txt .
This file was also created by cat .
This is the last line of test2.txt .
This text file was created by the cat command .
Cat could be the world ' s simplest text editor .
```

```
If you read this book , you ' ll learn how to use cat .  
This is the last line of text in this file .
```

需要注意的是如果使用了下面的命令：

```
# cat -n test.txt >> test2.txt
```

现在test2.txt文件的内容会是这样的：

```
# cat test2.txt  
This is the first line of test2.txt .  
This file was also created by cat .  
This is the last line of test2.txt .  
    1 This text file was created by the cat command .  
    2 Cat could be the world's simplest text editor .  
    3 If you read this book , you ' ll learn how to use cat .  
    4 This is the last line of text in this file .
```

最后，介绍一个小技巧，可以使用这个技巧在不使用字处理或者文本编辑程序的情况下建立一个短的文本文件。因为 cat命令可以读取标准输入，所以可以使用 cat命令建立一个文件并通过键盘直接向这个文件中输入内容。我们以 myfile.txt为示例文件说明如下。先输入以下命令：

```
# cat > myfile.txt
```

现在，输入一些文本内容，如下所示：

```
This is the cat word processor .  
This is the end of the file.
```

现在，按下Ctrl+D组合键关闭这个文件。使用下面的命令看看是否完成了操作：

```
# ls -l myfile.txt  
-r w -r - -r - - 1 bball users    61 Jan 4 16 : 11          myfile.txt  
# cat myfile.txt  
This is the cat word processor .  
This is the end of the file.
```

cat命令可以把任何文件(不仅仅是文本文件)的内容显示出来。虽然 cat命令在查看一个或者几个短文件的时候很有用，但是当你想读取一个很长的文本文件的时候，又该怎么办呢？我们继续往下学习，将会学习到页命令，它们能够使阅读长文件的工作变得简单些。

4.3.4 使用more命令阅读文件

新术语 more命令是Linux操作系统命令中我们称之为页命令的家族中的一员。页命令使你在浏览文件的时候可以一次阅读一屏或者一行。这在阅读大量使用手册页的时候特别有帮助，因为man命令是使用了一个页命令来显示每一页的。

more命令是一个传统意义上的页命令，因为它提供了早期页命令的基本特色，可以在命令行上像下面这样使用 more命令，如下所示：

```
# more longfile.txt
```

如果需要帮助，请按下“H”键，将看到一个帮助画面。如果使用了惊叹号(!)，还可以从more命令中去执行其他的命令。阅读一个文本文件是相当容易的，因为可以敲空格键阅读后一页，也可以敲B键阅读前一页。

more命令也有一些命令行参数。在各种常见的参数之外，还可以自行设置屏幕提示(more命令会显示正在阅读的文件的可读百分比)、设置屏幕画面大小(前后翻阅文本时显示的行数)、

使用多个文件名或通配符及打开或关闭滚屏功能。

虽然可能会觉得在阅读文件方面 more 命令的表现相当不错，可是 less 命令可能会更让你喜欢。

4.3.5 使用 less 命令浏览文件

less 命令多少有点象 more 命令，但是 less 命令的功能比 more 命令更强大。less 命令和 more 命令一样都是页命令。但是它的编写者 Mark Nudelman，改进了 more 命令中的一些特色，并又添加了许多其他的特色。

less 命令提供了许多比 more 命令优秀的功能：

- 可以使用光标键在文本文件中前后滚屏。
- 可以用行号或百分比作为书签来浏览文件。
- 可以实现在多个文件中进行复杂的检索、格式匹配、高亮度显示等操作。
- 键盘操作与字处理程序如 emacs 兼容。
- 阅读到文件结束或者标准输入结束的时候 less 命令不会退出。
- 屏幕底部的信息提示更容易控制使用，而且提供了更多的信息。
- 带有许多的附件，包括一个独立的键定义程序 lesskey，这样就可以定义使用哪些按键来控制 less 命令。

当安装了 Linux 操作系统之后，less 页命令将是许多程序（比如 man 命令等）使用的缺省的页命令。如果想阅读压缩文件（那些带有 .gz 后缀的文件，我们将在第 5 学时教程中学习），可以使用保存在子目录 /usr/bin 下的 zless 命令。

作为演示，首先使用 ls 命令建立一个内容为子目录 /usr/bin 中所有 OpenLinux 操作系统命令的文本文件，如下所示：

```
# ls /usr/bin > programs.txt
```

上面的命令建立了一个名为 programs.txt 的文本文件。然后，使用 less 命令阅读这个文件，如下所示：

```
# less programs.txt
```

less 命令装入这个文件并显示这个文件开头的一些行。如果想向下翻一页，按空格键。如果想向上翻一页，按 B 键。也可以用光标键向前、后、甚至左右移动。在一般的情况下，less 命令的命令提示符是显示在屏幕左下角的一个冒号 (:)。总之，可以使用命令行参数 -M 看到更多关于文件的信息，如下所示：

```
# less -M programs.txt
```

现在，当浏览文件的时候，less 命令将显示出这个文件的名称、当前页码及总的页码，表示当前位置在整个文件中的位置的百分比数值。这个提示符看起来有点象下面这个样子：

```
programs.txt line 91/1221 8%
```

这表明正在阅读的是文件 programs.txt，当前屏幕最顶端显示的是总数为 1221 行文本的第 91 行。如果想运行其他的程序，比如 wc 字数统计程序，需要敲入一个惊叹号 (!)，后面再跟上命令行，然后按下回车键，如下所示：

```
! wc programs.txt
```

wc 程序将在屏幕上回显文件 programs.txt 的行数、字数和字符数。当这个命令执行完毕之后，less 命令显示单词 done(完成)并等你按下回车键。

还可以使用less搜索命令在一个文本文件中进行快速查找。先按下斜杠键 (/), 再输入一个单词或者词组的一部分。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, 请按下斜杠键 (/), 再按下回车键。如果想退出阅读, 请按下Q键, 就返回到shell命令行。

4.4 使用head或tail命令阅读文件的开头和结尾

虽然head和tail命令并不是页命令, 但是只想阅读一个文件的开头或者结尾的部分的时候, 它们可以让工作变得更简单一些。虽然这两个程序象大多数 Linux操作系统命令一样, 只是被设计用来完成一两个功能的, 但是它们的确做得很好。

除了传统的用来显示文件开头 x行的-n x参数之外, head命令还有一些其他的参数。在Linux操作系统发行版本中的head命令是GNU文本工具包的一个组成部分, 它还可以从一个文件的开头显示任何数目的512个字符、1K(1024个字符)或者兆字节规模大小的文本块。象cat命令一样, head命令也可以处理二进制文件。如果按照传统方式使用head命令, 它就会从一个或几个文件的开头节选几行文本。比如, 如果想看看Linux操作系统安装指南中每个HTML文件的开头的两行, 应该使用head命令的参数-n 2, 再输入这些文件的路径, 如下所示:

```
# head -n 2 /usr/doc/LDP/install-guide/install-guide-3.2.html/*.html
==> /usr/doc/LDP/install-guide/install-guide-3.2.html/
↳footnode.html <==
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<!--Converted with LaTeX2HTML 96.1-h (September 30, 1996) by Nikos
↳Drakos (nikos@cbl.leeds.ac.uk), CBLU, University of Leeds -->

==> /usr/doc/LDP/install-guide/install-guide-3.2.html/gs.html <==
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<!--Converted with LaTeX2HTML 96.1-h (September 30, 1996) by Nikos
↳Drakos (nikos@cbl.leeds.ac.uk), CBLU, University of Leeds -->
```

请注意, 我在上面只给出了头两个文件的输出结果, 并且head命令的缺省输出是包括了文件名的(放在==>和<==之间)。如果只是想阅读那些不带文件名的信息, 请使用参数-q, 如下所示:

```
# head -n 2 /usr/doc/LDP/install-guide/install-guide-3.2.html/*.html
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<!--Converted with LaTeX2HTML 96.1-h (September 30, 1996) by Nikos
↳Drakos (nikos@cbl.leeds.ac.uk), CBLU, University of Leeds -->

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<!--Converted with LaTeX2HTML 96.1-h (September 30, 1996) by Nikos
↳Drakos (nikos@cbl.leeds.ac.uk), CBLU, University of Leeds -->
```

当面临的任务是阅读一个很长的文件, 而最有用的信息却是在这个文件的结尾部分的时候, 使用tail命令就特别有帮助了。我们举个例子, 假定需要在系统信息中查看出错信息。在子目录/var/log中有一个message文件, 这里面有系统操作的详细记录, 但是登录记录是在这个messages文件的末尾更新的。换句话说, 这个文件的内容是不断添加上去的, 所有最新的信息都在文件的结尾部分。如果想使用tail命令查看这个messages文件末尾最后的十二行, 必须先以根操作员的身份登录, 然后再输入下面的命令:

```
# tail -n 12 /var/log/messages
Nov 12 21:02:02 localhost cardmgr[152]: initializing socket 0
Nov 12 21:02:02 localhost cardmgr[152]: socket 0: ATA/IDE
↳Fixed Disk Card
Nov 12 21:02:02 localhost cardmgr[152]: executing: 'insmod /
```

```
➔lib/modules/2.0.30/p
cmcia/fixed_cs.o'
Nov 12 21:02:03 localhost kernel: hdc: SunDisk SDCFB-4,
➔ 3MB w/1kB Cache, LBA, CH
S=123/2/32
Nov 12 21:02:03 localhost kernel: ide1 at 0x100-0x107,0x10e on irq 3
Nov 12 21:02:03 localhost kernel: hdc: hdc1
Nov 12 21:02:03 localhost cardmgr[152]: executing:
➔ './fixed start hdc'
Nov 12 21:02:03 localhost cardmgr[152]: initializing socket 1
Nov 12 21:02:03 localhost cardmgr[152]: socket 1: Serial or
➔Modem Card
Nov 12 21:02:03 localhost kernel: tty01 at 0x02f8 (irq = 5)
➔is a 16550A
Nov 12 21:02:03 localhost cardmgr[152]: executing:
➔ 'insmod /lib/modules/2.0.30/p
cmcia/serial_cs.o'
Nov 12 21:30:17 localhost PAM_pwdb[556]: (su) session opened
➔for user root by bball(uid=0)
```



如果你试用这个例子的时候，OpenLinux给出下面的信息：

```
tail : /var/log/messages : Permission denied
```

这意味着必须以根操作员的身份登录！因为安全的原因，OpenLinux操作系统禁止无关的用户阅读系统登录记录。因此，可以试试使用下面的命令：

```
su -c tail -n 12 /var/log/messages
```

当按下回车键之后，su命令会向你提问根操作员口令字。输入根操作员口令字，再按下回车键，就可以查看信息了。这是一个以根操作员身份运行命令的好方法。

象这样阅读大型文件的确是很方便的，特别是在系统的 messages文件有可能增大到一百万个字符以上的情况下。

4.5 课时小结

本学时教程介绍了十多个可以在OpenLinux操作系统中使用的基本命令。知道如何获得帮助、怎样列出子目录内容清单以及如何在文件系统中移动等都是非常重要的技能。在继续学习OpenLinux操作系统的过程中，将会反复地用到许多这样的命令。

4.6 专家答疑

问：我在子目录/usr/doc中的不同的子目录下发现许多文件名为 README的文件，怎样才能阅读某个README文件？

答：在命令行中使用less页命令，后面再跟上想阅读的文件的文件名。比如说，如果想阅读关于pdks shell的README文件，输入下面这样一个less命令的命令行，然后按下回车键：

```
less /usr/dos/pdks-5.2.13/README
```

问：有一些路径名实在太长了！有没有一个容易点的阅读或者列文件清单的方法？

答：还使用上面的例子，可以象下面那样在路径名中使用一个星号（*）：

```
less /usr/doc/pdksh*/REDAME
```

问：我使用了上面的例子，但是是这样使用星号的：`less /usr/doc/p /README`，而`less`命令说它在读总数是三十五个文件的第一个！怎么回事，我怎样才能阅读到其他文件？

答：在子目录`/usr/doc`下至少有37个不同的子目录是以字母`p`开头的。很明显，有25个子目录中包含有名称为`REDAME`的文件。在`less`页命令中，如果想跳到下一个文件，需要输入一个冒号(:)，再按下`N`键。

问：怎样才能只列出子目录的名称？比如说，我想看看子目录`/usr/doc`下所有以字母`z`开头的子目录的名称。

答：使用`less`命令的`-d`参数，即子目录命令行参数。如果想查看所有名称以字母`z`开头的子目录，请输入下面的命令：

```
ls -d /usr/doc/z*
```

4.7 练习题

1. 使用`cd`命令进入子目录`/usr/bin`。然后输入`cd`并按下回车键。现在你在哪儿？
2. 接着，再用`cd`命令进入`/`目录。输入`cd ~`并按下回车键。再输入`cd -`并按下回车键。现在你在哪儿？
3. `cat`目录的`-n`参数会把显示在屏幕上面的文本行加上行号。怎样才能在使用`less`命令阅读的文件中加上行号？