

# 第 3 章

## Ubuntu 字符界面

Ubuntu 不但为用户提供了友好的图形界面,而且提供了功能强大的字符界面。在第 2 章中,读者学习了在图形界面下使用 Ubuntu 进行日常操作,本章将在 Ubuntu 的字符界面下介绍各种基本命令。具体包括用户的登录和注销、文件和目录的操作,以及如何进行磁盘管理、如何获取帮助及其他常用命令。

### 本章学习目标

- 熟练掌握字符界面常用的登录和注销命令,如 login、logout、halt 和 shutdown 等;
- 熟练掌握字符界面常用的目录与文件命令,如 ls、cd、rm 和 mv 等;
- 熟练掌握字符界面常用的文件内容显示命令,如 cat、more、less、head 和 tail 等;
- 熟练掌握字符界面常用的文件内容处理命令,如 sort、grep、diff、wc 和 cut 等;
- 掌握字符界面常用的文件查找命令,如 find、locate、whereis 和 which 等;
- 掌握字符界面常用的磁盘管理命令,如 df、du、mount 和 unmount 等;
- 掌握字符界面常用的备份压缩命令,如 zip、unzip、gzip、gunzip 和 bzip2 等;
- 了解字符界面常用的获得帮助命令,如 man、whatis、help 和 info 等;
- 了解字符界面常用的其他命令,如 clear、echo、date 和 history 等。

### 3.1 Ubuntu 命令简介

Ubuntu 的命令可根据不同的分类标准进行分类,若从其与 Shell 程序的关系这一标准来分类,可分为内部命令和外部命令。内部命令常驻内存,它们是 Shell 程序的一部分,这些命令由 Shell 程序识别并可在其内部完成运行;外部命令是 Ubuntu 系统中的实用程序,它通常不被包含在 Shell 中,和内部命令相比,其使用频率较低,因此仅在用户需要使用时才将其调入内存,而不像内部命令那样常驻内存。

本章将日常使用的 Ubuntu 命令按功能粗略地分为以下八大类:

(1) 登录与注销:本功能包含的典型命令有用户登录、用户注销、用户登录口令的修改、系统启动与关闭。

(2) 目录与文件:对工作目录的操作主要包括创建、移动、复制、删除和显示当前工作目录,对文件的操作则包括移动、复制与删除。

(3) 文件内容显示:Ubuntu 可提供多种方式显示文件内容,包括分页往后显示文件,分页自由显示文件,指定显示文件的前、后若干行。

(4) 文件内容处理: 对文件内容进行处理是 Ubuntu 十分重要的功能之一, 大致包括文件内容的排序、查重, 不同文件的比较, 以及对文件内容进行剪切、粘贴和统计。

(5) 文件查找: 文件查找功能是 Ubuntu 日常使用时最为频繁的操作之一, 分别包括在硬盘和数据库中查找文件或目录。

(6) 磁盘管理: 对磁盘进行高效管理是 Ubuntu 日常使用中必不可少的功能之一, 具体包括检查、统计磁盘空间的占用情况, 检查和关闭磁盘的使用空间限制、挂载与卸载文件系统。

(7) 备份压缩实用程序: 备份压缩是 Ubuntu 中十分重要的功能之一, 典型的实用程序有 zip、unzip 和 zipinfo, gzip、gunzip 和 gzexe, bzip2、bunzip2 和 bzip2recover, compress 和 uncompress, uuencode 和 undecode, dump 和 jar。

(8) 其他命令: 无法纳入上述 7 项, 但日常使用较为频繁的命令主要包括显示文本、显示日期和时间、查看和终止进程、显示最近登录系统的用户信息和历史命令(或称指令)等。

接下来, 我们将按上述分类对 Ubuntu 命令进行介绍。

## 3.2 登录与注销

本节将介绍在 Ubuntu 字符界面下如何登录系统、注销用户、退出当前 Shell、修改登录口令、关闭及重启系统。

### 3.2.1 用户登录

在上一章中我们学习了如何在 Ubuntu 图形界面下通过输入用户名和密码登录系统, 若用户名和密码正确无误, 则用户在图形化界面成功登录系统后默认将显示如图 3-1 所示。

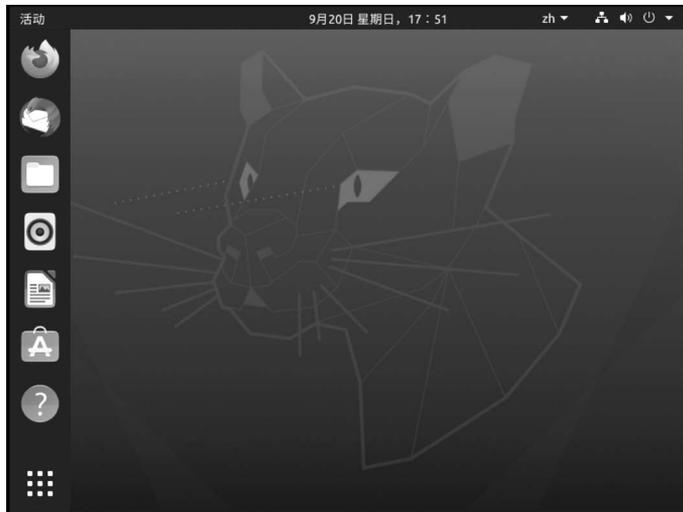


图 3-1 用户成功登录 Ubuntu 系统后的桌面

此时, 用户同时按下“Ctrl+Alt+F1”将打开 Ubuntu 字符界面, 提示输入用户名和密码, 如图 3-2 所示。

接下来, 输入用户名(默认使用安装系统时的用户名)并按 Enter 键, 系统提示输入密码

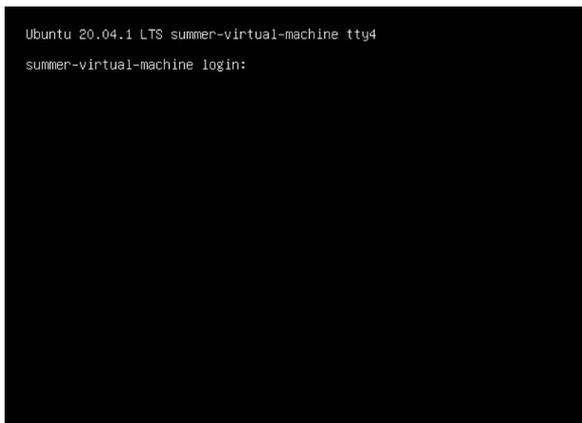


图 3-2 Ubuntu 字符界面下的登录窗口

(默认使用系统安装时设定的密码),输入密码时默认不显示所输入的字符(此时用户只需键入与当前用户名对应的密码即可,请注意这是 Ubuntu 系统的一大特色,而非 bug。此举可有效应对当前用户输入密码时旁人偷窥而导致的密码泄露问题)。无论是输入的用户名错误还是密码错误,系统均提示“Login incorrect”,如图 3-3 所示。

仅当用户名存在且和输入的密码对应并正确时,才会进入如图 3-4 所示的成功登录系统的界面。系统将显示当前用户最后一次登录的时间、当前 Ubuntu 的版本号以及可用的更新等相关信息。

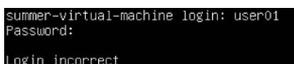


图 3-3 登录失败界面

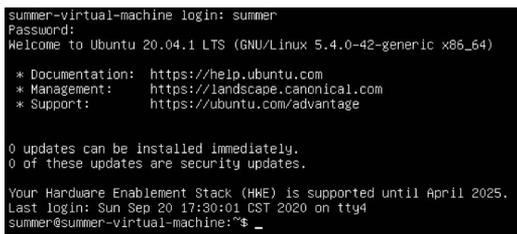


图 3-4 登录成功界面

**注意:** 在 Ubuntu 系统中默认是不允许 root 登录的,因为它具有最高权限,默认不被用来登录系统。

我们在使用时通常仅能以普通用户和访客的身份登录系统,如果希望以 root 身份登录系统,那么需要按以下步骤进行操作:

(1) 普通用户登录后,按下“Ctrl+Alt+F1”打开新窗口或者“Ctrl+Alt+T”打开一个新终端。

(2) 如果用户在安装 Ubuntu 系统时没有设置 root 用户的密码,则须执行以下命令设置 root 用户的密码。

```
sudo passwd root
```

在用户两次输入相同的为 root 用户设置的密码后,将会提示“passwd: password updated successfully”,这表明 root 用户密码设置成功。

(3) 在新窗口或者新终端输入 su,再输入 root 用户登录的密码,按 Enter 键即可进入

root 用户权限模式。

```
ninelle@ubuntu:~$ sudo passwd root
[sudo] password for ninelle:
New password:
Retype new password:
passwd: password updated successfully
ninellet@ubuntu:~$ su
Password:
root@ubuntu:~/home/ninelle#
```

图 3-5 设置 root 用户密码并进入 root 模式

上述为 root 用户设置密码并登录的过程如图 3-5 所示。

从图中可以看到,当前用户由普通用户变成 root 用户。

无论是通过打开新窗口还是新终端登录 Ubuntu 系统,用户都只有成功登录之后,才可以在字符界面下使用该系统。由于 root 具有系统操作的最高权限,为了避免使用 root 进行操作时可能导致的系统文件被修改或者删除,甚至对系统造成不可恢复的影响,如修改配置文件导致系统无法启动、删除分区信息导致系统崩溃。我们强烈建议用户在日常使用 Ubuntu 系统时只要有可能,均不使用 root 账户,而是使用系统安装时要求用户输入的用户名(默认用户)和密码来进行操作。

通常除了使用默认用户进行日常操作以外,我们还需要为其他有可能临时使用该系统的用户创建一些账户,并使用这些账户登录系统进行各种操作,具体如下。

#### (1) 使用 useradd 命令创建临时账户

root 用户可以通过执行以下命令来创建临时账户 useraddtest,效果如图 3-6 所示。

```
useradd useraddtest
```

#### (2) 使用 adduser 命令创建临时账户

root 用户可以通过执行以下命令来创建临时账户 addusertest,效果如图 3-7 所示。

```
adduser addusertest
```

```
root@ubuntu:~/home/user01# useradd useraddtest
```

图 3-6 使用 useradd 命令创建临时账户

```
root@ubuntu:~/home/user01# adduser addusertest
Adding user 'addusertest' ...
Adding new group 'addusertest' (1004) ...
Adding new user 'addusertest' (1004) with group 'addusertest' ...
Creating home directory '/home/addusertest' ...
Copying files from '/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for addusertest
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] y
```

图 3-7 使用 adduser 命令创建临时账户

根据系统提示输入新用户的信息,包括密码、全名、电话号码等,最后系统提示是否输入正确,输入 Y 则代表确认无误,n 则代表重新输入。

**注意:** useradd 和 adduser 命令都可以创建账户,但是两者是有区别的。

① 在使用 useradd 创建用户时不会在 /home 下自动创建与该账户的用户名同名的用户目录,而且不会自动选择 Shell 版本,也不会设置密码,因此这个账户创建和默认是不能被立即使用的,而是需要使用 passwd(后续我们将介绍此命令)修改密码后才可使用。

② 在使用 adduser 创建账户时则不存在上述问题,即账户创建后可以立即使用。需要特别留意的是,一般用户不具有创建账户的权限,需要使用 root 账户才能完成上述创建账户的操作。创建账户完成后,即可使用这个账户的用户名和密码进行登录。

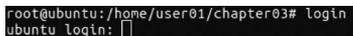
(3) 使用 login 实现登录和不同账户间的切换

执行以下命令：

```
login
```

效果如图 3-8 所示。

如果要由 root 切换至该用户,使用 login 命令即可进入登录界面,正确输入用户名和密码后即可对对应的账户进行操作。



```
root@ubuntu:/home/user01/chapter03# login
ubuntu login: █
```

图 3-8 login 命令切换账户

### 3.2.2 用户注销

在 Ubuntu 系统中注销命令是 logout。登录系统后,若要离开系统,用户只要直接下达 logout 命令即可,注销后重新返回登录命令行。

用户可执行以下注销命令。

```
logout
```

该命令执行后的效果如图 3-9 所示。

**注意:** logout 命令只在命令行界面使用,如果在 Shell 中使用该命令,那么系统会出现如图 3-10 所示的提示信息。

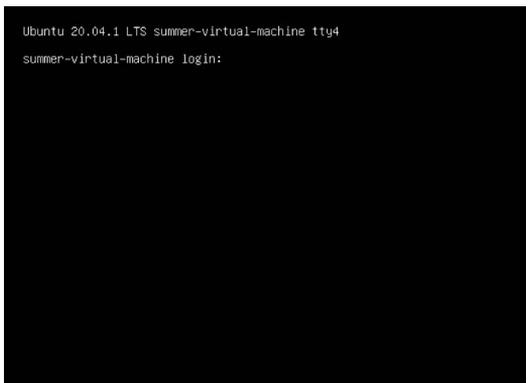


图 3-9 用户注销



```
user01@ubuntu:~$ logout
bash: logout: not login shell: use 'exit'
```

图 3-10 Shell 下执行 logout 命令的提示

### 3.2.3 退出当前 Shell

使用 login 登录系统,在退出系统时我们可以选择 exit 或者 logout,这两者之间有区别吗?对于大部分系统来说这两者之间是没有区别的,但其实 logout 命令的主要功能是注销用户,而 exit 命令则是退出控制台。

(1) 按下“Ctrl+Alt+T”打开一个 Shell 窗口,效果如图 3-11 所示。

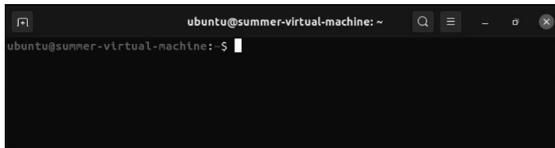


图 3-11 退出当前 Shell

(2) 打开该 Shell 窗口之后,我们输入以下命令并按 Enter 键执行之。

```
exit
```

在 Shell 窗口执行 exit 命令后的效果如图 3-12 所示。

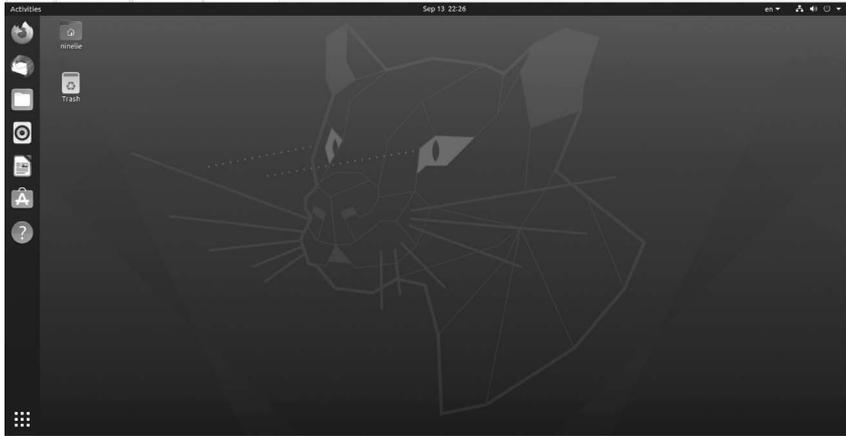


图 3-12 退出 Shell

可以看到,执行 exit 命令后,系统退出并关闭了当前 Shell 窗口,返回到图形界面的桌面,而当前账户并没有发生任何改变。

### 3.2.4 修改登录口令

在实际使用 Ubuntu 系统时,如果用户需要修改登录的口令,可以使用 passwd 命令。

执行以下命令:

```
passwd user01
```

效果如图 3-13 所示,系统要求用户输入当前密码,如果该密码正确则会提示用户输入新密码,否则会出现如图 3-13 所示的“Authentication token manipulation error”的提示,此时用户修改密码失败。

若用户输入原密码正确,则要求重复输入两次新密码且新密码不能与原密码相同,否则如图 3-14 所示。此时系统提示原密码未被修改 (Password unchanged),并要求用户再次输入新密码。

```
ubuntu@summer-virtual-machine:~$ passwd ubuntu
Changing password for ubuntu.
Current password:
passwd: Authentication token manipulation error
passwd: password unchanged
```

图 3-13 输入错误原密码的提示

```
ubuntu@summer-virtual-machine:~$ passwd ubuntu
Changing password for ubuntu.
Current password:
New password:
Retype new password:
Password unchanged
New password: █
```

图 3-14 新密码与旧密码相同时出现的提示

如果两次输入的新密码不一致,那么会出现如图 3-15 所示的情况。此时系统提示“password do not match”,此时用户修改密码失败。

正确完成所有输入后,如果出现如图 3-16 所示的“password updated successfully”的提示,那么表示用户成功修改密码。

```
ubuntu@summer-virtual-machine:~$ passwd ubuntu
Changing password for ubuntu.
Current password:
New password:
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged
```

图 3-15 因两次输入的新密码不一致而出现的提示

```
ubuntu@summer-virtual-machine:~$ passwd ubuntu
Changing password for ubuntu.
Current password:
New password:
Retype new password:
passwd: password updated successfully
```

图 3-16 成功修改密码

### 3.2.5 关闭系统

有些用户会使用直接关掉电源的方式来关闭 Ubuntu 系统,这是十分危险的。因为 Ubuntu 后台运行着许多进程,所以强制关机可能会导致这些进程中的数据丢失,进而使系统处于不稳定的状态,甚至在有的系统中会损坏硬件设备。

Ubuntu 系统中有一些常用的关机命令如 shutdown(-h)、halt 及 poweroff,它们都可以关闭系统。执行上述关机命令后,系统将自动关闭所有进程并关机,接下来我们来学习这些关机命令。

(1) 首先执行以下命令。

```
halt
```

halt 命令执行后,系统并没有如我们所想的那样自动关闭,而是出现了如图 3-17 所示的提示。这是因为 halt 命令只有 root 用户才可以执行,普通用户无权使用该命令。此时我们只需使用前面介绍过的 su 命令切换至 root 用户即可。

```
ubuntu@summer-virtual-machine:~$ halt
Failed to halt system via logind: Interactive authentication required.
Failed to open initctl fifo: Permission denied
Failed to talk to init daemon.
```

图 3-17 halt 命令执行后的提示

**注意:** 使用普通用户执行该命令时,会弹出认证对话框,若输入相应的密码,则会自动关闭系统。

(2) 切换至 root 用户后再次执行以下命令。

```
halt
```

命令成功执行后,系统将会在关闭所有进程后自动关机。

(3) 开机后登录系统并打开新终端,同时执行以下命令。

```
shutdown -h 10
```

系统将在 10 分钟后自动关机,如图 3-18 所示。如果想取消自动关机,则在关机前输入 shutdown -c 即可。

```
ubuntu@summer-virtual-machine:~$ shutdown -h 10
Shutdown scheduled for Sun 2020-09-20 19:01:39 CST, use 'shutdown -c' to cancel
```

图 3-18 设置 10 分钟后自动关机

(4) 最后,我们尝试输入 poweroff 命令以实现系统的关闭。

```
poweroff
```

该命令执行后,将会得到与以 root 用户去执行 halt 命令的相同结果,即系统将会自动关闭。

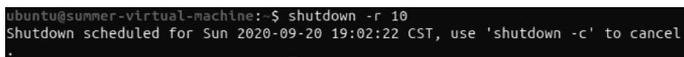
### 3.2.6 重启系统

当需要对系统进行重启的时候可以使用重启命令 `shutdown(-r)`、`init` 和 `reboot`。

(1) 首先执行以下命令。

```
shutdown -r 10
```

该命令执行后的效果如图 3-19 所示。



```
ubuntu@summer-virtual-machine:~$ shutdown -r 10
Shutdown scheduled for Sun 2020-09-20 19:02:22 CST, use 'shutdown -c' to cancel
```

图 3-19 设置 10 分钟后自动重启

这条命令的意思是系统将在 10 分钟后自动重启。如果想取消自动重启,那么在重启前输入 `shutdown -c` 命令即可。

(2) 继续输入 `init` 命令实现重启。

```
init 6
```

该命令执行完成后,系统将自动重启。

`init` 命令除了加上参数 6 可以实现重启以外,还有如表 3-1 所示的几个级别,它们具有不同的功能。

表 3-1 `init` 命令参数及其对应的功能

参数	功 能	参 数	功 能
0	停机	4	图形化
1	单用户模式	5	安全模式
2	多用户,不能使用 net file system	6	重启
3	完全多用户		

(3) 最后,使用 `reboot` 命令对系统进行重启,执行以下命令。

```
reboot
```

该命令执行后,会得到与 `init 6` 命令相同的结果,即系统将自动重启。

## 3.3 目录与文件

在完成上一节的学习之后,用户可以在 Ubuntu 字符界面下登录系统。在用户使用该系统时,会频繁地对目录与文件进行操作,本节将介绍目录与文件相关的 Shell 命令,包括显示、更改、创建和删除工作目录,移动、复制、删除、创建目录或文件等。

### 3.3.1 显示当前工作目录

用户在字符界面下进行操作时,通常无法知道当前所在的目录,此时可以使用 `pwd` 命令来显示当前工作目录的完整路径。

请执行以下命令。

```
pwd
```

该命令没有任何选项和参数,执行后将显示完整路径,效果如图 3-20 所示。

```
user01@ubuntu:~$ pwd
/home/user01
```

图 3-20 显示当前目录

可以看到,系统显示完整路径为“/home/user01”。

### 3.3.2 更改工作目录

cd 命令是 Ubuntu 中较为基本的目录操作命令之一,用户可以使用此命令改变当前工作目录,即从当前工作目录切换至 cd 命令中参数指定的目录。

(1) 首先执行以下命令。

```
cd chapter03
```

该命令执行后的效果如图 3-21 所示。

```
user01@ubuntu:~$ cd chapter03
user01@ubuntu:~/chapter03$
```

图 3-21 更改工作目录

(2) cd 命令执行完毕后,系统已切换至指定目录,输入以下命令可以查看。

```
pwd
```

效果如图 3-22 所示,可以看到,刚刚执行的 cd 命令成功地将当前目录由“/home/user01”更改为“/home/user01/chapter03”。

(3) 如图 3-23 所示,如果用户想要从当前目录切换至一个并不存在的目录 chapter033,那么命令将无法正确执行,并出现“No such file or directory”的提示。

```
user01@ubuntu:~/chapter03$ pwd
/home/user01/chapter03
```

图 3-22 查看当前目录

```
user01@ubuntu:~$ cd chapter033
-bash: cd: chapter033: No such file or directory
```

图 3-23 切换至不存在的目录出现的系统提示

### 3.3.3 创建工作目录

在 Ubuntu 下,我们可以使用 mkdir 命令创建目录,若执行成功,它会在当前目录下创建新的目录。

(1) 首先执行以下命令。

```
mkdir chapter03
```

该命令执行后的效果如图 3-24 所示。

如果在当前目录下用户想要创建的目录已经存在,那么将无法成功执行该命令,并且会出现如图 3-25 所示的系统提示。

```
user01@ubuntu:~$ mkdir chapter03
```

图 3-24 创建目录

```
user01@ubuntu:~$ mkdir chapter03
mkdir: cannot create directory 'chapter03': File exists
```

图 3-25 创建的文件夹已存在的提示

若用户想创建的目录在当前目录下不存在,那么在创建目录命令成功执行后,系统并不会出现任何提示。

(2) 我们可以在当前目录下使用 mkdir 命令,并使用相对路径创建目录。请执行以下命令。

```
mkdir ./chapter03/chapter030303 或 mkdir chapter03/chapter030303
```

执行后出现如图 3-26 所示的提示,这是因为普通用户没有权限,需要使用 root 账户进

行操作。

```
user01@ubuntu:~$ mkdir ./chapter03/chapter030303
mkdir: cannot create directory './chapter03/chapter030303': Permission denied
```

图 3-26 相对路径创建目录

(3) 使用 su 命令切换至 root 用户后,再次执行以下命令。

```
mkdir ./chapter03/chapter030303 或 mkdir chapter03/chapter030303
```

效果如图 3-27 所示,我们成功地在 chapter03 目录下创建了目录 chapter030303。

```
root@ubuntu:/home/user01# mkdir ./chapter03/chapter030303
```

图 3-27 以相对路径方式创建目录

使用相对路径后,我们不用转换目录就可以在任何具有读写权限的文件夹里创建目录。

(4) 我们也可以同时创建多个目录,比如我们要创建的目录 chapter030301、chapter030302 和 chapter030303,请执行以下命令。

```
mkdir chapter030301 chapter030302 chapter030303
```

从图 3-28 可以看到,这 3 个目录将同时被创建。

```
user01@ubuntu:~$ mkdir chapter030301 chapter030302 chapter030303
```

图 3-28 同时创建多个目录

**注意:** 使用 mkdir 命令操作时,若硬盘空间已满,则无法正确执行此操作,系统会给出相应提示。

### 3.3.4 删除工作目录

在 Ubuntu 系统中,我们可以使用 rmdir 命令删除一个空目录。

(1) 首先在当前目录下执行 ls 命令查看所有目录和文件,效果如图 3-29 所示。

```
user01@ubuntu:~$ ls
chapter03  chapter3.1  Desktop  examples.desktop  Public
chapter030301  chapter3.2  Documents  Music  Templates
chapter030302  chapter3.3  Downloads  Pictures  Videos
```

图 3-29 显示当前目录下所有目录和文件

(2) 然后执行以下命令删除目录。

```
rmdir chapter03
```

该命令执行后的效果如图 3-30 所示。

(3) 再次执行 ls 命令列出所有文件和目录。

```
ls
```

该命令执行后的效果如图 3-31 所示,可以看到,chapter03 这个目录已经被删除,说明命令执行成功。

```
user01@ubuntu:~$ rmdir chapter03
```

图 3-30 删除工作目录

```
user01@ubuntu:~$ ls
chapter04  chapter3.3  Downloads  Pictures  Videos
chapter3.1  Desktop  examples.desktop  Public
chapter3.2  Documents  Music  Templates
```

图 3-31 显示当前目录下所有目录和文件

(4) 需要注意的是,若用户想要删除的目录不存在时,则不能成功执行 rmdir 命令,而

是会出现如图 3-32 所示的“No such file or directory”的提示。

```
user01@ubuntu:~$ rmdir chapter033
rmdir: failed to remove 'chapter033': No such file or directory
```

图 3-32 删除不存在的目录出现的系统提示

### 3.3.5 移动目录或文件

如果在 Ubuntu 系统中想让一个目录向另一个目录移动文件,或对某一文件进行重命名操作,可以使用 mv 命令。该命令的功能是把文件移动到指定的目录或对文件进行重命名。

(1) 首先使用 mkdir 命令创建一个名为 chapter030305 的目录,然后使用 touch 命令(该命令在后续章节将会进行介绍)创建一个名为 ex030305 的文件,命令如下。

```
mkdir chapter030305
touch ex030305
```

(2) 然后执行 ls 命令列出所有文件和目录,如图 3-33 所示。

```
user01@ubuntu:~$ ls
chapter03  chapter3.1  Desktop  ex030305  Pictures  Videos
chapter030301  chapter3.2  Documents  examples.desktop  Public
chapter030302  chapter3.3  Downloads  Music  Templates
```

图 3-33 显示当前目录下所有目录和文件

(3) 接下来执行以下命令。

```
mv ex030305 chapter030305
```

(4) 我们可以使用 ls 命令查看当前目录,再使用 cd 命令切换至 chapter030305 目录,最后执行 ls 命令查看 chapter030305 目录中的文件,具体如下。

```
ls
cd chapter030305
ls
```

从图 3-33 中可以看到当前工作目录下存在名为 ex030305 的文件,执行 ls 和 mv 命令后,效果如图 3-34 所示,这也就是说我们成功地将当前工作目录下的 ex030305 文件移动到了当前工作目录的子目录 chapter030305 中。

```
user01@ubuntu:~$ ls
chapter03  chapter3.1  Documents  examples.desktop  Templates
chapter030301  chapter3.2  Downloads  Music  Videos
chapter030302  chapter3.3  ex030308  Pictures
chapter030305  Desktop  ex030311  Public
user01@ubuntu:~$ cd chapter030305
user01@ubuntu:~/chapter030305$ ls
ex030305
```

图 3-34 查看文件原目录和新目录下的文件

(5) 需要注意的是,若用户想要将 ex030305 移动至不存在的目录 chapter030306 中时,该操作并不能成功执行,而是会出现如图 3-35 所示的“Not a directory”的提示。

```
user01@ubuntu:~/chapter03/chapter030305$ cd ex030305 chapter030306
-bash: cd: ex030305: Not a directory
```

图 3-35 移动文件至不存在的目录时出现的提示

**注意:** mv 命令也可用于文件的重命名操作,读者可自行尝试。

### 3.3.6 复制目录或文件

在 Ubuntu 系统中复制是十分重要的操作之一,使用 `cp` 命令可以实现目录或文件的复制,请读者按以下步骤进行。

(1) 首先在当前目录下执行 `ls` 命令查看所有目录和文件,效果如图 3-36 所示。

```
user01@ubuntu:~$ ls
chapter03  chapter3.1  Desktop  ex030305  Pictures  Videos
chapter030301  chapter3.2  Documents  examples.desktop  Public
chapter030302  chapter3.3  Downloads  Music  Templates
```

图 3-36 显示当前目录下的所有目录和文件

(2) 然后使用 `mkdir` 命令创建一个名为 `chapter030306` 的目录,再使用 `touch` 命令创建一个名为 `ex030306` 的文件,具体如前所述。

(3) 创建 `chapter030306` 目录和 `ex030306` 文件后,在当前目录下执行以下命令。

```
cp ex030306 ./chapter030306
```

或

```
cp ex030306 chapter030306
```

此命令执行后,并没有复制文件到 `chapter030306` 目录中,而是出现如图 3-37 所示的提示。

```
ubuntu@summer-virtual-machine:~$ cp ex030306 ./chapter030306
cp: cannot create regular file './chapter030306/ex030306': Permission denied
```

图 3-37 执行 `cp` 命令时系统提示无权限

(4) 切换至 `root` 用户后再次执行上述命令。

```
cp ex030306 ./chapter030306
```

或

```
cp ex030306 chapter030306
```

(5) 在当前目录中使用 `ls` 命令查看目录中的所有文件,再切换至 `chapter030306` 目录下执行 `ls` 命令查看所有文件,执行如下命令。

```
ls
cd chapter030306
ls
```

上述命令执行后的效果如图 3-38 所示。

```
user01@ubuntu:~/chapter03$ ls
chapter030305  chapter030306  chapter030307  ex030306
user01@ubuntu:~/chapter03$ cd chapter030306
user01@ubuntu:~/chapter03/chapter030306$ ls
ex030306
```

图 3-38 查看 `cp` 命令执行情况

命令执行后,我们可以发现 `chapter03` 目录下的 `ex030306` 文件依然存在,而 `chapter030306` 目录下多了一个内容和 `ex030306` 完全相同的文件,说明复制命令执行成功。

(6) `cp` 命令还可以复制一个文件后并重命名,那么当前目录下就存在原文件和重命名后的原文件,命令如下。

```
cp ex030306 ex030306_01
```

```
ls
```

该命令执行后的效果如图 3-39 所示。

```
user01@ubuntu:~$ cp ex030306 ex030306_01
user01@ubuntu:~$ ls
chapter03      ex030311      ex03050402    ex030506_03    ex030802~
chapter030301 ex030401      ex030504_02   ex030506_04    ex030803.bz2
chapter030302 ex030402      ex03050403    ex030507        ex030804
chapter030305 ex030403      ex030504_03   ex03050801     ex030805
chapter3.1     ex030404      ex03050501    ex030508_01    examples.desktop
chapter3.2     ex030405      ex030505_01   ex03050802     Music
chapter3.3     ex030406      ex03050502    ex030508_02    Pictures
Desktop        ex030501      ex030505_02   ex030509        Public
Documents      ex030502      ex03050601    ex030602        Templates
Downloads      ex030503      ex030506_01   ex030801        Videos
ex030306      ex03050401    ex03050602    ex030801.zip
ex030306_01   ex030504_01   ex030506_02   ex030802
```

图 3-39 使用 cp 命令复制并重命名新文件

可以看到,命令执行后,ex030306 依然存在,而它的复制文件也被在当前目录创建,并被重命名为“ex030306\_01”。

### 3.3.7 删除目录或文件

在 Ubuntu 中很多人仍然习惯用 rmdir 执行删除目录这一任务,但是 rmdir 命令只能删除空目录(即目录下不包含任何文件和子目录),如果被删除的目录不是空目录,那么将无法使用该命令完成这一操作。

(1) 首先使用 ls 命令查看 chapter030305 目录下的文件和子目录,发现该目录下有一名为 ex030305 的文件,如图 3-40 所示。

(2) 然后执行 rmdir 命令试图删除 chapter030305 文件夹,会发现系统出现如图 3-41 所示的提示,即“Directory not empty”。我们只需清空该目录内的所有文件(包括子目录及子目录下的所有文件),即可正确执行该命令。

```
user01@ubuntu:~/chapter030305$ ls
ex030305
```

图 3-40 显示 chapter030305 文件夹内的文件和子目录

```
user01@ubuntu:~$ rmdir chapter030305
rmdir: failed to remove 'chapter030305': Directory not empty
```

图 3-41 使用 rmdir 命令删除含有文件的目录时出现的提示

(3) 若一个目录中既含有文件又含有子目录,那么如果想删除该目录,就得删除该目录下所有文件和子目录(此时子目录必须为空),若子目录不为空,则需递归删除之,会耗费大量的时间和精力。使用 rm -rf 命令则可跳过这一步骤直接删除目录。首先,执行以下命令查看当前目录和 chapter03 目录下的文件及目录,效果如图 3-42 所示。

```
ls
cd chapter03
ls
```

```
user01@ubuntu:~$ ls
chapter03  chapter3.1  Desktop  ex030305  Pictures  Videos
chapter030301  chapter3.2  Documents  examples.desktop  Public
chapter030302  chapter3.3  Downloads  Music      Templates
user01@ubuntu:~$ cd chapter03
user01@ubuntu:~/chapter03$ ls
chapter030305  chapter030306  chapter030307  ex030306
```

图 3-42 查看当前目录和 chapter03 目录下的文件及目录

(4) 再执行 cd .. 命令回到前一目录下。

```
cd ..
```

(5) 然后,执行 `rm -rf` 命令删除 `chapter03` 目录及其目录下所有文件和目录。

```
rm -rf chapter03
```

(6) 最后,执行 `ls` 命令,查看文件和目录,如图 3-43 所示。

```
ls
```

```
user01@ubuntu:~$ ls
chapter030301  chapter3.2  Documents  examples.desktop  Public
chapter030302  chapter3.3  Downloads  Music              Templates
chapter3.1     Desktop     ex030305   Pictures           Videos
```

图 3-43 删除指定目录

可以发现 `chapter03` 目录已经不存在了,并且文件夹内的文件和文件夹都被删除了,说明命令执行成功。

**注意:** 使用 `rm -rf` 命令的时候一定要格外小心,因为使用该命令删除的文件是不放入 Ubuntu 系统的回收站的,因此若想恢复这些被删除的文件有些困难。

### 3.3.8 创建文件或修改时间

在 Ubuntu 中我们经常使用 `touch` 命令创建一个文件,此命令也可更改文档或目录的日期时间。

(1) 首先执行 `ls` 命令查看当前工作目录下的所有文件,具体如下。

```
ls
```

效果如图 3-44 所示。

```
user01@ubuntu:~$ ls
chapter03  chapter030305  chapter3.3  Downloads  Music  Templates
chapter030301  chapter3.1  Desktop  ex030311  Pictures  Videos
chapter030302  chapter3.2  Documents  examples.desktop  Public
```

图 3-44 查看当前工作目录下的所有文件

命令执行后,可以看到当前工作目录下不存在名为 `ex030308` 的文件。

(2) 然后执行 `touch` 命令创建一个名为 `ex030308` 的文件,并查看当前工作目录下的所有文件,具体如下。

```
touch ex030308
```

```
ls
```

命令执行后可以看到文件创建成功,效果如图 3-45 所示。

```
user01@ubuntu:~$ touch ex030308
user01@ubuntu:~$ ls
chapter03  chapter3.1  Desktop  ex030305  Music  Templates
chapter030301  chapter3.2  Documents  ex030308  Pictures  Videos
chapter030302  chapter3.3  Downloads  examples.desktop  Public
```

图 3-45 touch 命令创建文件

(3) 若想使用 `touch` 命令修改文件最后被编辑的时间,那么就得加上 `-at` 参数。在修改前先查看一下该文件最后被编辑的时间和访问的时间。

可执行如下命令用来查看 `ex030308` 这一文件最后被编辑的时间和访问的时间,效果如图 3-46 所示。

```
ls -l ex030308
```

```
ls -lu ex030308
```

```
user01@ubuntu:~$ ls -l ex030308
-rw-rw-r-- 1 user01 user01 0 May  8 22:19 ex030308
user01@ubuntu:~$ ls -lu ex030308
-rw-rw-r-- 1 user01 user01 0 May  8 22:19 ex030308
```

图 3-46 查看文件被编辑的时间和访问的时间

(4) 接着,可执行如下命令将文件的最后访问时间修改为4月1日10点11分。

```
touch -at 04011011 ex030308
```

(5) 执行如下命令可查看更新后的文件的最后被编辑时间和访问时间。

```
ls -l ex030308
ls -lu ex030308
```

该命令执行后的效果如图3-47所示。

```
user01@ubuntu:~$ touch -at 04011011 ex030308
user01@ubuntu:~$ ls -l ex030308
-rw-rw-r-- 1 user01 user01 0 May  8 22:19 ex030308
user01@ubuntu:~$ ls -lu ex030308
-rw-rw-r-- 1 user01 user01 0 Apr  1 10:11 ex030308
```

图 3-47 修改最后访问时间并显示最后被编辑的时间和访问的时间

从图中可以看到,上述文件最后被编辑的时间没有变动,但是最后访问的时间通过执行上述命令已被修改。

### 3.3.9 查看目录和文件

查看目录和文件的命令是ls,它在Ubuntu中是使用频率较高的命令。ls命令的输出信息可以进行彩色加亮显示,以区分不同类型的文件,下面介绍几种常见的用法。

(1) 执行以下命令查看目录和文件。

```
ls
```

该命令执行后的效果如图3-48所示。

```
user01@ubuntu:~$ ls
chapter03  chapter3.1  Desktop  ex030305  Music  Templates
chapter030301  chapter3.2  Documents  ex030308  Pictures  Videos
chapter030302  chapter3.3  Downloads  examples.desktop  Public
```

图 3-48 查看目录和文件

(2) 执行以下命令可以查看所有文件(包括以“.”开头的隐藏文件)。

```
ls -a
```

该命令执行后的效果如图3-49所示。

```
user01@ubuntu:~$ ls -a
.          chapter030301  .config  examples.desktop  .profile
..         chapter030302  Desktop  .gnupg            Public
.bash_history  chapter030305  .dirc   .gnupg            Templates
.bash_logout  chapter3.1     Documents  .ICEauthority     Videos
.bashrc       chapter3.2     Downloads  .local            .Xauthority
.cache        chapter3.3     ex030311  Music             .xsession-errors
chapter03    .compiz        ex030401  Pictures          .xsession-errors.old
```

图 3-49 查看所有文件(包括隐藏文件)

(3) 执行以下命令可以显示文件索引节点号。

```
ls -li
```

该命令执行后的效果如图3-50所示。

(4) 执行以下命令可以列出文件的详细信息,如创建者、创建时间、文件的读写权限列

表等。

```
ls -l
```

该命令执行后的效果如图 3-51 所示。

```
user01@ubuntu:~$ ls -l
540388 chapter03      540442 chapter3.3  537036 examples.desktop
537052 chapter030301 537005 Desktop    537022 Music
537057 chapter030302 537020 Documents  537024 Pictures
540463 chapter030305 537008 Downloads  537018 Public
540431 chapter3.1      540449 ex030311    537010 Templates
540433 chapter3.2      540415 ex030401    537044 Videos
```

图 3-50 显示文件索引节点号

```
user01@ubuntu:~$ ls -l
total 76
drwxrwxr-x 2 user01 user01 4096 May  8 22:19 chapter03
drwxrwxr-x 2 user01 user01 4096 May  7 20:40 chapter030301
drwxrwxr-x 2 user01 user01 4096 May  7 20:40 chapter030302
drwxrwxr-x 2 user01 user01 4096 May  9 07:03 chapter030305
drwxrwxr-x 2 user01 user01 4096 May  7 06:54 chapter3.1
drwxrwxr-x 2 user01 user01 4096 May  7 06:54 chapter3.2
drwxrwxr-x 2 user01 user01 4096 May  7 06:54 chapter3.3
drwxr-xr-x 2 user01 user01 4096 May  1 22:18 Desktop
drwxr-xr-x 2 user01 user01 4096 May  1 22:18 Documents
drwxr-xr-x 2 user01 user01 4096 May  1 22:18 Downloads
-rw-rw-r-- 1 user01 user01   0 May  8 23:31 ex030311
-rw-rw-r-- 1 user01 user01   8 May 11 21:34 ex030401
-rwxrwxrwx 1 user01 user01   0 May 12 21:55 ex030402
-rw-r--r-- 1 user01 user01 8980 Apr 29 08:26 examples.desktop
drwxr-xr-x 2 user01 user01 4096 May  1 22:18 Music
drwxr-xr-x 2 user01 user01 4096 May  1 22:18 Pictures
drwxr-xr-x 2 user01 user01 4096 May  1 22:18 Public
drwxr-xr-x 2 user01 user01 4096 May  1 22:18 Templates
drwxr-xr-x 2 user01 user01 4096 May  1 22:18 Videos
```

图 3-51 列出文件的详细信息

(5) 执行以下命令可以用“,”号区隔每个文件和目录的名称。

```
ls -m
```

该命令执行后的效果如图 3-52 所示。

```
user01@ubuntu:~$ ls -m
chapter03, chapter030301, chapter030302, chapter030305, chapter3.1, chapter3.2,
chapter3.3, Desktop, Documents, Downloads, ex030311, ex030401, examples.desktop,
Music, Pictures, Public, Templates, Videos
```

图 3-52 用“,”号区隔每个文件和目录的名称

(6) 执行以下命令可以列出当前工作目录下的所有文件和子目录。

```
ls -R
```

该命令执行后的效果如图 3-53 所示。

```
user01@ubuntu:~$ ls -R
.:
chapter03  chapter3.1  Documents  ex030402  Public
chapter030301  chapter3.2  Downloads  examples.desktop  Templates
chapter030302  chapter3.3  ex030311  Music  Videos
chapter030305  Desktop    ex030401  Pictures

./chapter03:

./chapter030301:

./chapter030302:

./chapter030305:
ex030305

./chapter3.1:

./chapter3.2:
```

图 3-53 列出当前工作目录下的所有文件和子目录

### 3.3.10 以树状图列出目录内容

使用 tree 命令可以列出目录内容,但以默认方式安装的 Ubuntu 系统并不包含这个命令,所以需要按以下步骤安装后才能使用。

(1) 使用 root 账户执行如下命令。

```
sudo apt - get install tree
```

该命令执行后,系统将自动进行安装,出现如图 3-54 所示的提示信息说明安装成功。

```

root@ubuntu:~# sudo apt-get install tree
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  tree
0 upgraded, 1 newly installed, 0 to remove and 167 not upgraded.
Need to get 40.6 kB of archives.
After this operation, 138 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu xenial/universe amd64 tree amd64 1.7.0-3 [40.6 kB]
Fetched 40.6 kB in 1s (37.2 kB/s)
Selecting previously unselected package tree.
(Reading database ... 174194 files and directories currently installed.)
Preparing to unpack .../tree_1.7.0-3_amd64.deb ...
Unpacking tree (1.7.0-3) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up tree (1.7.0-3) ...

```

图 3-54 安装 tree 命令

**注意：**执行该安装命令时若提示“dpkg was interrupted, you must manually run 'sudo dpkg --configure -a' to correct the problem”，则请按该提示在命令窗口执行“sudo dpkg --configure -a”，然后再重新执行上述安装命令“sudo apt-get install tree”。

(2) tree 命令安装成功后,当前用户默认为 root,需使用 login 命令切换至普通用户,并在当前工作目录中执行以下命令。

```
tree -l
```

该命令执行后的效果如图 3-55 所示。

(3) 执行以下命令可以查看所有文件(包括以“.”开头的隐藏文件)。

```
tree -a
```

该命令执行后的效果如图 3-56 所示。

```

user01@ubuntu:~$ tree -l
├── chapter03
│   ├── chapter030301
│   ├── chapter030302
│   ├── chapter3.1
│   ├── chapter3.2
│   └── chapter3.3
├── Desktop
├── Documents
├── Downloads
├── ex030305
├── ex030308
├── examples.desktop
├── Music
├── Pictures
├── Public
├── Templates
├── Videos
└── 14 directories, 3 files

```

图 3-55 生成目录树结构

```

user01@ubuntu:~$ tree -a
├── .bash_history
├── .bash_logout
├── .bashrc
├── .cache
├── compizconfig-1
│   ├── animation.pb
│   ├── commands.pb
│   ├── compiztoolbox.pb
│   ├── composite.pb
│   ├── copytex.pb
│   ├── core.pb
│   ├── decor.pb
│   ├── expo.pb
│   └── ezoom.pb

```

图 3-56 查看所有文件(包括以“.”开头的隐藏文件)

(4) 执行以下命令将不以阶梯状列出文件或目录名称。

```
tree -i
```

该命令执行后的效果如图 3-57 所示。

(5) 执行以下命令可以列出文件或目录大小。

```
tree -s
```

该命令执行后的效果如图 3-58 所示。

(6) 执行以下命令可以按文件和目录的更改时间排序。

```
tree -t
```

该命令执行后的效果如图 3-59 所示。

```

user01@ubuntu:~$ tree -l
.
├── chapter03
│   ├── chapter030301
│   ├── chapter030302
│   ├── chapter030305
│   ├── ex030305
│   ├── chapter3.1
│   ├── chapter3.2
│   ├── chapter3.3
│   ├── Desktop
│   ├── Documents
│   ├── Downloads
│   ├── ex030311
│   ├── ex030401
│   ├── examples.desktop
│   ├── Music
│   ├── Pictures
│   ├── Public
│   ├── Templates
│   └── Videos
└── 15 directories, 4 files

```

图 3-57 不以阶梯状列出文件或目录名称

```

user01@ubuntu:~$ tree -s
.
├── [4096] chapter03
│   ├── [4096] chapter030301
│   ├── [4096] chapter030302
│   ├── [4096] chapter030305
│   └── [0] ex030305
├── [4096] chapter3.1
├── [4096] chapter3.2
├── [4096] chapter3.3
├── [4096] Desktop
├── [4096] Documents
├── [4096] Downloads
├── [0] ex030311
├── [8] ex030401
├── [8980] examples.desktop
├── [4096] Music
├── [4096] Pictures
├── [4096] Public
├── [4096] Templates
└── [4096] Videos
└── 15 directories, 4 files

```

图 3-58 列出文件或目录大小

```

user01@ubuntu:~$ tree -t
├── examples.desktop
├── Desktop
├── Documents
├── Downloads
├── Music
├── Pictures
├── Public
├── Templates
├── Videos
├── chapter3.1
├── chapter3.2
├── chapter3.3
├── chapter030301
├── chapter030302
├── chapter03
├── ex030311
├── chapter030305
│   └── ex030305
├── ex030401
└── 15 directories, 4 files

```

图 3-59 按文件和目录的更改时间排序

### 3.3.11 显示文件或文件系统的详细信息

如果想要查看文件或者文件系统的详细信息,可以使用 `file` 命令,它用于检验文件的类型。请读者按以下步骤来操作。

(1) 首先,执行 `touch` 命令,创建一个名为 `ex030311` 的文件。

```
touch ex030311
```

该命令执行后,文件创建成功。

(2) 然后,执行以下命令查看文件的信息。

```
file -b ex030311
```

该命令执行后的效果如图 3-60 所示,显示 `ex030311` 为空文件。

(3) 最后,在 Ubuntu 系统中,每一个文件都存有 3 组日期和时间,它们包括最后编辑时间(即使用 `ls -l` 命令时显示的日期和时间)、最近状态改变时间(包括对文件重命名)和最后访问时间,而用于显示文件详细信息的命令是 `stat`。

我们执行 `stat ex030311` 命令,得到如图 3-61 所示的结果。

```

user01@ubuntu:~$ touch ex030311
user01@ubuntu:~$ file -b ex030311
empty

```

图 3-60 显示文件类型

```

user01@ubuntu:~$ stat ex030311
  File: 'ex030311'
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 801h/2049d Inode: 540449      Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1001/  user01)   Gid: ( 1001/  user01)
Access: 2017-05-08 23:31:55.455935999 -0700
Modify: 2017-05-08 23:31:55.455935999 -0700
Change: 2017-05-08 23:31:55.455935999 -0700
 Birth: -

```

图 3-61 显示文件各种信息

通过 `stat` 命令,我们可以清楚地看到这个文件的详细信息,具体如表 3-2 所示。

表 3-2 文件的详细信息

名称	含 义	名称	含 义
File	文件名	IO Block	IO 模块
Size	文件大小	Device	设备
Blocks	分区的容量	Inode	索引节点

续表

名称	含 义	名称	含 义
Links	连接数	Access	最后一次访问的时间
Access	权限	Modify	最后一次修改数据的时间
Uid	用户身份唯一标识	Change	最后一次修改元数据的时间
Gid	用户组身份唯一标识	Birth	创建时间

## 3.4 文件内容显示

在字符界面下使用 Ubuntu 系统时,需要以各种方式显示文件内容,接下来将介绍这一方面的知识。

### 3.4.1 创建和显示文件

如果想在系统中创建和显示文件,可以使用 cat 命令。

(1) 使用 ls 命令查看当前工作目录,效果如图 3-62 所示。

ls

```
user01@ubuntu:~$ ls
chapter03      chapter030305  chapter3.3  Downloads  Music  Templates
chapter030301 chapter3.1     Desktop     ex030311  Pictures Videos
chapter030302 chapter3.2     Documents   examples.desktop  Public
```

图 3-62 查看当前工作目录下的所有文件

可以看到,当前工作目录下不存在名为 ex030401 的文件。

(2) 使用 cat 命令创建文件 ex030401,执行如下命令。

cat > ex030401

(3) 如图 3-63 所示,命令执行成功后,系统无任何提示,

此时用户可以自由地向 ex030401 文件内输入相应内容。

abc  
defg

```
user01@ubuntu:~$ cat > ex030401
abc
defg
```

图 3-63 使用 cat 命令创建并向文件内写入内容

输入完成后,用户按下 Ctrl+D 快捷键结束编辑并退出。

(4) 我们再次使用 ls 命令查看当前工作目录,发现已存在名为 ex030401 的文件,效果如图 3-64 所示。

(5) cat 命令还可以用于查看文件,执行 cat -n ex030401 命令,可对输出的内容以行为单位进行编号,效果如图 3-65 所示。

```
user01@ubuntu:~$ ls
chapter03      chapter3.1  Documents  examples.desktop  Templates
chapter030301 chapter3.2  Downloads  Music              Videos
chapter030302 chapter3.3  ex030311  Pictures
chapter030305 Desktop    ex030401  Public
```

图 3-64 查看当前工作目录下的所有文件

```
user01@ubuntu:~$ cat -n ex030401
1 abc
2 defg
```

图 3-65 cat 命令查看文件

**注意:** 如果当前目录下已存在 ex030401 这一文件,那么输入的新内容将自动覆盖原文件中的内容。

### 3.4.2 改变文件权限

Ubuntu 中 chmod 命令的语法为: chmod **【u/g/o/a】【+/-/=】【r/w/x】**。其中:

① u 表示用户(user,指文件或目录所有人);g 表示同组用户(group,与文件或目录所有人组 ID 相同的用户);o 表示其他用户(others);a 表示所有用户(all),包括 u、g 和 o。

② + 表示添加权限;- 表示移除权限;= 表示重置权限。

③ r 表示读取文件或目录的权限(read);w 表示写入文件或目录的权限(write),x 表示执行的权限(execute)。

通过使用 1(执行)、2(写入)和 4(读取)3 种数值及其任意形式组合来确定权限,其中 1 代表执行权限,2 代表写入权限,4 代表读取权限。例如,5(5=4+1)代表有读取和执行权限,6(6=4+2)代表有读取和写入权限,7(7=4+2+1)代表有读取、写入和执行权限。我们以文件所有人 u 的权限为例,解释数值与对应权限的关系,具体如表 3-3 所示。

表 3-3 数值及对应的权限

独立权限			组合权限		
数值	权限	备注	数值	权限	备注
0	无	无动作	3	wx	执行和写入
1	x	执行	5	rx	读取和执行
2	w	写入	6	rw	读取和写入
4	r	读取	7	rwX	读取、写入和执行

**注意:** 指定组 g 和其他用户 o 的权限与本表相同。

用户可以按以下步骤学习 chmod 命令的用法。

(1) 首先,使用 touch 创建一个名为 ex030402 的文件,并使用 ls -l 命令查看其权限,具体如下。

```
touch ex030402
ls -l ex030402
```

执行效果如图 3-66 所示。

可以看到,该文件对文件所有人和组用户具有读取和写入权限,而对其他用户只有读取权限。

(2) 然后,使用 chmod 命令为 ex030402 更改权限。执行如下命令。

```
chmod 777 ex030402
```

上述命令执行完毕后,所有人均具有对 ex030402 文件的读取、写入和执行权限。

此时,可以使用 ls -l ex030402 命令查看 ex030402 文件的权限,如图 3-67 所示。

```
user01@ubuntu:~$ touch ex030402
user01@ubuntu:~$ ls -l ex030402
-rw-rw-r-- 1 user01 user01 0 May 12 22:04 ex030402
```

图 3-66 创建文件并查看权限

```
user01@ubuntu:~$ chmod 777 ex030402
user01@ubuntu:~$ ls -l ex030402
-rwxrwxrwx 1 user01 user01 0 May 12 22:04 ex030402
```

图 3-67 更改并查看文件权限

**注意:** 执行命令 chmod a=rwx ex030402 和 chmod 777 ex030402 的效果相同,即表示所有人(a 表示 all)均对 ex030402 文件具有读取、写入和执行的权限。

### 3.4.3 分页往后显示文件

如果想分页显示文件内容,可以使用 `more` 命令。

(1) 首先,使用 `cat` 命令创建一个名为 `ex030403` 的文件,并向文件内写入以下数据,具体如下。

```
cat > ex030403
the
be
of
and
a
to
in
he
have
it
that
for
they
he
with
as
not
on
she
at
by
this
we
you
do
but
from
or
which
onewould
all
will
there
say
who
make
when
can
more
if
no
man
out
other
so
what
time
up
```

```

go
about
than
into
could
state
only
new
year
some
take
come

```

该命令执行后的效果如图 3-68 所示。

(2) 然后,使用 more 命令分页显示 ex030403 文件的内容,命令如下。

```
more ex030403
```

该命令执行后的效果如图 3-69 所示。

(3) 最后,当前文档的内容并没有被全部显示出来,而是如图 3-69 底部所示,只显示了文档的部分内容。这时我们想查看下一页的内容,可以按下 Space(空格)键进行翻页,效果如图 3-70 所示。

```

user01@ubuntu:~$ cat > ex030403
the
be
of
and
a
to
in
he
have
it
that
for
they
he i
with
as
not
on
she
at
by
this
we

```

图 3-68 创建并向文件内写入内容

```

user01@ubuntu:~$ more ex030403
the
be
of
and
a
to
in
he
have
it
that
for
they
he I
with
as
not
on
she
at
by
this
we
--More-- (44%)

```

图 3-69 使用 more 命令分页显示文件内容

```

with
as
not
on
she
at
by
this
we
you
do
but
from
or
which
onewould
all
will
there
say
who
make
when
can
more
if
no
man
out
other
so
what

```

图 3-70 使用 more 命令分页并翻页显示文件内容

### 3.4.4 分页自由显示文件

在 Ubuntu 系统中分页显示文件内容除了可以使用 more 命令之外还可以使用 less 命令,现介绍如下。

(1) 首先,执行以下命令创建 ex030404 文件,并在当前目录下执行 ls 命令查看文件是否被创建成功。

```

cp ex030403 ex030404
ls

```

该命令执行后的效果如图 3-71 所示。

```

user01@ubuntu:~$ cp ex030403 ex030404
user01@ubuntu:~$ ls
chapter03      chapter3.2  ex030311  ex030501  examples.desktop  Videos
chapter030301  chapter3.3  ex030401  ex030502  Music
chapter030302  Desktop    ex030402  ex03050401  Pictures
chapter030305  Documents  ex030403  ex03050402  Public
chapter3.1     Downloads  ex030404  ex03050403  Templates

```

图 3-71 复制 ex030403 文件并更名为 ex030404

(2) 然后,使用 less 命令进行分页显示,命令如下。

```
less ex030404
```

该命令执行后的效果如图 3-72 所示。

**注意:** 在使用 more 命令和 less 命令显示文件内容时,有以下区别。

- ① 使用 more 命令显示文件内容时,只能通过按下 Space(空格)键向后翻页。
- ② 使用 less 命令时,除了可以像使用 more 命令那样通过按下 Space(空格)键向后翻页,还可以通过按下 Page Up 键和 Page Down 键进行前后翻页。

再次执行 less 命令。

```
less ex030404
```

先按下 Page Down 键后,效果如图 3-73 所示。

再按下 Page Up 键后,效果如图 3-74 所示。

```

the
be
of
and
a
to
in
he
have
it
that
for
they
he I
with
as
not
on
she
at
by
this
we
you
do
but
from
or
which
onewould
all
will
:

```

图 3-72 使用 less 命令分页显示文件

```

with
as
not
on
she
at
by
this
we
you
do
but
from
or
which
onewould
all
will
there
say
who
make
when
can
more
if
no
man
out
other
so
what
(END)

```

图 3-73 按下 Page Down 键

```

the
be
of
and
a
to
in
he
have
it
that
for
they
he I
with
as
not
on
she
at
by
this
we
you
do
but
from
or
which
onewould
all
will
:

```

图 3-74 按下 Page Up 键

若被显示的文件太长,用户可以在任意时刻直接在键盘上按下 Q 键退出当前文件的显示。用 more 命令查看文件时也可以如此操作(即按 Q 键退出当前文件的显示)。

### 3.4.5 指定显示文件前若干行

若我们需要指定显示文件的前若干行,使用 more 命令则无法实现,而使用 less 命令时,操作较为烦琐。这时我们可以使用 head 命令来实现,具体操作如下。

(1) 使用 cp 命令复制 ex030403 文件并更名为 ex030405,再执行如下命令,查看文件 ex030405 开头的几行内容,该命令默认显示前 10 行内容。

```
head ex030405
```

该命令执行后的效果如图 3-75 所示。

(2) 如果想查看 ex030405 文件中更多的内容,可以通过一个数字选项来设置需要显示的行数,具体命令如下。

```
head -13 ex030405
```

通过以上命令可查看 ex030405 文件中前 13 行的内容,该命令执行后的效果如图 3-76 所示。

```
user01@ubuntu:~$ head ex030405
the
be
of
and
a
to
in
he
have
it
```

图 3-75 查看 ex030405 文件开头的几行内容

```
user01@ubuntu:~$ head -13 ex030405
the
be
of
and
a
to
in
he
have
it
that
for
they
```

图 3-76 查看 ex030405 文件中前 13 行的内容

### 3.4.6 指定显示文件后若干行

学习了如何使用 head 命令查看文件开头指定的若干行内容后,接下来我们将介绍查看文件末尾指定若干行内容的命令 tail。

(1) 首先,使用 cp 命令复制 ex030403 文件并更名为 ex030406,再执行如下命令,查看文件 ex030406 的最后几行内容,该命令默认是后 10 行的内容。

```
tail ex030406
```

该命令执行后的效果如图 3-77 所示。

(2) 其次,如果想查看 ex030406 文件后面更多内容,可通过设置一个数字选项来更改所需显示的行数,具体命令如下。

```
tail -13 ex030406
```

通过执行上面的命令,可以查看 ex030406 文件后 13 行的内容,其效果如图 3-78 所示。

```
ubuntu@summer-virtual-machine:~$ tail ex030406
than
into
could
state
only
new
year
some
take
come
```

图 3-77 查看 ex030406 文件的最后几行内容

```
ubuntu@summer-virtual-machine:~$ tail -13 ex030406
up
go
about
than
into
could
state
only
new
year
some
take
come
```

图 3-78 查看 ex030406 文件后 13 行的内容

## 3.5 文件内容处理

我们已经学习了显示文件内容的一系列命令,接下来继续学习如何处理文件内容。例如,对文件内容进行排序,在文件中查找指定内容,对文件内容进行剪切、粘贴和统计等。其

中,某些命令更高级的用法将会在后续章节中介绍。

### 3.5.1 对文件内容进行排序

在 Ubuntu 系统中,可以使用 `sort` 命令对文件内容进行排序,它可以将排序结果显示出来。请按以下步骤进行操作。

(1) 使用 `cat` 命令创建一个名为 `ex030501` 的文件并写入数据,具体如下。

```
cat > ex030501
1
4
3
5
2
```

该命令执行后的效果如图 3-79 所示。

(2) 文件创建成功之后,我们可以使用 `sort` 命令对此文件的内容进行排序并显示,具体如下。

```
sort ex030501
```

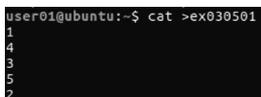
该命令执行后的效果如图 3-80 所示。

从图 3-80 可以看到,原来是乱序的文件 `ex030501` 已被成功排序。

(3) 如果想改成以逆序(与已被排顺序相反)的形式显示,那么只需加上“-r”参数即可,命令如下。

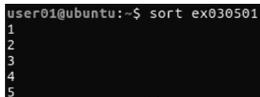
```
sort -r ex030501
```

效果如图 3-81 所示。



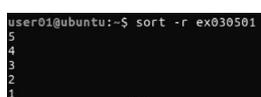
```
user01@ubuntu:~$ cat >ex030501
1
4
3
5
2
```

图 3-79 创建 `ex030501` 文件并写入数据



```
user01@ubuntu:~$ sort ex030501
1
2
3
4
5
```

图 3-80 将文件的内容排序后显示出来



```
user01@ubuntu:~$ sort -r ex030501
5
4
3
2
1
```

图 3-81 将文件的内容以逆序形式显示

### 3.5.2 检查文件中重复内容

`uniq` 命令用于检查文件中的重复内容,它可以报告或删除文件中重复的行。

(1) 首先,使用 `cat` 命令创建一个名为 `ex030502` 的文件并写入数据,具体如下。

```
cat > ex030502
a
a
b
b
```

该命令执行后的效果如图 3-82 所示。

如图所示,文件内有 4 行内容,其中第 2 行与第 1 行内容相同,第 4 行与第 3 行相同。

(2) 然后,执行 `uniq` 命令,具体如下。

```
uniq ex030502
```

该命令执行后的效果如图 3-83 所示。

```
user01@ubuntu:~$ cat >ex030502
a
a
b
b
```

图 3-82 创建 ex030502 文件并写入数据

```
user01@ubuntu:~$ uniq ex030502
a
b
```

图 3-83 使用 uniq 命令显示去除重复行后的文件内容

可以看到,仅输出了 2 行,其中第 1 行为“a”,第 2 行为“b”,而原文件中共有 4 行,其中前两行为“a”,后两行为“b”。这也就是说,使用 uniq 命令显示文件内容时,重复的内容仅显示一次。

### 3.5.3 在文件中查找指定内容

用户若需在文件中搜索某一关键字,可通过 grep 命令实现。具体如下。

(1) 我们复制 ex030502 文件并重命名为 ex030503,若想使用 grep 命令显示该文件中所有包含字母“a”的行,则命令具体如下。

```
grep 'a' ex030503
```

该命令执行后的效果如图 3-84 所示。

可以看到,上述命令执行成功后,grep 命令筛选出 ex030502 文件中所有包含字母“a”的行并显示出来。

(2) 如果只想显示 ex030503 文件中包含字母“a”的行数,只需执行以下命令即可。

```
grep -c 'a' ex030503
```

该命令执行后的效果如图 3-85 所示。

```
user01@ubuntu:~$ grep 'a' ex030503
a
```

图 3-84 显示 ex030503 文件中所有包含“a”的行

```
user01@ubuntu:~$ grep -c 'a' ex030503
2
```

图 3-85 显示包含字母“a”的行数

### 3.5.4 逐行对不同文件进行比较

在 Ubuntu 系统中,对不同文件进行比较的命令有 diff、diff3 和 sdiff 等。

(1) 使用 cat 命令创建 3 个文件,分别命名为 ex030504\_01、ex030504\_02 和 ex030504\_03,执行的具体情况如图 3-86 所示。

(2) 如果想对 2 个文件进行比较,那么可以使用 diff 命令,请输入如下命令。

```
diff ex030504_01 ex030504_02
```

该命令执行后的效果如图 3-87 所示。

```
user01@ubuntu:~$ cat > ex030504_01
a
b
c
user01@ubuntu:~$ cat > ex030504_02
a
d
c
user01@ubuntu:~$ cat > ex030504_03
d
b
c
```

图 3-86 创建 3 个文件并分别写入内容

```
user01@ubuntu:~$ diff ex030504_01 ex030504_02
2c2
< b
---
> d
```

图 3-87 使用 diff 命令对 2 个文件进行比较

我们可以看到上述结果可分为 2 部分：第 1 部分为“2c2”，它的含义是指第 1 个文件中的第 2 行需要做出修改才能与第 2 个文件中的第 2 行内容相同；剩余部分均为第 2 部分，它告诉我们 2 个文件的不同之处：

- ① “<b”表示左边文件的第 2 行内容为“b”；
- ② “>d”表示右边文件的第 2 行内容为“d”；
- ③ “--”则是 2 个文件内容的分隔符号。

(3) 如果想对 3 个文件进行比较,那么我们可以使用 diff3 命令,具体如下。

```
diff3 ex030504_01 ex030504_02 ex030504_03
```

该命令执行后的效果如图 3-88 所示。

我们可以看到上述结果可分为 3 个部分：第 1 部分为“1:1,2c”，它的含义是第 1 个文件中的第 1 行和第 2 行内容与其他 2 个文件不匹配,分别为“a”和“b”；第 2 部分为“2:1,2c”，它的含义是第 2 个文件中的第 1 行和第 2 行与其他 2 个文件不匹配,分别为“a”和“d”；第 3 部分为“3:1,2c”，它的含义是第 3 个文件中的第 1 行和第 2 行与其他 2 个文件不匹配,分别为“d”和“b”。这 3 部分内容中的字符“c”表示需要修改的意思。

(4) sdiff 命令的主要作用是合并 2 个文件,并以交互方式输出结果,请执行如下命令。

```
sdiff ex030504_01 ex030504_02
```

该命令执行后的效果如图 3-89 所示。

```
user01@ubuntu:~$ diff3 ex030504_01 ex030504_02 ex030504_03
====
1:1,2c
a
b
2:1,2c
a
d
3:1,2c
d
b
```

图 3-88 使用 diff3 命令对 3 个文件进行比较

```
user01@ubuntu:~$ sdiff ex030504_01 ex030504_02
a
b
c
```

图 3-89 使用 sdiff 命令对 2 个文件进行比较

可以看到,sdiff 命令将 2 个文件全部显示出来,左侧是 ex030504\_01 的内容,右侧是 ex030504\_02 的内容。其中,第 2 行有“|”符号标识,这是因为 2 个文件在此处内容不同。

### 3.5.5 逐字节对不同文件进行比较

如果需要逐字节对不同文件进行比较,那么可以使用 cmp 命令。

(1) 使用 cp 命令复制上一节创建的文件 ex030504\_01、ex030504\_02 并分别重命名为 ex030505\_01、ex030505\_02,具体如下。

```
cp ex030504_01 ex030505_01
cp ex030504_02 ex030505_02
```

(2) 然后使用 cmp 命令对这两个文件进行比较,命令如下。

```
cmp ex030505_01 ex030505_02
```

该命令执行后的结果如图 3-90 所示。

图中提示“ex030505\_01 ex030505\_02 differ: byte3,line2”表示 2 个文件的第 2 行内容有差异。

**注意：**使用 cmp 命令时,如果 2 个文件相同,比较完成后不显示任何提示;如果文件不

同,则将 2 个文件在逐字节比较过程中第 1 次出现不同时的行号及内容显示出来;如果想列出 2 个文件所有不一样的地方,那么只需加上“-l”参数即可,命令如下。

```
cmp -l ex030505_01 ex030505_01
```

该命令执行后的效果如图 3-91 所示。

```
user01@ubuntu:~$ cmp ex030505_01 ex030505_02
ex030505_01 ex030505_02 differ: byte 3, line 2
```

```
user01@ubuntu:~$ cmp -l ex030505_01 ex030505_02
3 142 144
```

图 3-90 使用 cmp 命令逐字节对不同文件进行比较 图 3-91 使用 cmp 命令标示出所有不一样的地方

图中第 1 个 3 的含义是 2 个文件的第 3 个字节有区别。在 ex030505\_01 中是八进制 142(即字符 b),在 ex030505\_02 中是八进制 144(即字符 d)。

### 3.5.6 对有序文件进行比较

如果需要对 2 个有序文件进行比较,那么可以使用 comm 命令。

(1) 首先,使用 cat 命令分别创建 ex030506\_01 和 ex030506\_02 文件,并输入内容,具体如下。

```
cat > ex030506_01
aaa
bbb
ccc
ddd
```

按下 Ctrl+D 键结束编辑,继续创建 ex030506\_02 文件。

```
cat > ex030506_02
bbb
ccc
eee
```

输入完毕后按下 Ctrl+D 键结束编辑,效果如图 3-92 所示。

(2) 然后,使用 comm 命令对 2 个文件进行比较,命令如下。

```
comm ex030506_01 ex030506_02
```

该命令执行后的效果如图 3-93 所示。

```
user01@ubuntu:~$ cat > ex030506_01
aaa
bbb
ccc
ddd
user01@ubuntu:~$ cat > ex030506_02
bbb
ccc
eee
```

图 3-92 创建 ex030506\_01 和 ex030506\_02 文件

```
user01@ubuntu:~$ comm ex030506_01 ex030506_02
aaa
bbb      bbb
ccc      ccc
ddd
eee
```

图 3-93 使用 comm 命令对 ex030506\_01 和 ex030506\_02 文件进行比较

可以看到,命令执行后的结果被分为了 3 列。第 1 列只显示在 ex030506\_01 文件中出现的行,第 2 列只显示在 ex030506\_02 文件中出现的行,第 3 列显示 ex030506\_01 和 ex030506\_02 文件中相同的行。

请执行以下复制文件的命令。

```
cp ex030502 ex030506_03
```

```
cp ex030501 ex030506_04
```

上述文件复制完毕后,再执行以下命令。

```
comm ex030506_03 ex030506_04
```

执行该命令后的效果如图 3-94 所示。

**注意:** comm 命令只适合对 2 个有序的文件进行比较,若待比较的文件无序,则会给出相应提示;如图 3-94 所示,ex030506\_04 文件无序,因此提示“comm:file 2 is not in sorted order”。

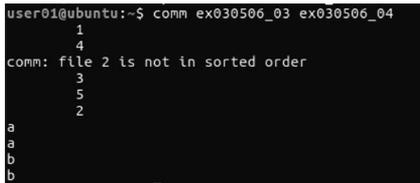


图 3-94 使用 comm 命令对无序文件进行比较

### 3.5.7 对文件内容进行剪切

cut 是 Ubuntu 下常用的命令之一,它负责剪切数据,其用法如下。

(1) 首先,使用 cat 创建一个名为 ex030507 的文件,该文件的内容如下。

```
cat > ex030507
abc
abc
abc
```

(2) 然后,使用以下命令截取文件中每行的第 2 个字符,命令如下。

```
cut -b 2 ex030507
```

该命令执行后的效果如图 3-95 所示。

从图中可以看到,ex030507 文件中的每一行的第 2 个字符“b”被筛选出来。

(3) 接着,使用以下命令截取每行的第 3 个字符,命令如下。

```
cut -c 3 ex030507
```

该命令执行后的效果如图 3-96 所示。

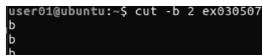


图 3-95 截取每行的第 2 个字符

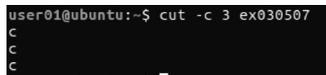


图 3-96 截取每行的第 3 个字符

**注意:** 从上面我们可以看到,似乎 cut -b 与 cut -c 并没有什么区别,其实不然,cut -b 命令在处理中文文档时会出现乱码,而 cut -c 命令则可以正常输出。

(4) 最后,使用下述命令截取以字母“b”为分割的第 2 个域,命令如下。

```
cut -d 'b' -f 2 ex030507
```

该命令执行后的效果如图 3-97 所示。

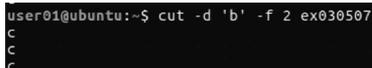


图 3-97 截取以字母“b”为分割的第 2 个域

**注意:** -d 参数用于以字符“b”分割,-f 参数用于截取以字母“b”为分割的第 2 个域。

### 3.5.8 对文件内容进行粘贴

若需对多个文件的内容进行合并,可以使用 paste 命令。

(1) 首先,使用 cat 命令分别创建 ex030508\_01 和 ex030508\_02 文件,具体如下。

```
cat > ex030508_01
abc
123
cat > ex030508_01
def
456
```

该命令执行后的效果如图 3-98 所示。

(2) 然后,使用 paste 命令将 ex030508\_02 文件的内容粘贴到 ex030508\_01 文件中去,请执行以下命令。

```
paste ex030508_01 ex030508_02
```

该命令执行后的效果如图 3-99 所示。

```
user01@ubuntu:~$ cat > ex030508_01
abc
123
user01@ubuntu:~$ cat > ex030508_02
def
456
```

图 3-98 创建 ex030508\_01 和 ex030508\_02 文件并输入内容

```
user01@ubuntu:~$ paste ex030508_01 ex030508_02
abc def
123 456
```

图 3-99 使用 paste 命令将 ex030508\_02 的内容粘贴到 ex030508\_01 中

### 3.5.9 对文件内容进行统计

Ubuntu 系统中 wc 命令的功能为统计指定文件中的字符数、单词数和行数,并将统计结果显示输出。

(1) 首先,执行以下命令创建 ex030509 文件。

```
cp ex030508_01 ex030509
```

然后,在当前目录执行 ls 命令查看文件是否创建成功。

```
ls
```

该命令执行后的效果如图 3-100 所示。

```
user01@ubuntu:~$ cp ex030508_01 ex030509
user01@ubuntu:~$ ls
chapter03      ex030311      ex03050402    ex030506_03    ex030802~
chapter030301 ex030401      ex030504_02   ex030506_04    ex030803.bz2
chapter030302 ex030402      ex03050403    ex030507        ex030804
chapter030305 ex030403      ex030504_03   ex03050801     ex030805
chapter3.1     ex030404      ex03050501    ex030508_01    examples.desktop
chapter3.2     ex030405      ex030505_01   ex03050802     Music
chapter3.3     ex030406      ex03050502    ex030508_02    Pictures
Desktop        ex030501      ex030505_02   ex030509        Public
Documents      ex030502      ex03050601    ex030602        Templates
Downloads      ex030503      ex030506_01   ex030801        Videos
ex030306      ex03050401    ex03050602    ex030801.zip
ex030306_01   ex030504_01   ex030506_02   ex030802
```

图 3-100 创建 ex030509 文件

(2) 使用 wc 命令统计文件中的字符数、单词数和行数,命令如下。

```
wc ex030509
```

该命令执行后的效果如图 3-101 所示。

其中,前 3 个数字分别表示 ex030509 文件的行数、单词数,以及该文件的字符数。可以看到,ex030509 文件中的内容共有 2 行,包括 2 个单词及 8 个字符。

(3) 如果只想查看文件中的字符数,执行如下命令即可。

```
wc -c ex030509
```

该命令执行后的效果如图 3-102 所示。

```
user01@ubuntu:~$ wc ex030509
2 2 8 ex030509
```

图 3-101 统计文件中的字符数、单词数和行数

```
user01@ubuntu:~$ wc -c ex030509
8 ex030509
```

图 3-102 统计文件中的字符数

(4) 如果只想查看文件中单词数,执行如下命令即可。

```
wc -w ex030509
```

该命令执行后的效果如图 3-103 所示。

(5) 如果只想查看文件中行数,执行如下命令即可。

```
wc -l ex030509
```

该命令执行后的效果如图 3-104 所示。

```
user01@ubuntu:~$ wc -w ex030509
2 ex030509
```

图 3-103 统计文件中的单词数

```
user01@ubuntu:~$ wc -l ex030509
2 ex030509
```

图 3-104 统计文件中的行数

## 3.6 文件查找

本节将学习文件查找的相关知识。Ubuntu 系统由成千上万个文件组成,我们在日常使用 Ubuntu 时必须熟练掌握文件查找的功能,包括在硬盘上和数据库中查找文件或目录及其位置等。

### 3.6.1 在硬盘上查找文件或目录

如果想要在本地硬盘中搜索文件或者目录,可以使用 `find` 命令完成这一操作。

(1) 使用模糊查询搜索当前工作目录下所有以 `ex` 开头的文件,命令如下。

```
find -name "ex *"
```

该命令执行后的效果如图 3-105 所示。

可以看到,当前工作目录下所有以 `ex` 开头的文件都被列了出来。

(2) 使用 `find` 命令搜索一天之内被存取过的文件,命令如下。

```
find -atime -1
```

该命令执行后的效果如图 3-106 所示。

命令执行完成后,一天之内被存取过的文件都被列了出来,其中大部分都是系统文件。

(3) 也可以使用 `find` 命令查找系统中为空的文件或者文件夹,命令如下。

```
find / -empty
```

该命令执行后的效果如图 3-107 所示。

(4) 还可以根据需要查找系统中属于用户 `user01` 的文件,命令如下。

```
find / -user user01
```

该命令执行后的效果如图 3-108 所示。

```

user01@ubuntu:~$ find -name "ex*"
./ex030402
./ex030406
./chapter030305/ex030305
./ex03030601
./ex030311
./examples.desktop
./ex03050501
./ex030403
./ex03050403
./ex03050802
./ex030509
./ex030503
./ex030306
./ex030404
./ex030502
./ex030401
./ex03050402
./ex03050401
./ex03050601
./ex03050801
./ex030501

```

图 3-105 搜索当前工作目录下所有以 ex 开头的文件

```

user01@ubuntu:~$ find -atime -1
./xsession-errors
./Pictures
./chapter030305
./profile
./Videos
./chapter3.2
./ex03050802
./ex030509
./chapter3.3
./compiz
./compiz/session
./chapter3.1
./chapter030302
./bash_history
./oracle_jre_usage
./Downloads
./Xauthority
./local
./local/share
./local/share/applications
./local/share/gvfs-metadata

```

图 3-106 搜索一天之内被存取过的文件

```

/usr/src/linux-headers-4.8.0-36-generic/include/config/afe4404.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lb700/wdt.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/max30100.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/mmu.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/pantherlord/ff.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/slimware/wml.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/pptp.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/bpf/events.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/bpf/jit.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/bpf/syscall.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lowpan/nhc.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lowpan/nhc/dest.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lowpan/nhc/ipv6.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lowpan/nhc/udp.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lowpan/nhc/routing.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lowpan/nhc/fragment.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lowpan/nhc/hop.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lowpan/nhc/mobility.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/pccard.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/vt6055.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/lbnasr.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/t8712f/wdt.h
/usr/src/linux-headers-4.8.0-36-generic/include/config/ad7476.h

```

图 3-107 查找系统中为空的文件或者文件夹

```

/proc/1972/task/1980/attr/prev
/proc/1972/task/1980/attr/exec
/proc/1972/task/1980/attr/fscreate
/proc/1972/task/1980/attr/keycreate
/proc/1972/task/1980/attr/socketcreate
/proc/1972/task/1980/wchan
/proc/1972/task/1980/stack
/proc/1972/task/1980/schedstat
/proc/1972/task/1980/cpuset
/proc/1972/task/1980/cgroup
/proc/1972/task/1980/oom_score
/proc/1972/task/1980/oom_adj
/proc/1972/task/1980/oom_score_adj
/proc/1972/task/1980/loginuid
/proc/1972/task/1980/sessionid
/proc/1972/task/1980/to
/proc/1972/task/1980/utd_map
/proc/1972/task/1980/gid_map
/proc/1972/task/1980/projid_map
/proc/1972/task/1980/setgroups
/proc/1972/task/1982
/proc/1972/task/1982/fd
/proc/1972/task/1982/fd/0
/proc/1972/task/1982/fd/1

```

图 3-108 查找系统中属于用户 user01 的文件

### 3.6.2 在数据库中查找文件或目录

locate 命令用于查找文件,它比 find 命令搜索速度快,让使用者可以更加快速地搜寻指定文件。

```

user01@ubuntu:~$ locate /user01/ex
/home/user01/ex030306
/home/user01/ex03030601
/home/user01/ex030311
/home/user01/ex030401
/home/user01/ex030402
/home/user01/ex030403
/home/user01/ex030404
/home/user01/ex030405
/home/user01/ex030406
/home/user01/ex030501
/home/user01/ex030502
/home/user01/ex030503
/home/user01/ex03050401
/home/user01/ex03050402
/home/user01/ex03050403
/home/user01/ex03050501
/home/user01/ex03050502
/home/user01/ex03050601
/home/user01/ex03050602
/home/user01/ex030507
/home/user01/ex03050801
/home/user01/ex03050802
/home/user01/ex030509
/home/user01/examples.desktop

```

图 3-109 使用 locate 命令查找 user01 目录下所有以 ex 开头的文件

(1) 首先,使用 locate 命令查找 user01 目录下所有以 ex 开头的文件,命令如下。

```
locate /user01/ex
```

该命令执行的效果如图 3-109 所示。

**注意:** locate 命令在后台数据库中按文件名搜索,搜索速度比 find 更快,但对于刚建立的文件,立即使用该命令进行查找会搜索不到所创建的文件;

若执行 locate 命令时出现“Command ‘locate’ not found, but can be installed with: sudo apt install mlocate”提示,请按提示安装。

接下来,我们使用 touch 命令创建 ex030602 文件

并使用 locate 命令查找,效果如图 3-110 所示。

如果想使刚创建的文件被 locate 命令立即搜索到,必须以 root 用户登录执行 updatedb 命令,更新数据库。更新完毕之后再使用 locate 命令就能搜索到刚创建的文件,否则要等到第二天才能搜索到该文件,因为后台数据库默认一天更新一次。

该命令执行后的效果如图 3-111 所示。

```
hul199@hul199-virtual-machine:~$ touch ex030602
hul199@hul199-virtual-machine:~$ ls
公共的 文档 桌面 下载 ex030602 模板 音乐 ex0506_02_tr
图片 桌面 snap
hul199@hul199-virtual-machine:~$ locate ex030602
hul199@hul199-virtual-machine:~$
```

图 3-110 使用 locate 命令搜索刚刚创建的文件

```
root@ubuntu:/home/user01# updatedb
root@ubuntu:/home/user01# locate ex030602
/home/user01/ex030602
```

图 3-111 使用 updatedb 命令更新数据库后搜索刚刚创建的文件

可以看到执行完 updatedb 命令后,刚刚创建的 ex030602 文件已经被成功查找了出来。

(2) 然后,使用 locate -r 命令搜索所有以 02 结尾的文件,命令如下。

```
locate -r 02 $
```

该命令执行后的效果如图 3-112 所示。

```
user01@ubuntu:~$ locate -r 02$
/home/user02
/home/user01/chapter030302
/home/user01/ex030402
/home/user01/ex030502
/home/user01/ex03050402
/home/user01/ex03050502
/home/user01/ex03050602
/home/user01/ex03050802
/home/user01/ex030602
/jdk/jdk1.8.0_131/lib/missioncontrol/features/org.eclipse.babel.nls_eclipse_ja_4.4.0.v20140623020002
/jdk/jdk1.8.0_131/lib/missioncontrol/features/org.eclipse.babel.nls_eclipse_zh_4.4.0.v20140623020002
/lib/modules/4.8.0-36-generic/kernel/drivers/media/usb/as102
/lib/modules/4.8.0-36-generic/kernel/drivers/media/usb/gspca/m5602
/lib/modules/4.8.0-36-generic/kernel/net/802
/lib/modules/4.8.0-49-generic/kernel/drivers/media/usb/as102
/lib/modules/4.8.0-49-generic/kernel/drivers/media/usb/gspca/m5602
/lib/modules/4.8.0-49-generic/kernel/net/802
```

图 3-112 搜索所有以 02 结尾的文件

### 3.6.3 查找指定文件或目录的位置

查找文件的命令除了 find 和 locate 之外,还有 whereis 命令,该命令会在特定目录中查找符合条件的文件。这些文件只能是源代码、二进制文件,或是帮助文件。

使用 whereis 命令查看命令 ls 的位置,命令如下。

```
whereis ls
```

该命令执行后的效果如图 3-113 所示。

```
user01@ubuntu:~$ whereis ls
ls: /bin/ls /usr/share/man/man1/ls.1.gz
```

图 3-113 使用 whereis 命令查看命令 ls 的位置

### 3.6.4 查找可执行文件的位置

which 命令也用于查找文件,该命令会在环境变量 \$PATH 设置的目录里查找符合条件的文件。

使用 which 命令查看命令 pwd 的路径,命令如下。

```
which pwd
```

该命令执行后的效果如图 3-114 所示。

```
user01@ubuntu:~$ which pwd
/bin/pwd
```

图 3-114 使用 which 命令查看命令 pwd 的路径

## 3.7 磁盘管理

本节将学习在 Ubuntu 中如何进行磁盘管理,包括检查磁盘空间、挂载和卸载文件系统、显示和分配磁盘空间等。

### 3.7.1 检查磁盘空间占用情况

检查磁盘空间占用情况的命令是 df,它还可以显示文件系统的类型等信息。

(1) 首先,使用 df -h 命令显示磁盘空间,命令如下。

```
df -h
```

该命令执行后的效果如图 3-115 所示。

```
user01@ubuntu:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            466M   0 466M   0% /dev
tmpfs           98M   6.2M  92M   7% /run
/dev/sda1       19G   5.7G  13G  32% /
tmpfs           487M  212K  487M   1% /dev/shm
tmpfs           5.0M   4.0K   5.0M   1% /run/lock
tmpfs           487M   0 487M   0% /sys/fs/cgroup
tmpfs           98M   68K   98M   1% /run/user/1001
/dev/sr0        49M   49M   0 100% /media/user01/CDROM
/dev/sr1        1.5G  1.5G   0 100% /media/user01/Ubuntu 16.04.2 LTS amd64
```

图 3-115 使用 df -h 命令显示磁盘空间

(2) 然后,使用 df -T 命令列出文件系统的类型,命令如下。

```
df -T
```

该命令执行后的效果如图 3-116 所示。

```
user01@ubuntu:~$ df -T
Filesystem      Type      1k-blocks  Used Available Use% Mounted on
udev            devtmpfs  477800    0    477800   0% /dev
tmpfs           tmpfs     99636    6288  93348   7% /run
/dev/sda1       ext4      19478204 5876312 12589412 32% /
tmpfs           tmpfs     498172    212  497960   1% /dev/shm
tmpfs           tmpfs     5120     4    5116   1% /run/lock
tmpfs           tmpfs     498172    0  498172   0% /sys/fs/cgroup
tmpfs           tmpfs     99636     68  99568   1% /run/user/1001
/dev/sr0        iso9660   49772    49772   0 100% /media/user01/CDROM
/dev/sr1        iso9660   1517760 1517760   0 100% /media/user01/Ubuntu 16.04.2 LTS amd64
```

图 3-116 使用 df -T 命令列出文件系统的类型

(3) 最后,可以使用 df -t 命令查看选定文件系统的磁盘信息,也可以使用 df -x 命令不显示选定的文件系统的磁盘信息,命令如下。

```
df -t ext4
```

```
df -x ext4
```

该命令执行后的效果如图 3-117 所示。

```
user01@ubuntu:~$ df -t ext4
Filesystem      1k-blocks  Used Available Use% Mounted on
/dev/sda1       19478204 5882048 12583676 32% /
user01@ubuntu:~$ df -x ext4
Filesystem      1k-blocks  Used Available Use% Mounted on
udev            477800    0    477800   0% /dev
tmpfs           99636    6340   93296   7% /run
tmpfs           498172    212  497960   1% /dev/shm
tmpfs           5120     4    5116   1% /run/lock
tmpfs           498172    0  498172   0% /sys/fs/cgroup
tmpfs           99636     44  99592   1% /run/user/1001
/dev/sr0        49772    49772   0 100% /media/user01/CDROM
/dev/sr1        1517760 1517760   0 100% /media/user01/Ubuntu 16.04.2 LTS amd64
```

图 3-117 查看选定文件系统的磁盘信息

### 3.7.2 统计目录或文件所占磁盘空间大小

在 Ubuntu 系统中查看目录或文件所占磁盘空间大小的命令是 du。

(1) 首先,使用 du -h 命令查看 chapter03 目录占用的磁盘空间大小,命令如下。

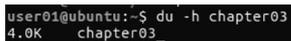
```
du -h chapter03
```

该命令执行后的效果如图 3-118 所示。

(2) 然后,使用 cd 命令切换到上一级目录并使用 du -a 命令查看 user01 目录及其子目录和文件占用磁盘空间的大小,命令如下。

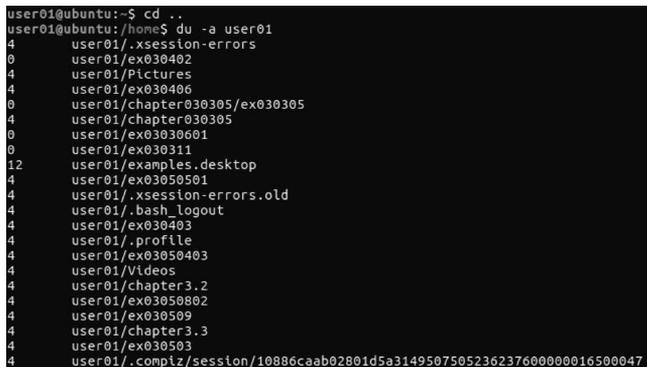
```
cd ..  
du -a user01
```

该命令执行后的效果如图 3-119 所示。



```
user01@ubuntu:~$ du -h chapter03  
4.0K chapter03
```

图 3-118 查看 chapter03 目录占用的磁盘空间大小



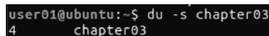
```
user01@ubuntu:~$ cd ..  
user01@ubuntu:~/home$ du -a user01  
4 user01/.xsession-errors  
0 user01/ex030402  
4 user01/Pictures  
4 user01/ex030406  
0 user01/chapter030305/ex030305  
4 user01/chapter030305  
0 user01/ex03030601  
0 user01/ex030311  
12 user01/examples_desktop  
4 user01/ex03050501  
4 user01/.xsession-errors.old  
4 user01/.bash_logout  
4 user01/ex030403  
4 user01/.profile  
4 user01/ex03050403  
4 user01/Videos  
4 user01/chapter3_2  
4 user01/ex03050802  
4 user01/ex030509  
4 user01/chapter3_3  
4 user01/ex030503  
4 user01/.compiz/session/10886caab02801d5a3149507505236237600000016500047
```

图 3-119 查看 user01 目录及其子目录和文件占用的磁盘空间大小

(3) 最后,使用 du -s 命令查看某一文件夹所占用的磁盘空间的大小,命令如下。

```
du -s chapter03
```

该命令执行后的效果如图 3-120 所示。



```
user01@ubuntu:~$ du -s chapter03  
4 chapter03
```

图 3-120 查看某一文件夹所占用的磁盘空间的大小

### 3.7.3 挂载文件系统

mount 命令用于将文件系统挂载到指定的挂载点上,本小节将介绍这一命令的基本用法,请按照以下步骤执行。

(1) 将 U 盘连接至系统后,按下 Ctrl+Alt+T 键打开终端,执行 df 命令可以看到系统显示其为/dev/sdb4。接下来执行以下命令对该 U 盘进行格式化。

```
mkfs.ext4 /dev/sdb4
```

(2) 上述命令将 sdb4 分区格式化为 ext4 文件系统,接下来创建一个目录作为其挂载点,并使用 mount 命令进行挂载,具体如下。

```
mkdir /mnt/quotadir
mount /dev/sdb4 /mnt/quotadir
mount | grep sdb4
```

可以看到,出现了如图 3-121 中显示的“Permission denied”的提示,这时只需切换至 root 用户并再次执行上述所有命令即可,效果如图 3-122 所示。

```
user01@xf-Dell-System-Inspiron-N4110:~$ mkdir /mnt/quotadir
mkdir: cannot create directory '/mnt/quotadir': Permission denied
```

图 3-121 创建文件夹时权限不足

```
root@xf-Dell-System-Inspiron-N4110:/home/user01# mkdir /mnt/quotadir
root@xf-Dell-System-Inspiron-N4110:/home/user01# mount /dev/sdb4 /mnt/quotadir
root@xf-Dell-System-Inspiron-N4110:/home/user01# mount | grep sdb4
/dev/sdb4 on /media/xf/a0fc22dd-51e6-40f8-bd58-77b3690c60f6 type ext4 (rw,nosuid
,nodev,relatime,data=ordered,uhelper=udisks2)
/dev/sdb4 on /mnt/quotadir type ext4 (rw,relatime,data=ordered)
root@xf-Dell-System-Inspiron-N4110:/home/user01#
```

图 3-122 挂载 U 盘至文件夹

(3) 为此目录进行配额的权限分配,命令如下。

```
mount -o remount,usrquota,grpquota /mnt/quotadir/
mount | grep sdb4
```

该命令执行后的效果如图 3-123 所示。

```
root@xf-Dell-System-Inspiron-N4110:/home/user01# mount -o remount,usrquota,grpquota /mnt/quotadir
root@xf-Dell-System-Inspiron-N4110:/home/user01# mount | grep sdb4
/dev/sdb4 on /media/xf/a0fc22dd-51e6-40f8-bd58-77b3690c60f6 type ext4 (rw,nosuid
,nodev,relatime,quota,usrquota,grpquota,data=ordered,uhelper=udisks2)
/dev/sdb4 on /mnt/quotadir type ext4 (rw,relatime,quota,usrquota,grpquota,data=ordered)
```

图 3-123 进行配额的权限分配

### 3.7.4 检查磁盘的使用空间与限制

因为 Ubuntu 是多用户多任务的操作系统,许多人共用磁盘空间,为了合理地分配磁盘空间,我们需要对其进行配额以便高效地使用磁盘空间。在 Ubuntu 中,通常使用 quota 命令来对磁盘进行配额,请按照以下步骤进行。

(1) 若尚未安装 quota,则须使用 root 用户下载安装 quota,命令如下。

```
apt install quota
```

该命令执行后的效果如图 3-124 所示。

```
root@ubuntu:/home/user01# apt install quota
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-4.8.0-36 linux-headers-4.8.0-36-generic
  linux-image-4.8.0-36-generic linux-image-extra-4.8.0-36-generic
Use 'apt autoremove' to remove them.
The following additional packages will be installed:
  libtirpc1
Suggested packages:
  libnet-ldap-perl rpcbind default-mta | mail-transport-agent
The following NEW packages will be installed:
  libtirpc1 quota
0 upgraded, 2 newly installed, 0 to remove and 98 not upgraded.
Need to get 325 kB of archives.
After this operation, 1,097 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 libtirpc1 amd64 0.2.5-1 [75.6 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 quota amd64 4.03-2 [250 kB]
Fetched 325 kB in 2s (136 kB/s)
```

图 3-124 安装 quota 命令

(2) 完成安装 quota 之后,在上一节的基础上使用 quotacheck 命令检查 U 盘空间配

置,命令如下。

```
cd /mnt/quotadir
ll
quotacheck -avugn
ll
```

该命令执行后的效果如图 3-125 所示。

```
root@xf-Dell-System-Inspiron-N4110:/# cd /mnt/quotadir
root@xf-Dell-System-Inspiron-N4110:/mnt/quotadir# ll
total 499624
drwxrwxrwx 3 xf xf 4896 Jun 8 15:02 /
drwxrwxrwx 3 root root 4896 Jun 4 13:58 /
-rw-rw-r-- 1 user01 user01 419430400 Jun 8 13:47 bigfile2
dwx----- 2 root root 16384 May 31 21:05 lost+found/
root@xf-Dell-System-Inspiron-N4110:/mnt/quotadir# quotacheck -avugn
quotacheck: Your kernel probably supports journaled quota but you are not using it. Consider switching to journaled quota to avoid running quotacheck after an unclean shutdown.
quotacheck: Scanning /dev/sdb4 [/media/xf/a0fc22dd-51e6-40f8-bd58-77b369bc60f6] done
quotacheck: Cannot stat old user quota file /media/xf/a0fc22dd-51e6-40f8-bd58-77b369bc60f6/aquota.user: No such file or directory. Usage will not be subtracted.
quotacheck: Cannot stat old group quota file /media/xf/a0fc22dd-51e6-40f8-bd58-77b369bc60f6/aquota.group: No such file or directory. Usage will not be subtracted.
quotacheck: Cannot stat old user quota file /media/xf/a0fc22dd-51e6-40f8-bd58-77b369bc60f6/aquota.user: No such file or directory. Usage will not be subtracted.
quotacheck: Cannot stat old group quota file /media/xf/a0fc22dd-51e6-40f8-bd58-77b369bc60f6/aquota.group: No such file or directory. Usage will not be subtracted.
quotacheck: Checked 2 directories and 1 files
quotacheck: Old file not found.
quotacheck: Old file not found.
root@xf-Dell-System-Inspiron-N4110:/mnt/quotadir# ll
total 499640
drwxrwxrwx 3 xf xf 4896 Jun 8 15:02 /
drwxrwxrwx 3 root root 4896 Jun 4 13:58 /
-rw-rw-r-- 1 root root 7168 Jun 8 15:02 aquota.group
-rw-rw-r-- 1 user01 user01 419430400 Jun 8 13:47 bigfile2
dwx----- 2 root root 16384 May 31 21:05 lost+found/
```

图 3-125 使用 quotacheck 扫描 U 盘并建立相应文件

注意: 上述命令成功执行后,在当前目录下会产生 aquota.user 和 aquota.group 两个文件。

### 3.7.5 打开磁盘空间限制

接下来需要启动 quota 服务以完成配额操作,命令如下。

```
quotaon -vug /mnt/quotadir/
```

该命令执行后的效果如图 3-126 所示。

```
root@xf-Dell-System-Inspiron-N4110:/home/user01# quotaon -vug /mnt/quotadir
/dev/sdb4 [/mnt/quotadir]: group quotas turned on
/dev/sdb4 [/mnt/quotadir]: user quotas turned on
```

图 3-126 打开磁盘空间限制

### 3.7.6 为指定用户分配限额

在上一节的基础上,可以使用 edquota 命令为用户或工作组指定磁盘配额。

(1) 使用 edquota 为 user01 用户指定限额,命令如下。

```
edquota -u user01
```

该命令执行后的效果如图 3-127 所示。

```
Disk quotas for user user01 (uid 1001):
Filesystem          blocks      soft      hard    inodes   soft$
/dev/sdb4            0            200000   300000  0         5
```

图 3-127 为 user01 用户指定限额

(2) 使用 `edquota` 为 `user01` 用户组指定限额, 命令如下。

```
edquota -g user01
```

该命令执行后的效果如图 3-128 所示。

```

Disk quotas for group user01 (gid 1001):
Filesystem      blocks    soft    hard    inodes   soft$
/dev/sdb4        0        200000 300000      0        0
  
```

图 3-128 为 `user01` 用户组指定限额

(3) 使用 `edquota` 命令为磁盘配额限定时间, 命令如下。

```
edquota -t
```

该命令执行后的效果如图 3-129 所示。

```

GNU nano 2.5.3      File: /tmp//EdP.apBUzUV
Grace period before enforcing soft limits for users:
Time units may be: days, hours, minutes, or seconds
Filesystem      Block grace period  Inode grace period
/dev/sdb4        7days              7days
  
```

图 3-129 为磁盘配额限定时间

### 3.7.7 显示用户或工作组的磁盘配额信息

在 Ubuntu 系统中显示磁盘已使用的空间与限制的命令是 `quota`, 它可以查询磁盘空间的限制, 并显示已使用多少空间。

(1) 首先, 使用 `quota` 命令显示用户 `user01` 的配额信息, 命令如下。

```
quota -vus user01
```

该命令执行后的效果如图 3-130 所示。

```

root@xf-Dell-System-Inspiron-N4110:/mnt/quotadir# quota -vus user01
Disk quotas for user user01 (uid 1001):
Filesystem  space  quota  limit  grace  files  quota  limit  grace
/dev/sdb4   400M* 196M   293M   7days    1      0      0
  
```

图 3-130 显示用户 `user01` 的配额信息

(2) 然后, 使用 `quota` 命令显示组 `user01` 的配额信息, 命令如下。

```
quota -vgs user01
```

该命令执行后的效果如图 3-131 所示。

```

root@xf-Dell-System-Inspiron-N4110:/mnt/quotadir# quota -vgs user01
Disk quotas for group user01 (gid 1001):
Filesystem  space  quota  limit  grace  files  quota  limit  grace
/dev/sdb4   400M* 196M   293M   7days    1      0      0
  
```

图 3-131 显示组 `user01` 的配额信息

### 3.7.8 显示文件系统的配额信息

在 Ubuntu 系统中,如果需要显示文件系统的配额信息,那么可以使用 `repquota` 命令。

(1) 首先,使用 `repquota` 命令查看 `/mnt/quotadir` 目录的磁盘空间限制,命令如下。

```
repquota -u /mnt/quotadir
```

该命令执行后的效果如图 3-132 所示。

```
root@xf-Dell-System-Inspiron-N4110:/home/user01# repquota -u /mnt/quotadir
*** Report for user quotas on device /dev/sdb4
Block grace time: 7days; Inode grace time: 7days
Block limits
User      used      soft      hard      grace    File limits
-----
root     --      16         0         0         1 0 0
xf       --       4         0         0         1 0 0
user01   +- 300000  200000  300000  none     1 0 0
```

图 3-132 查看 `/mnt/quotadir` 目录的磁盘空间限制

(2) 然后,继续使用 `repquota` 命令显示用户或群组的所有空间限制,命令如下。

```
repquota -v /mnt/quotadir
```

该命令执行后的效果如图 3-133 所示。

```
root@xf-Dell-System-Inspiron-N4110:/mnt/quotadir# quota -vgs user01
Disk quotas for group user01 (gid 1001):
Filesystem space quota limit grace files quota limit grace
/dev/sdb4 400M* 196M 293M 7days 1 0 0
```

图 3-133 显示用户或群组的所有空间限制

(3) 也可以使用 `repquota` 命令查看加入 `quota` 设置分区的磁盘使用状况,命令如下。

```
repquota -a
```

该命令执行后的效果如图 3-134 所示。

```
root@xf-Dell-System-Inspiron-N4110:/home/user01# repquota -a
*** Report for user quotas on device /dev/sdb4
Block grace time: 7days; Inode grace time: 7days
Block limits
User      used      soft      hard      grace    File limits
-----
root     -- 409616     0         0         2 0 0
xf       --       4         0         0         1 0 0
user01   +- 300000  200000  300000  none     1 0 0
```

图 3-134 查看所有文件系统的磁盘使用情况

(4) 最后,测试配额是否生效,使用 `chmod` 命令将 `mnt` 目录及 `quotadir` 目录的权限设为所有人可读写执行(777),然后切换至 `user01` 用户向该文件夹内写入大文件,命令如下。

```
dd if = /dev/zero of = bigfile bs = 400M count = 3
```

该命令执行后的效果如图 3-135 所示。

```
user01@xf-Dell-System-Inspiron-N4110:/mnt/quotadir$ dd if=/dev/zero of=bigfile b
s=400M count=3
dd: error writing 'bigfile': Disk quota exceeded
```

图 3-135 向文件夹内写入大文件

### 3.7.9 关闭磁盘空间限制

如果想要关闭指定文件系统的磁盘配额功能,可以使用 `quotaoff` 命令。

(1) 若只需关闭用户配额功能,则可以使用如下命令。

```
quotaoff -u /mnt/quotadir
```

(2) 若只需关闭组配额功能,则可以使用如下命令。

```
quotaoff -g /mnt/quotadir
```

(3) 若需显示命令执行过程,则可以使用如下命令。

```
quotaoff -v /mnt/quotadir
```

该命令执行后的效果如图 3-136 所示。

```
root@xf-Dell-System-Inspiron-N4110:/home/user01# quotaoff -v /mnt/quotadir
/dev/sdb4 [/mnt/quotadir]: group quotas turned off
/dev/sdb4 [/mnt/quotadir]: user quotas turned off
```

图 3-136 关闭所有配额功能

(4) 在关闭磁盘配额的前提下重新执行写入大文件的命令,效果如图 3-137 所示。

```
user01@xf-Dell-System-Inspiron-N4110:/mnt/quotadir$ dd if=/dev/zero of=bigfile2 bs=400M count=1
1+0 records in
1+0 records out
419430400 bytes (419 MB, 400 MiB) copied, 32.1031 s, 13.1 MB/s
```

图 3-137 向文件夹内写入大文件

从图中可以看到,在关闭磁盘空间配额的情况下,可以成功写入大文件。

### 3.7.10 卸载文件系统

在 Ubuntu 中,卸载文件系统的命令是 `umount`。

(1) 首先,通过设备名卸载文件系统,命令如下。

```
umount -v /dev/sdb4
```

该命令执行后的效果如图 3-138 所示。

```
root@xf-Dell-System-Inspiron-N4110:/home/user01# umount -v /dev/sdb4
umount: /mnt/quotadir (/dev/sdb4) unmounted
```

图 3-138 通过设备名卸载文件系统

(2) 其次,也可以通过挂载点卸载文件系统,命令如下。

```
umount -v /mnt/quotadir
```

该命令执行后的效果如图 3-139 所示。

```
root@xf-Dell-System-Inspiron-N4110:/home/user01# umount -v /mnt/quotadir
umount: /mnt/quotadir unmounted
```

图 3-139 通过挂载点卸载文件系统

## 3.8 备份压缩

同 Windows 一样,Ubuntu 系统中也有备份与压缩功能,接下来我们来学习这方面的内容,包括各个命令的使用方法。

### 3.8.1 zip、unzip 和 zipinfo

Ubuntu 系统中压缩命令有很多,如果想要产生具有“zip”扩展名的压缩文件,那么我们

就要使用 zip 命令。

(1) 执行 touch 命令创建一个名为 ex030801 的文件,然后执行以下命令将它进行压缩并另存为 ex030801.zip。

```
zip ex030801.zip ex030801
```

该命令执行后的效果如图 3-140 所示。

(2) 执行 zip -d 命令删除压缩包 ex030801.zip 中的 ex030801 文件,命令如下。

```
zip -d ex030801.zip ex030801
```

该命令执行后的效果如图 3-141 所示。

```
user01@ubuntu:~$ touch ex030801
user01@ubuntu:~$ zip ex030801.zip ex030801
adding: ex030801 (stored 0%)
```

图 3-140 压缩 ex030801 并另存

```
user01@ubuntu:~$ zip -d ex030801.zip ex030801
deleting: ex030801
zip warning: zip file empty
```

图 3-141 删除压缩包 ex030801.zip 中的 ex030801 文件

可以看到,系统提示当前压缩包为空,说明文件删除成功。

(3) 向压缩文件 ex030801.zip 中添加 ex030801 文件,命令如下。

```
zip -m ex030801.zip ex030801
```

该命令执行后的效果如图 3-142 所示。

(4) 如果我们已经有.zip 类型的压缩文件,该如何对它进行解压并获取其中的文件呢?在 Ubuntu 中对.zip 文件解压缩的命令是 unzip。执行如下命令将实现对 ex030801.zip 进行解压缩并覆盖原文件。

```
unzip -o ex030801.zip
```

该命令执行后的效果如图 3-143 所示。

```
user01@ubuntu:~$ zip -m ex030801.zip ex030801
adding: ex030801 (stored 0%)
```

图 3-142 向压缩文件 ex030801.zip 中添加 ex030801 文件

```
user01@ubuntu:~$ unzip ex030801.zip
Archive: ex030801.zip
extracting: ex030801
```

图 3-143 解压缩 ex030801.zip 并覆盖原文件

(5) 如果在解压缩过程中不想覆盖原文件,那么执行如下命令。

```
unzip -n ex030801.zip
```

该命令执行后的效果如图 3-144 所示。

(6) zipinfo 命令用来列出压缩文件信息。执行 zipinfo 命令可得知 zip 压缩文件的详细信息,具体如下。

```
zipinfo ex030801.zip
```

该命令执行后的效果如图 3-145 所示。

```
user01@ubuntu:~$ unzip -n ex030801.zip
Archive: ex030801.zip
```

图 3-144 解压缩 ex030801.zip 且不覆盖原文件

```
user01@ubuntu:~$ zipinfo ex030801.zip
Archive: ex030801.zip
Zip file size: 166 bytes, number of entries: 1
-rw-rw-r-- 3.0 unix      0 bX stor 17-Aug-07 00:06 ex030801
1 file, 0 bytes uncompressed, 0 bytes compressed: 0.0%
```

图 3-145 列出压缩文件信息

### 3.8.2 gzip、gunzip 和 gzexe

如果想要产生具有“.gz”扩展名的压缩文件,那么我们就需要使用 gzip 命令。

(1) 首先,把当前目录下的每个文件压缩成 .gz 文件,具体命令如下。

```
gzip *
```

该命令执行后的效果如图 3-146 所示。

(2) 压缩完成后,可执行以下命令完成对上例中每个压缩文件进行解压,并列出详细的信息,命令如下。

```
gzip -dv *
```

该命令执行后的效果如图 3-147 所示。

```
user01@ubuntu:~$ gzip *
gzip: chapter03 is a directory -- ignored
gzip: chapter030301 is a directory -- ignored
gzip: chapter030302 is a directory -- ignored
gzip: chapter030305 is a directory -- ignored
gzip: chapter3.1 is a directory -- ignored
gzip: chapter3.2 is a directory -- ignored
gzip: chapter3.3 is a directory -- ignored
gzip: Desktop is a directory -- ignored
gzip: Documents is a directory -- ignored
gzip: Downloads is a directory -- ignored
gzip: Music is a directory -- ignored
gzip: Pictures is a directory -- ignored
gzip: Public is a directory -- ignored
gzip: Templates is a directory -- ignored
gzip: Videos is a directory -- ignored
```

图 3-146 把当前目录下的每个文件压缩成 .gz 文件

```
user01@ubuntu:~$ gzip -dv *
gzip: chapter03 is a directory -- ignored
gzip: chapter030301 is a directory -- ignored
gzip: chapter030302 is a directory -- ignored
gzip: chapter030305 is a directory -- ignored
gzip: chapter3.1 is a directory -- ignored
gzip: chapter3.2 is a directory -- ignored
gzip: chapter3.3 is a directory -- ignored
gzip: Desktop is a directory -- ignored
gzip: Documents is a directory -- ignored
gzip: Downloads is a directory -- ignored
ex03030601.gz:  0.0% -- replaced with ex03030601
ex030306.gz:   0.0% -- replaced with ex030306
ex030311.gz:   0.0% -- replaced with ex030311
ex030401.gz:  -25.0% -- replaced with ex030401
ex030402.gz:   0.0% -- replaced with ex030402
ex030403.gz:  34.9% -- replaced with ex030403
ex030404.gz:  34.9% -- replaced with ex030404
ex030405.gz:  34.9% -- replaced with ex030405
ex030406.gz:  34.9% -- replaced with ex030406
ex030501.gz: -20.0% -- replaced with ex030501
ex030502.gz:   0.0% -- replaced with ex030502
ex030503.gz:   0.0% -- replaced with ex030503
```

图 3-147 把每个压缩的文件解压并列出详细信息

(3) 然后,使用 gunzip 命令对压缩包进行解压。假定存在压缩文件 ex030801.gz(若 ex030801.gz 不存在,可以使用命令 gzip ex030801 创建之;若 ex030801 不存在,可以使用命令 touch ex030801 创建之),可执行以下命令对其解压。

```
gunzip ex030801.gz
```

该命令执行后,ex030801.gz 被成功解压。我们也可以使用 gzip-d ex030801.gz 对该压缩包解压,读者可自行尝试。

(4) 最后,创建一个名为 ex030802 的文件并使用 gzexe 命令压缩,命令如下。

```
gzexe ex030802
```

该命令执行后的效果如图 3-148 所示。

如果想要解压缩该文件,那么我们只需加上 d 参数即可,如图 3-149 所示。

```
user01@ubuntu:~$ touch ex030802
user01@ubuntu:~$ gzexe ex030802
ex030802:  0.0%
```

图 3-148 创建一个名为 ex030802 的文件  
并使用 gzexe 命令压缩

```
user01@ubuntu:~$ gzexe -d ex030802
```

图 3-149 解压缩文件 ex030802

### 3.8.3 bzip2、bunzip2 和 bzip2recover

如果想要产生具有 .bz2 扩展名的压缩文件,那么就要使用 bzip2 命令。

(1) 首先,使用 touch 命令创建名为 ex030803 的文件,再使用 bzip2 命令进行压缩并查看,命令如下。

```
touch ex030803
bzip2 ex030803
ls
```

该命令执行后的效果如图 3-150 所示。

```
user01@ubuntu:~$ touch ex030803
user01@ubuntu:~$ bzip2 ex030803
user01@ubuntu:~$ ls
chapter03  Documents  ex030404  ex03050403  ex030509  Music
chapter030301  Downloads  ex030405  ex03050501  ex030602  Pictures
chapter030302  ex030306  ex030406  ex03050502  ex030801  Public
chapter030305  ex03030601  ex030501  ex03050601  ex030801.zip  Templates
chapter3_1  ex030311  ex030502  ex03050602  ex030802  Videos
chapter3_2  ex030401  ex030503  ex030507  ex030802~
chapter3_3  ex030402  ex03050401  ex03050801  ex030803.bz2
Desktop  ex030403  ex03050402  ex03050802  examples.desktop
```

图 3-150 创建、压缩并查看文件

上述命令执行时会先将文件 ex030803 删除,替换成 ex030803.bz2。如果以前有 ex030803.bz2 则不会替换并提示错误。

(2) 然后,进行解压操作,bzip2 和 bunzip2 命令都可完成该操作。执行以下命令。

```
bzip2 -d ex030803.bz2
```

该命令执行后的效果如图 3-151 所示。

也可执行以下命令完成解压操作。

```
bunzip2 ex030803.bz2
```

该命令执行后的效果如图 3-152 所示。

```
ubuntu@summer-virtual-machine:~$ bzip2 -d ex030803.bz2
```

图 3-151 使用 bzip2 命令进行解压

```
user01@ubuntu:~$ bunzip2 ex030803.bz2
```

图 3-152 使用 bunzip2 命令进行解压

**注意:** 如果在当前文件夹下没有文件 ex030803,那么这里解压的时候不会输出任何提示,而是直接将原来的文件 ex030803.bz2 替换成 ex030803;若在当前文件夹下存在 ex030803 这一文件,则不会被替换,而是给出相应提示。

(3) 如果由于压缩包损坏而无法读取,那么这时候可以使用 bzip2recover 命令修复压缩包,命令如下。

```
bzip2recover ex030803.bz2
```

该命令执行后的效果如图 3-153 所示。

```
user01@ubuntu:~$ bzip2recover ex030803.bz2
bzip2recover 1.0.6: extracts blocks from damaged .bz2 files.
bzip2recover: searching for block boundaries ...
bzip2recover: sorry, I couldn't find any block boundaries.
```

图 3-153 使用 bzip2recover 命令修复压缩包

可以看到,由于当前压缩包没有损坏,命令执行后提示用户无须修复。

### 3.8.4 compress 和 uncompress

如果想要产生扩展名为.Z 的压缩文件,那么就要使用 compress 命令。

(1) 使用 touch 命令创建名为 ex030804 的文件,再使用 compress 命令进行压缩,命令

如下。

```
compress ex030804
```

该命令执行后的效果如图 3-154 所示。

命令执行后,系统提示该命令未安装,这时我们只需切换至 root 用户并执行“apt install uncompress”命令即可,效果如图 3-155 所示。

```
ubuntu@summer-virtual-machine:~$ compress ex030804
Command 'compress' not found, but can be installed with:
sudo apt install ncompress
```

图 3-154 使用 compress 命令进行压缩

```
root@ubuntu:/home/user01# apt install ncompress
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-4.8.0-36 linux-headers-4.8.0-36-generic
  linux-image-4.8.0-36-generic linux-image-extra-4.8.0-36-generic
Use 'apt autoremove' to remove them.
The following NEW packages will be installed:
  ncompress
0 upgraded, 1 newly installed, 0 to remove and 98 not upgraded.
Need to get 19.8 kB of archives.
After this operation, 49.2 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu xenial/universe amd64 ncompress amd64
4.2.4.4-15 [19.8 kB]
Fetched 19.8 kB in 1s (14.9 kB/s)
Selecting previously unselected package ncompress.
(Reading database ... 244102 files and directories currently installed.)
Preparing to unpack .../ncompress_4.2.4.4-15_amd64.deb ...
Unpacking ncompress (4.2.4.4-15) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up ncompress (4.2.4.4-15) ...
```

图 3-155 安装 compress 命令

(2) 命令安装完成后,切换至原用户并执行以下命令对 ex030804 进行压缩,效果如图 3-156 所示。

```
compress -f ex030804
```

(3) 压缩完成后,会生成一个 ex030804.Z 的文件替代原文件,图 3-157 所示为执行 ls -a 查看后的效果。

```
user01@ubuntu:~$ compress -f ex030804
```

图 3-156 对 ex030804 进行压缩

```
user01@ubuntu:~$ ls -a
.          .compiz    ex030405  ex03050801  .ICEauthority
..         .config    ex030406  ex03050802  .lesshst
.bash_history Desktop    ex030501  ex030509    .local
.bash_logout .dmrc      ex030502  ex030602    Music
.bashrc     Documents  ex030503  ex030801    .oracle_jre_usage
.cache      Downloads  ex03050401 ex030801.zip Pictures
chapter03  ex030306  ex03050402 ex030802    .profile
chapter030301 ex03030601 ex03050403 ex030802- Public
chapter030302 ex030311  ex03050501 ex030803.bz2 Templates
chapter030305 ex030401  ex03050502 ex030804.Z  Videos
chapter3_1  ex030402  ex03050601 examp1es.desktop .Xauthority
chapter3_2  ex030403  ex03050602 .gconf      .xsession-errors
chapter3_3  ex030404  ex030507  .gnupg      .xsession-errors.old
```

图 3-157 显示所有文件

(4) 如果想要对压缩后的文件进行解压,我们可以使用 compress 或者 uncompress 命令。执行以下命令。

```
compress -df ex030804.Z
```

该命令执行后的效果如图 3-158 所示。

```
ubuntu@summer-virtual-machine:~$ compress -df ex030804.Z
```

图 3-158 使用 compress 命令解压缩

也可执行以下命令实现文件的解压。

```
uncompress ex030804.Z
```

该命令执行后的效果如图 3-159 所示。

```
ubuntu@summer-virtual-machine:~$ uncompress ex030804.Z
```

图 3-159 使用 uncompress 命令解压缩

### 3.8.5 uuencode 和 uudecode

本节将介绍 uuencode 和 uudecode 命令。uuencode 编码后的资料都以 begin 开始,以 end 作为结束。

(1) 首先使用 touch 命令创建名为 ex030805 的文件,再使用 uuencode 命令进行编码,命令如下。

```
uuencode ex030805
```

该命令执行后的效果如图 3-160 所示。

```
ubuntu@summer-virtual-machine:~$ touch ex030805
ubuntu@summer-virtual-machine:~$ uuencode ex030805

Command 'uuencode' not found, but can be installed with:
sudo apt install sharutils
```

图 3-160 创建名为 ex030805 的文件并使用 uuencode 命令进行压缩

上述命令执行后,系统提示 uuencode 未安装,这时只需切换至 root 用户并执行 apt-get install sharutils 命令即可,效果如图 3-161 所示。

```
root@ubuntu:/home/user01# apt-get install sharutils
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-4.8.0-36 linux-headers-4.8.0-36-generic
  linux-image-4.8.0-36-generic linux-image-extra-4.8.0-36-generic
Use 'apt autoremove' to remove them.
Suggested packages:
  bsd-mailx | mailx
The following NEW packages will be installed:
  sharutils
0 upgraded, 1 newly installed, 0 to remove and 98 not upgraded.
Need to get 148 kB of archives.
After this operation, 631 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu xenial/main amd64 sharutils amd64 1:4.15.2-1 [148 kB]
Fetched 148 kB in 1s (95.6 kB/s)
Selecting previously unselected package sharutils.
(Reading database ... 244112 files and directories currently installed.)
Preparing to unpack .../sharutils_1%3a4.15.2-1_amd64.deb ...
Unpacking sharutils (1:4.15.2-1) ...
Processing triggers for install-info (6.1.0.dfsg.1-5) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up sharutils (1:4.15.2-1) ...
```

图 3-161 安装 uuencode 命令

(2) 命令安装完成后,执行以下命令对图片 snali.jpg 进行编码并查看。

```
uuencode snali.jpg snali.jpg > snali.jpg.uue
```

```
cat snali.jpg.uue
```

该命令执行后的效果如图 3-162 所示。

```
root@ubuntu:/home/user01/chapter030805# uuencode snali.jpg snali.jpg > snali.jpg.uue
root@ubuntu:/home/user01/chapter030805# cat snali.jpg.uue
begin 644 snali.jpg
M_jc_X`02D9)1@!`0S`8`!@`#`VP!#`@&!@<!@!P<)"0@*#10-#`L+
M#IDZSP`4`1H?`AT:`IP@)"XG("L(QP<*#<I+#`Q-#0T`R<Y/3@R/"XS-#+
MVP!#`0D)"0P+#!@-#1@R(1PA,C(R,C(R,C(R,C(R,C(R,C(R,C(R,C(R,C(R
M,C(R,C(R,C(R,C(R,C(R,C(R,C(R,C+`P@`1`"U`0L#`2(``AS!`QS!`0`
M&P`04!`0`.....`0`"P0%!@?`Q`9`0S!`0$!`0`.....`0`
M`.....0(#!`7`V@`,`PS`A`#S`...)=>[RI]"0`X6-#0-:]M-!`X4T.0T
M.0T.2,3D,3TK`#\#!(1@>!@>C6#ABA)0(%`!0`130X#`Y#0Y4T6F^7M745KT\
M85*ROB>!H<I6)P`"1@GNYN8W1H5JA[8:"`QJ`'`'`<*>';')&F7-X-;EGSW
MO1Z?/M*WE>CGT%SE^G<P`KT<HWVH/GK`?N.SK)T)VXU'4LU2C!>A[<[X>VQ
MK7A&1XIH<(JF)P1J])>=Z6E`UUM;`V^`^&12[9W=CE>LZ<[>B^3I`A#SU
M)`$RR:.(4YB;K+H18:M-<.G-H<+&AP&IS: *&IPL:2B?F>HY_P"?ZV+&R0&
M]r<R*N_USR/4X6DG6Q1P]^;HF/UDOACS[&+698@I07R9LX<-Y:'`QH<AB>+&)
```

图 3-162 对图片进行编码

(3) 编码完成后,为了方便查看结果,可将编码后的 snali.jpg.uue 文件移动至空白文件夹并进行解码,命令如下。

```
uudecode snali.jpg.uue
```

该命令执行后,若在当前文件夹下生成 snali.jpg,则说明命令执行成功。读者亦可进行验证。

### 3.8.6 dump 和 tar

本节将介绍 dump 和 tar 命令。

(1) 首先,使用 dump 命令将 home 目录内的文件进行备份,命令如下。

```
dump -0 -f home.dump /home
```

该命令执行后的效果如图 3-163 所示。

```
ubuntu@summer-virtual-machine:~$ dump -0 -f home.dump/home
Command 'dump' not found, but can be installed with:
sudo apt install dump
```

图 3-163 使用 dump 命令将目录内的文件进行备份

可以看到,系统提示 dump 命令未安装,此时只需切换至 root 用户并执行 apt install dump 即可,效果如图 3-164 所示。

```
root@ubuntu:/home/user01# apt install dump
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-4.8.0-36 linux-headers-4.8.0-36-generic
  linux-image-4.8.0-36-generic linux-image-extra-4.8.0-36-generic
Use 'apt autoremove' to remove them.
The following NEW packages will be installed:
  dump
0 upgraded, 1 newly installed, 0 to remove and 91 not upgraded.
Need to get 189 kB of archives.
After this operation, 594 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu/xenial/universe amd64 dump amd64 0.4b4
4-7 [189 kB]
Fetched 189 kB in 1s (109 kB/s)
Selecting previously unselected package dump.
(Reading database ... 244129 files and directories currently installed.)
Preparing to unpack .../dump_0.4b44-7_amd64.deb ...
Unpacking dump (0.4b44-7) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up dump (0.4b44-7) ...
update-alternatives: using /usr/sbin/rmt-dump to provide /usr/sbin/rmt (rmt) in
auto mode
```

图 3-164 安装 dump 命令

(2) dump 安装完毕后,接下来执行以下命令进行备份。

```
dump -0 -f /tmp/boot.dump /boot
```

该命令执行后的效果如图 3-165 所示。

(3) 也可以使用如下命令对 user01 文件夹进行备份。

```
tar cvf backup.tar /home/user01
```

该命令执行后的效果如图 3-166 所示。

(4) 然后,使用 tar 命令将当前目录下所有文件打包并压缩归档到文件 this.tar.gz 中,具体如下。

```
tar czvf this.tar.gz ./
```

```

root@ubuntu:/home/user01# dump -0 -f /tmp/boot.dump /boot
DUMP: Date of this level 0 dump: Sat Jun 10 00:48:14 2017
DUMP: Dumping /dev/sda1 (/ (dtr boot)) to /tmp/boot.dump
DUMP: Label: none
DUMP: Writing 10 Kilobyte records
DUMP: mapping (Pass I) [regular files]
DUMP: mapping (Pass II) [directories]
DUMP: estimated 216218 blocks.
DUMP: Volume 1 started with block 1 at: Sat Jun 10 00:48:15 2017
DUMP: dumping (Pass III) [directories]
DUMP: dumping (Pass IV) [regular files]
DUMP: Closing /tmp/boot.dump
DUMP: Volume 1 completed at: Sat Jun 10 00:48:36 2017
DUMP: Volume 1 216180 blocks (211.11MB)
DUMP: Volume 1 took 0:00:21
DUMP: Volume 1 transfer rate: 10294 kB/s
DUMP: 216180 blocks (211.11MB) on 1 volume(s)
DUMP: finished in 19 seconds, throughput 11377 kBbytes/sec
DUMP: Date of this level 0 dump: Sat Jun 10 00:48:14 2017
DUMP: Date this dump completed: Sat Jun 10 00:48:36 2017
DUMP: Average transfer rate: 10294 kB/s
DUMP: DUMP IS DONE

```

图 3-165 使用 dump 命令进行备份

该命令执行后的效果如图 3-167 所示。

```

root@ubuntu:/home/user01# tar cvf backup.tar /home/user01
tar: Removing leading '/' from member names
/home/user01/
/home/user01/.xsession-errors
/home/user01/ex030402
/home/user01/Pictures/
/home/user01/ex030406
/home/user01/chapter030305/
/home/user01/chapter030305/ex030305
/home/user01/ex030802~
/home/user01/ex030311
/home/user01/ex030504_02
/home/user01/examples.desktop
/home/user01/ex03050501
/home/user01/.xsession-errors.old
/home/user01/ex030801.zip
/home/user01/test
/home/user01/.bash_logout
/home/user01/ex030403

```

图 3-166 使用 tar 命令进行备份

```

user01@ubuntu:~$ tar czvf this.tar.gz ./
./
./xsession-errors
./ex030402
./Pictures/
./ex030406
./chapter030305/
./chapter030305/ex030305
./ex030802~
./ex030311
./ex030504_02
./examples.desktop
./ex03050501
./xsession-errors.old
./ex030801.zip
./test
./bash_logout
./ex030403
./ex030801
./profile
./ex03050403
./Videos/

```

图 3-167 将当前目录下所有文件打包并压缩

(5) 最后,将 this.tar.gz 文件移动至空白文件夹中并解压缩,具体如下。

```
tar xzvf this.tar.gz ./
```

命令执行后,当前空白文件夹下将出现被压缩的所有文件。

## 3.9 获取帮助

在使用 Ubuntu 系统时,可能会遇到各种问题。下面就来学习在遇到问题时该如何获取帮助。

### 3.9.1 使用 man 获取帮助

在 Ubuntu 中可以使用 man 来获取帮助。

(1) 输入 man ls 查看 ls 命令的帮助手册,命令如下。

```
man ls
```

该命令执行后的效果如图 3-168 所示。

可以看到,它会在左上角显示“LS(1)”,这里,“LS”表示手册名称,而“(1)”表示该手册位于第 1 章。

(2) 继续执行 man sleep 命令查看 sleep 命令的帮助手册,命令如下。

```

LS(1)                                User Commands                                LS(1)
NAME
  ls - list directory contents
SYNOPSIS
  ls [OPTION]... [FILE]...
DESCRIPTION
  List information about the FILES (the current directory by default).
  Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
  fied.

  Mandatory arguments to long options are mandatory for short options
  too.

  -a, --all
      do not ignore entries starting with .
  -A, --almost-all
      do not list implied . and ..

  --author
Manual page ls(1) line 1 (press h for help or q to quit)

```

图 3-168 查看 ls 命令的帮助手册

man sleep

该命令执行后的效果如图 3-169 所示。

```

SLEEP(1)                              User Commands                              SLEEP(1)
NAME
  sleep - delay for a specified amount of time
SYNOPSIS
  sleep NUMBER[SUFFIX]...
  sleep OPTION
DESCRIPTION
  Pause for NUMBER seconds. SUFFIX may be 's' for seconds (the default),
  'm' for minutes, 'h' for hours or 'd' for days. Unlike most implemen-
  tations that require NUMBER be an integer, here NUMBER may be an arbi-
  trary floating point number. Given two or more arguments, pause for
  the amount of time specified by the sum of their values.

  --help display this help and exit
  --version
      output version information and exit
AUTHOR
  Written by Jim Meyering and Paul Eggert.
Manual page sleep(1) line 1 (press h for help or q to quit)

```

图 3-169 查看 sleep 命令的帮助手册

可以看到,它会在左上角显示“SLEEP(1)”,在这里,“SLEEP”表示手册名称,而“(1)”表示该手册位于第 1 章。

(3) 如果记不清楚完整的命令,可以考虑用 -k 参数,命令如下。

man -k sleep

该命令执行后的效果如图 3-170 所示。

```

user01@ubuntu:~$ man -k sleep
clock_nanosleep (2) - high-resolution sleep with specifiable clock
nanosleep (2) - high-resolution sleep
rtctime (8) - enter a system sleep state until specified wakeup time
sleep (1) - delay for a specified amount of time
sleep (3) - sleep for the specified number of seconds
sleep.conf.d (5) - Suspend and hibernation configuration file
systemd-hibernate.service (8) - System sleep state logic
systemd-hybrid-sleep.service (8) - System sleep state logic
systemd-sleep (8) - System sleep state logic
systemd-sleep.conf (5) - Suspend and hibernation configuration file
systemd-suspend.service (8) - System sleep state logic
usleep (3) - suspend execution for microsecond intervals

```

图 3-170 查找和 sleep 有关的帮助

### 3.9.2 使用 whatis 获取帮助

如果想查看一个命令的功能,那么可以使用 whatis 命令。执行 whatis ls 命令可查看 ls

命令的功能,命令如下。

```
whatis ls
```

该命令执行后的效果如图 3-171 所示。

```
user01@ubuntu:~$ whatis ls
ls (1)                  - list directory contents
```

图 3-171 查看 ls 命令的功能

### 3.9.3 使用 help 获取帮助

如果想查看一个内部命令的帮助信息,这时我们可以使用 help 命令。

(1) 执行 help 命令查看 cd 命令的帮助信息,命令如下。

```
help cd
```

该命令执行后的效果如图 3-172 所示。

(2) 执行 help -d 命令查看 cd 命令的简短描述,命令如下。

```
help -d cd
```

该命令执行后的效果如图 3-173 所示。

```
cd: cd [-L][-P [-e]] [-@]] [dir]
Change the shell working directory.

Change the current directory to DIR. The default DIR is the value of the
HOME shell variable.

The variable CDPATH defines the search path for the directory containing
DIR. Alternative directory names in CDPATH are separated by a colon (:).
A null directory name is the same as the current directory. If DIR begins
with a slash (/), then CDPATH is not used.

If the directory is not found, and the shell option 'cdable_vars' is set,
the word is assumed to be a variable name. If that variable has a value,
its value is used for DIR.

Options:
-L force symbolic links to be followed: resolve symbolic links in
  DIR after processing instances of '..'
-P use the physical directory structure without following symbolic
```

图 3-172 查看 cd 命令的帮助信息

```
user01@ubuntu:~$ help -d cd
cd - Change the shell working directory.
```

图 3-173 查看 cd 命令的简短描述

(3) 执行 help -s 命令查看 cd 命令用法的简介,命令如下。

```
help -s cd
```

该命令执行后的效果如图 3-174 所示。

(4) 使用 help 命令查看非内部命令 ls,命令如下。

```
help ls
```

该命令执行后的效果如图 3-175 所示。

```
user01@ubuntu:~$ help -s cd
cd: cd [-L][-P [-e]] [-@]] [dir]
```

图 3-174 查看 cd 命令用法的简介

```
user01@ubuntu:~$ help ls
bash: help: no help topics match `ls'. Try 'help help' or 'man -k ls' or 'info
ls'.
```

图 3-175 查看非内部命令 ls

可以看到,系统提示无法找到 ls 命令的帮助信息,建议尝试使用 man -k 或 info 命令。

### 3.9.4 使用 info 获取帮助

info 命令也可用来获取帮助。我们执行如下命令查看 who 命令的帮助信息。

info who

该命令执行后的效果如图 3-176 所示。

```

rev: users invocation, Up: User information

20.6 'who': Print who is currently logged in
=====
'who' prints information about users who are currently logged on.
Synopsis:

    who [OPTION] [FILE] [an i]

    If given no non-option arguments, 'who' prints the following
    information for each user currently logged on: login name, terminal
    line, login time, and remote hostname or X display.

    If given one non-option argument, 'who' uses that instead of a
    default system-maintained file (often '/var/run/utmp' or '/etc/utmp') as
    the name of the file containing the record of users logged on.
    '/var/log/wtmp' is commonly given as an argument to 'who' to look at who
    has previously logged on.

    If given two non-option arguments, 'who' prints only the entry for
    the user running it (determined from its standard input), preceded by
    -----Info: (coreutils.info.gz)who invocation, 107 lines --Top-----
Unknown command (M-C-a).

```

图 3-176 查看 who 命令的帮助信息

## 3.10 其他命令

本节简要介绍 Ubuntu 系统中的其他命令,包括清除屏幕信息、显示日期时间、查看和终止进程等。

### 3.10.1 清除屏幕信息

清除屏幕信息的命令是 clear,在终端上执行 clear 命令前,效果如图 3-177 所示。

```

user01@ubuntu:~$ ls
1234      Downloads      ex03050401  ex030506_02  ex030802-
abc.txt   ex030306      ex030504_01  ex030506_03  ex030803.bz2
backup.tar ex030306_01  ex03050402  ex030506_04  ex030804
ex030311  ex030504_02  ex030507    ex030805
chapter030301 ex030401    ex03050403  ex03050801  examples.desktop
chapter030302 ex030402    ex030504_03  ex030508_01  Music
chapter030305 ex030403    ex03050501  ex03050802  Pictures
chapter030805 ex030404    ex030505_01  ex030508_02  Public
chapter3.1    ex030405    ex03050502  ex030509    Templates
chapter3.2    ex030406    ex030505_02  ex030602    test
chapter3.3    ex030501    ex03050601  ex030801    this.tar.gz
Desktop      ex030502    ex030506_01  ex030801.zip Videos
Documents    ex030503    ex03050602  ex030802
user01@ubuntu:~$ clear

```

图 3-177 使用 clear 命令前的终端

按 Enter 键执行 clear 命令后,效果如图 3-178 所示。

```

user01@ubuntu: ~
user01@ubuntu:~$

```

图 3-178 使用 clear 命令后的终端

可以看到,命令成功执行后,终端中显示的内容被全部清空。

### 3.10.2 显示文本

(1) echo 命令的功能是在显示器上显示一段文字,一般起到提示的作用,具体如下。

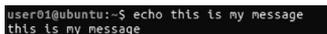
```
echo this is my message
```

该命令执行后的效果如图 3-179 所示。

(2) 也可以使用 echo 命令输入某个变量,例如,可以定义一个名为“x”的变量并赋值为 10,然后将其显示出来,具体如下。

```
x = 10
echo $ x
```

该命令执行后的效果如图 3-180 所示。



```
user01@ubuntu:~$ echo this is my message
this is my message
```

图 3-179 显示一段文字



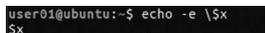
```
user01@ubuntu:~$ x=10
user01@ubuntu:~$ echo $x
10
```

图 3-180 显示一个变量中存的值

(3) 如果只是输入“\$ x”这个字符串,那么可以使用转义符号进行操作,具体如下。

```
echo -e \ $ x
```

该命令执行后的效果如图 3-181 所示。



```
user01@ubuntu:~$ echo -e \$x
$x
```

这样,系统就把“\$ x”当做普通的字符串进行解析,而不是变量。图 3-181 显示一个转义后的字符串

(4) 使用 echo 命令输出当前工作目录,具体如下。

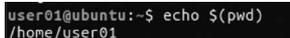
```
echo $(pwd)
```

该命令执行后的效果如图 3-182 所示。

(5) 也可以使用以下命令输出某个用户的 home 目录名。

```
echo ~user01
```

该命令执行后的效果如图 3-183 所示。



```
user01@ubuntu:~$ echo $(pwd)
/home/user01
```

图 3-182 输出当前工作目录



```
user01@ubuntu:~$ echo ~user01
/home/user01
```

图 3-183 输出某个用户的 home 目录名

命令执行完成后,会输出浪纹线后指定用户的 home 目录名。如果没有指定用户名,则是当前用户的 home 目录。

### 3.10.3 显示日期时间

(1) 显示日期和时间的命令是 date,执行以下命令。

```
date
```

该命令执行后的效果如图 3-184 所示。

(2) 如果只想显示年月日中的某一个,可以执行以下命令。

```
date + %y
date + %m
date + %d
```

该命令执行后的效果如图 3-185 所示。

```
ubuntu@summer-virtual-machine:~$ date
Mon 21 Sep 2020 02:50:52 AM CST
```

图 3-184 显示日期和时间

```
ubuntu@summer-virtual-machine:~$ date +%y
20
ubuntu@summer-virtual-machine:~$ date +%m
09
ubuntu@summer-virtual-machine:~$ date +%d
21
```

图 3-185 显示年月日

可以看到,当前时间为 2020 年 09 月 21 日。

(3) 除了 date 命令外,cal 命令也可以显示日历及当前日期,执行以下命令。

```
cal
```

该命令执行后的效果如图 3-186 所示。

(4) 也可以使用 cal 命令查看当前年份的日历,命令如下。

```
cal -y
```

该命令执行后的效果如图 3-187 所示。

```
user01@ubuntu:~$ cal
      June 2017
Su Mo Tu We Th Fr Sa
                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
```

图 3-186 显示日历及当前日期

```
user01@ubuntu:~$ cal -y
                2017
January February March
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
 1  2  3  4  5  6  7           1  2  3  4           1  2  3  4
 8  9 10 11 12 13 14   5  6  7  8  9 10 11   5  6  7  8  9 10 11
15 16 17 18 19 20 21 12 13 14 15 16 17 18 12 13 14 15 16 17 18
22 23 24 25 26 27 28 19 20 21 22 23 24 25 19 20 21 22 23 24 25
29 30 31           26 27 28           26 27 28 29 30 31

April May June
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
                1           1  2  3  4  5  6           1  2  3
 2  3  4  5  6  7  8   7  8  9 10 11 12 13   4  5  6  7  8  9 10
 9 10 11 12 13 14 15 14 15 16 17 18 19 20 11 12 13 14 15 16 17
16 17 18 19 20 21 22 21 22 23 24 25 26 27 18 19 20 21 22 23 24
23 24 25 26 27 28 29 28 29 30 31           25 26 27 28 29 30
30

July August September
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
                1           1  2  3  4  5           1  2
 2  3  4  5  6  7  8   6  7  8  9 10 11 12   3  4  5  6  7  8  9
 9 10 11 12 13 14 15 13 14 15 16 17 18 19 10 11 12 13 14 15 16
16 17 18 19 20 21 22 20 21 22 23 24 25 26 17 18 19 20 21 22 23
23 24 25 26 27 28 29 27 28 29 30 31   24 25 26 27 28 29 30
30 31

October November December
Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa Su Mo Tu We Th Fr Sa
```

图 3-187 查看当前年份的日历

### 3.10.4 查看当前进程

(1) 执行查看当前进程的命令是 ps,效果如图 3-188 所示。

```
user01@ubuntu:~$ ps
  PID TTY          TIME CMD
 2315 pts/4    00:00:00 bash
 2886 pts/4    00:00:00 ps
```

图 3-188 查看当前进程

结果默认会显示 4 列信息,具体如表 3-4 所示。

表 3-4 ps 命令的结果及描述

英文简称	PID	TTY	TIME	CMD
中文描述	运行中的命令 (CMD)的进程编号	命令所运行的位置 (终端)	运行中的该命令所占用的 CPU 处理时间	该进程所运行的命令

(2) 接下来查看所有进程,命令如下。

```
ps -A
```

该命令执行后的效果如图 3-189 所示。

(3) 最后查看 user01 用户的所有进程,命令如下。

```
ps U user01
```

该命令执行后的效果如图 3-190 所示。

```
user01@ubuntu:~$ ps -A
  PID TTY          TIME CMD
    1 ?        00:00:02 systemd
    2 ?        00:00:00 kthreadd
    3 ?        00:00:00 ksoftirqd/0
    4 ?        00:00:00 kworker/0:0
    5 ?        00:00:00 kworker/0:0H
    7 ?        00:00:00 rcu_sched
    8 ?        00:00:00 rcu_bh
    9 ?        00:00:00 migration/0
   10 ?        00:00:00 lru-add-drain
   11 ?        00:00:00 watchdog/0
   12 ?        00:00:00 cpuhp/0
   13 ?        00:00:00 kdevtmpfs
   14 ?        00:00:00 metns
   15 ?        00:00:00 khungtaskd
   16 ?        00:00:00 oom_reaper
   17 ?        00:00:00 writeback
   18 ?        00:00:00 kcompactd0
   19 ?        00:00:00 ksmd
```

图 3-189 查看所有进程

```
user01@ubuntu:~$ ps U user01
  PID TTY          STAT TIME   COMMAND
 1486 ?        Ss   0:00 /lib/systemd/systemd --user
 1487 ?        S    0:00 (sd-pam)
 1492 ?        Sl   0:00 /usr/bin/gnome-keyring-daemon --daemonize --login
 1494 ?        Ss   0:00 /sbin/upstart --user
 1580 ?        S    0:00 upstart-udev-bridge --daemon --user
 1581 ?        Ss   0:00 dbus-daemon --fork --session --address=unix:abstract
 1593 ?        Ss   0:00 /usr/lib/x86_64-linux-gnu/udm/window-stack-bridge
 1621 ?        S    0:00 upstart-dbus-bridge --daemon --session --user --bus-
 1622 ?        S    0:00 upstart-dbus-bridge --daemon --system --user --bus-n
 1623 ?        Ssl  0:00 /usr/bin/ibus-daemon --daemonize --xim --address uni
 1640 ?        S    0:00 upstart-file-bridge --daemon --user
 1646 ?        Ss   0:00 gpg-agent --homedir /home/user01/.gnupg --use-standa
 1650 ?        Sl   0:00 /usr/lib/gvfs/gvfsd
 1655 ?        Sl   0:00 /usr/lib/gvfs/gvfsd-fuse /run/user/1001/gvfs -f -o b
 1664 ?        Sl   0:00 /usr/lib/ibus/ibus-dconf
 1665 ?        Sl   0:00 /usr/lib/ibus/ibus-ui-gtk3
 1667 ?        Sl   0:00 /usr/lib/ibus/ibus-x11 --kill-daemon
 1685 ?        Sl   0:00 /usr/lib/at-spi2-core/at-spi-bus-launcher
 1690 ?        S    0:00 /usr/bin/dbus-daemon --config-file=/etc/at-spi2/acce
 1692 ?        Sl   0:00 /usr/lib/at-spi2-core/at-spi2-registrd --use-gnome-
```

图 3-190 查看 user01 用户的所有进程

### 3.10.5 终止某一进程

(1) 如果想关闭图 3-188 中 PID 为 2315 的进程,执行以下命令。

```
kill -9 2315
```

上述命令成功执行后,此进程被关闭,也就是系统将退出终端。

(2) 也可以根据程序名终止某一进程,例如,想终止程序名为 sendmail 的这一进程,可执行以下命令。

```
ps -ef | grep sendmail
pkill sendmail
```

上述命令成功执行后,程序名为 sendmail 的进程将被终止。

(3) 还可以使用 killall 命令来实现 pkill 的功能,接下来仍以终止程序名为 sendmail 这一进程为例,执行以下命令。

```
ps -ef | grep sendmail
killall -9 sendmail
```

若 killall 命令被成功执行,则进程 sendmail 将被关闭。

### 3.10.6 显示最近登录系统的用户信息

显示最近登录系统的用户信息的命令是 last,在终端上输入以下命令。

```
last
```

该命令执行后的效果如图 3-191 所示。

结果默认会显示 6 列信息,表 3-5 展示了执行该命令后,结果的前 3 行及其释义。

```

user01@ubuntu:~$ last
user01 tty7 :0 Wed Jun 14 23:09 gone - no logout
reboot system boot 4.8.0-54-generic Wed Jun 14 23:08 still running
user01 tty7 :0 Mon Jun 12 04:09 crash (2+18:58)
reboot system boot 4.8.0-54-generic Mon Jun 12 04:08 still running
user01 tty7 :0 Sat Jun 10 03:06 crash (2+01:02)
reboot system boot 4.8.0-54-generic Sat Jun 10 03:05 still running
user01 tty7 :0 Sat Jun 10 00:19 crash (02:46)
reboot system boot 4.8.0-54-generic Sat Jun 10 00:06 still running
user01 pts/4 192.168.191.1 Fri Jun 9 21:23 21:27 (00:03)
user01 pts/17 Fri Jun 9 21:17 crash (02:48)
user01 tty7 :0 Fri Jun 9 19:50 crash (04:15)
reboot system boot 4.8.0-54-generic Fri Jun 9 19:48 still running
user01 tty7 :0 Wed Jun 7 00:02 crash (2+19:45)
reboot system boot 4.8.0-52-generic Wed Jun 7 00:02 still running
user01 tty7 :0 Mon Jun 5 06:35 down (01:02)
reboot system boot 4.8.0-52-generic Mon Jun 5 06:34 07:37 (01:03)
user01 pts/18 Mon Jun 5 00:08 crash (06:25)
user01 tty7 :0 Sun Jun 4 23:44 crash (06:49)
reboot system boot 4.8.0-52-generic Sun Jun 4 23:43 07:37 (07:54)
user01 tty7 :0 Sun Jun 4 02:00 crash (21:42)
reboot system boot 4.8.0-52-generic Sun Jun 4 02:00 07:37 (1+05:37)

wtmp begins Sun Jun 4 02:00:00 2017

```

图 3-191 显示最近登录系统的用户信息

表 3-5 last 命令的结果及释义

序号	用户名	终端	登录 IP 或者内核	开始时间	结束时间	持续时间
1	user01	tty7	:0	Wed Jun 14 23:09	gone-no logout	/
2	reboot	system boot	4.8.0-54-generic	Wed Jun 14 23:08	still running	/
3	user01	tty7	:0	Mon Jun 12 04:09	crash	2+18:58

### 3.10.7 显示历史命令

显示历史命令的命令是 history, 系统默认保留最近执行的 1000 条命令。图 3-192 显示了执行 history 之后的结果, 该结果分为 2 列, 第 1 列为执行过的命令的编号, 第 2 列为具体的命令。

也可以使用如下命令显示最近执行过的 10 条命令。

```
history 10
```

该命令执行后的效果如图 3-193 所示。

```

user01@ubuntu:~$ history
1 pwd
2 ls
3 ls -a
4 tree user01
5 sudo apt-get install tree
6 tree user01
7 tree -L user01
8 ls
9 cd chapter03
10 ls
11 cd ..
12 ls
13 cd user01
14 ls
15 tree -l chapter03
16 cd chapter03
17 touch ex030310
18 ls
19 cd ..
20 tree -L chapter03
21 tree -l chapter03
22 cd chapter03
23 touch ex030311

```

图 3-192 显示历史命令

```

user01@ubuntu:~$ history 10
504 cat > test
505 grep -c 'a' test
506 ls
507 cd chapter030805
508 unencode snali.jpg snali.jpg
509 uuencode snali.jpg snali.jpg
510 uuencode snali.jpg snali.jpg>snali.jpg.uue
511 su
512 history
513 history 10

```

图 3-193 显示最近执行过的 10 条命令

从图中可以看到, 到目前为止在终端上一共执行了 513 条命令, 最近 10 条命令为第 504 条到第 513 条。

**注意:** 成功执行 history 命令之后, 系统默认将会显示最近执行的 1000 条命令, 这 1000 条命令由 ~/.bash\_history 里的命令及当前 Shell 中命令所组成。

若需修改这一默认命令数目,则可执行以下操作:

```
sed -i 's/^HISTSIZE = 1000/HISTSIZE = 300/' /etc/profile
source /etc/profile (使其生效)
```

其中,HISTSIZE=1000 表示显示最近执行的 1000 条命令,我们将其值修改为 300。上述命令成功执行后,history 执行时只会显示最近执行的 300 条命令。

### 3.10.8 超级权限用户及操作

在 Ubuntu 系统中的权限分为超级(root)用户权限和普通用户权限,两者的主要区别如表 3-6 所示。

表 3-6 超级用户与普通用户的区别

序号	超级用户	普通用户
1	没有磁盘空间的限制	可使用的磁盘空间有限制
2	对所有文件都具有读写和修改的权限	只有文件的所有者具有文件的修改权限
3	可以使用任何命令	可以使用大部分命令,某些命令如果没有权限可以使用 sudo 申请 root 权限

以用户执行命令为例,普通用户可以被允许使用 sudo 命令,临时赋予 root 权限,但是必须要将其加入到 sudoers 用户组里,具体步骤如下。

(1) 首先,在普通用户的字符界面下执行 halt 命令,具体如下。

```
halt
```

该命令执行后的效果如图 3-194 所示。

```
ubuntu@summer-virtual-machine:~$ halt
Failed to halt system via loginctl: Interactive authentication required.
Failed to open initctl fifo: Permission denied
Failed to talk to init daemon.
```

图 3-194 普通用户执行 halt 命令

可以看到,系统提示由于权限不够无法执行该操作。

(2) 然后,使用 sudo 命令进行操作,具体如下。

```
sudo halt
```

该命令执行后的效果如图 3-195 所示。

系统提示输入当前用户 user01 的密码,如果密码输入错误,系统将提示重新输入,输入正确后可能会出现如图 3-196 所示的提示。

```
user01@ubuntu:~$ sudo halt
[sudo] password for user01: █
```

图 3-195 使用 sudo 命令执行 halt

```
user01@ubuntu:~$ sudo halt
[sudo] password for user01:
user01 is not in the sudoers file. This incident will be reported.
```

图 3-196 密码输入正确后出现的提示

这是因为 user01 用户是普通用户,不具有超级用户的权限,它不在具有超级用户权限的用户组 sudoers 的列表里。

(3) 使用 su 命令切换至 root 用户并执行以下命令以完成将普通用户 user01 加入具有超级用户权限的 sudoers 用户组内。

```
adduser user01 sudo
```

该命令执行后的效果如图 3-197 所示。

```
user01@ubuntu:~$ su
Password:
root@ubuntu:/home/user01# adduser user01 sudo
Adding user `user01' to group `sudo' ...
Adding user user01 to group sudo
Done.
```

图 3-197 切换至 root 用户并将 user01 加入 sudoers 列表里

这时切换至 user01 用户后继续执行 sudo halt 命令,系统将自动关闭,这说明用户 user01 已经被成功加入超级用户组并取得超级用户的权限。

### 3.10.9 定义别名

(1) alias 命令用来设置命令的别名。可以通过使用该命令将一些较长的命令进行简化,具体如下。

```
alias l='ls'
l
ls
```

上述命令执行的效果如图 3-198 所示。

```
user01@ubuntu:~$ alias l='ls'
user01@ubuntu:~$ l
1234      Downloads      ex03050401    ex030506_02    ex030802~
abc.txt   ex030306      ex030504_01  ex030506_03    ex030803.bz2
backup.tar ex030306_01  ex03050402   ex030506_04    ex030804
chapter03 ex030311     ex030504_02  ex030507       ex030805
chapter030301 ex030401    ex03050403   ex03050801     examples.desktop
chapter030302 ex030402    ex030504_03  ex030508_01    Music
chapter030305 ex030403    ex03050501   ex03050802     Pictures
chapter030805 ex030404    ex030505_01  ex030508_02    Public
chapter3.1 ex030405    ex03050502   ex030509       Templates
chapter3.2 ex030406    ex030505_02  ex030602       test
chapter3.3 ex030501    ex03050601   ex030801       this.tar.gz
Desktop    ex030502    ex030506_01  ex030801.zip   Videos
Documents  ex030503    ex03050602   ex030802

user01@ubuntu:~$ ls
1234      Downloads      ex03050401    ex030506_02    ex030802~
abc.txt   ex030306      ex030504_01  ex030506_03    ex030803.bz2
backup.tar ex030306_01  ex03050402   ex030506_04    ex030804
chapter03 ex030311     ex030504_02  ex030507       ex030805
chapter030301 ex030401    ex03050403   ex03050801     examples.desktop
chapter030302 ex030402    ex030504_03  ex030508_01    Music
chapter030305 ex030403    ex03050501   ex03050802     Pictures
chapter030805 ex030404    ex030505_01  ex030508_02    Public
```

图 3-198 设置命令的别名

可以看到,我们将 ls 命令的别名设置为“l”,执行“l”命令后会得到与 ls 命令相同的结果。如果要删除“l”这个别名,只需执行 unalias l 即可。

(2) 也可以使用 alias 命令列出当前系统中所有的别名及其对应的命令,具体如下。

```
alias
```

该命令执行的效果如图 3-199 所示。

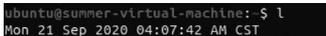
```
user01@ubuntu:~$ alias
alias alert='notify-send --urgency=low -i "${@} [ $? = 0 ] && echo terminal || echo error)" "${history|tail -n1|sed -e '\''s/^\s*[0-9]\s*//;s/[;&|]\s*/\|'\''}'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l='ls'
alias la='ls -A'
alias ll='ls -alF'
alias ls='ls --color=auto'
```

图 3-199 列出当前系统中所有的别名及其对应的命令

(3) 我们已经设置 ls 命令的别名为“l”,这时我们将 date 命令的别名也设置为“l”,尝试以下命令。

```
alias l= 'date'  
l
```

上述命令执行的效果如图 3-200 所示。



```
ubuntu@summer-virtual-machine:~$ l  
Mon 21 Sep 2020 04:07:42 AM CST
```

图 3-200 将 date 命令的别名也设置为“l”

可以看到,命令成功执行后,别名“l”被设置为 date 命令,自动覆盖之前设置的 ls 命令。这也是为什么我们在最初执行“alias l= 'ls'”时无须查看“l”是否被设置为其他命令的别名,因为它会被自动覆盖。

## 本章小结

在 Ubuntu 字符界面下完全使用命令来进行所有操作。在本章中,我们仅介绍了最为常用的命令,大致包括:用户登录与注销、关闭和重启系统;目录与文件的更改、创建、删除、移动等;文件内容的显示及剪切、粘贴、比较和统计;在硬盘和数据库中查找文件或目录及其位置;检查磁盘空间占用情况、挂载文件系统、分配和显示配额、卸载文件系统等;压缩和解压缩、备份和恢复;获取帮助以及其他一些常用的命令。

## 习题 3

### 1. 选择题

- (1) 普通用户登录后,以下( )命令可以打开新终端。  
A. Ctrl+Alt+R  
B. Ctrl+Alt+T  
C. Ctrl+Alt+Y  
D. Ctrl+Alt+U
- (2) 普通用户打开 Shell 后,以下( )命令可以切换至 root 账户。  
A. su  
B. adduser  
C. login  
D. exit
- (3) ( )命令不能执行关闭系统操作。  
A. shutdown -h  
B. halt  
C. shutdown -r  
D. poweroff
- (4) 如果想要使用 init 命令重启,那么应该使用以下( )级别。  
A. 1  
B. 3  
C. 5  
D. 6
- (5) ( )命令不能执行重启系统的操作。  
A. shutdown -h  
B. init 6  
C. shutdown -r  
D. reboot
- (6) ( )命令可以创建名为 abc 的文件。  
A. touch abc  
B. mkdir abc  
C. rmdir abc  
D. cd abc
- (7) Ubuntu 系统的文件不包括( )权限。  
A. 写入  
B. 执行  
C. 删除  
D. 读取
- (8) 如果想查看文件的权限,那么需要使用( )命令。

- A. ls -l                      B. ls -d                      C. ls -m                      D. ls -R
- (9) (     ) 按键是 more 命令向后翻页的快捷键。
- A. Del                      B. Enter                      C. Space                      D. Page Up
- (10) 如果想以逆序查看文件内容,那么 sort 命令后需要加上(     )参数。
- A. -a                      B. -d                      C. -m                      D. -r
- (11) cut -b 命令在处理中文文档的时候会出现乱码,而(     )命令可以正常输出。
- A. cut -a                      B. cut -c                      C. cut -d                      D. cut -e
- (12) 使用 wc 命令统计文件中字符数,(     )命令可以正确执行。
- A. wc -c                      B. wc -l                      C. wc -w                      D. wc -d
- (13) 如果我们想使刚创建的文件被 locate 命令立即搜索到,那么我们可以执行(     )命令。
- A. update                      B. update database  
C. updatedb                      D. updating
- (14) 如果我们需要使用 locate 命令搜索当前工作目录下以“test”结尾的文件,那么以下(     )命令是正确的。
- A. locate -r test \$                      B. locate -r test #  
C. locate -r test %                      D. locate -r test \$ \*
- (15) 如果在使用 unzip 命令解压缩过程中不想覆盖原文件,那么执行(     )命令。
- A. unzip -n                      B. unzip -ne                      C. unzip -m                      D. unzip -a
- (16) (     ) 仅用于查看一个内部命令的帮助信息。
- A. whatis                      B. help                      C. man                      D. info
- (17) 如果想使用 echo 命令输出 pwd 命令的结果,(     )命令可以正确执行。
- A. echo # (pwd)                      B. echo % (pwd)  
C. echo pwd                      D. echo \$ (pwd)
- (18) 如果想使用 date 命令输出当前的年份,(     )命令可以正确执行。
- A. date +%y                      B. date +%m                      C. date +%d                      D. date +%w
- (19) (     ) 选项不是 ps 命令执行后的结果。
- A. PID                      B. TTY                      C. CMD                      D. DATE
- (20) (     ) 选项不是 last 命令执行后的结果。
- A. 用户名                      B. 密码                      C. 开始时间                      D. 结束时间

## 2. 填空题

- (1) Ubuntu 系统里具有最高权限的用户名是\_\_\_\_\_。
- (2) 如果想要退出 Shell,那么我们要使用\_\_\_\_\_命令。
- (3) 如果想要创建可以立刻使用的账户,那么我们需要使用\_\_\_\_\_命令。
- (4) 在 Ubuntu 系统中,显示当前工作目录的命令是\_\_\_\_\_。
- (5) 如果我们想要切换工作目录,那么我们需要使用\_\_\_\_\_命令。
- (6) 使用 cat 命令创建文件后系统提示输入内容,按下\_\_\_\_\_可以结束输入。
- (7) 如果使用 chmod 命令给一个文件赋予 444 权限,那么任何人对该文件都具有\_\_\_\_\_权限。

- (8) 使用 less 命令分页显示文件内容时,如果想翻到上一页需要按下\_\_\_\_\_键。
- (9) head 命令默认显示文件的前\_\_\_\_\_行。
- (10) \_\_\_\_\_命令可以对 3 个文件的内容进行比较。
- (11) 命令\_\_\_\_\_ /dev/sdb4 将 sdb4 分区格式化为 ext4 文件系统。
- (12) 使用 quotacheck 命令检查 U 盘空间配置后,会生成 aquota.user 和\_\_\_\_\_文件。
- (13) 如果想要产生具有“zip”扩展名的压缩文件,那么我们就要使用\_\_\_\_\_命令。
- (14) 如果由于压缩包损坏而无法读取,那么这时候可以使用\_\_\_\_\_命令修复压缩包。
- (15) 在解压以“.z”结尾的压缩文件时,我们可以使用 compress -df 或者\_\_\_\_\_命令。
- (16) 在使用 Ubuntu 系统时,如果我们遇到各种问题需要查看某条命令的帮助手册时,应该使用\_\_\_\_\_命令。
- (17) 显示历史命令的命令是 history,系统默认保留最近执行的\_\_\_\_\_条命令。
- (18) 普通用户可以被允许使用\_\_\_\_\_命令,临时赋予 root 权限。
- (19) 如果要删除“w”这个别名,我们只需执行\_\_\_\_\_即可。
- (20) 执行\_\_\_\_\_命令可以完成将普通用户 user01 加入具有超级用户权限的 sudoers 用户组内的操作。

### 3. 简答题

- (1) 说一说 Ubuntu 系统内部命令和外部命令的区别。
- (2) 用户在登录系统输入密码时,会显示出来吗?为什么?
- (3) 在使用 useradd 和 adduser 命令创建用户时,我们需要注意哪些地方?
- (4) 请说出修改当前用户密码的详细步骤。
- (5) 如果想使用 rmdir 命令删除某个文件夹(文件夹内包含子目录与文件),该如何操作?
- (6) cat 命令有哪些功能?
- (7) less 命令和 more 命令的区别是什么?
- (8) 举例说一说 diff、diff3 和 sdiff 之间的区别。
- (9) 在使用 Ubuntu 系统时,为什么要对磁盘进行配额分配?
- (10) 用自己的话总结一下普通用户和 root 账户之间的区别。