

第 9 章 Shell 命令行操作¹

内容目录

第 9 章 Shell 命令行操作.....	1
Linux shell 简介.....	2
启动 shell.....	2
Shell 命令基本规则.....	3
一般格式.....	3
命令补全.....	3
历史命令.....	4
联机帮助.....	4
常用的 Shell 命令.....	4
目录和文件操作.....	4
用户管理.....	7
安装和卸载软件.....	7
文件的打包和解包.....	8
远程 Shell 登录.....	9
磁盘信息的低级复制.....	9
使用缩略符号.....	9
输出重定向.....	10
管道.....	10
练习.....	11
远程 Shell 服务.....	11
求赠打印件.....	14

¹本章由 UNDP-APDIP 的《User Guide to Using the Linux Desktop》Chapter 11:The Shell 翻译衍生而来的

Linux shell 简介

Linux shell 指的是一种程序，有了它，用户就能通过键盘输入指令来操作计算机了。Shell 会执行用户输入的命令，并且在显示器上显示执行结果。这种交互的全过程都是基于文本的，与其他各章介绍的图形化操作不同。这种面向命令行的用户界面被称为 CLI(Command Line interface)。在图形化用户界面(GUI)出现之前，人们一直是通过命令行界面来操作计算机的。

现在，基于图形界面的工具越来越多，许多工作都不必使用 Shell 就可以完成了。然而，专业的 Linux 用户认为 Shell 是一个非常有用的工具，学习 Linux 时一定要学习 Shell，至少要掌握一些基础知识和基本的命令

启动 shell

在启动 Linux 桌面系统后，Shell 已经在后台运行起来了，但并没有显示出来。如果想让它显示出来，按如下的组合键就可以：

`<Ctrl> + <Alt> + <F2>`

组合键中的 F2 可以替换为 F3、F4、F5、F6。

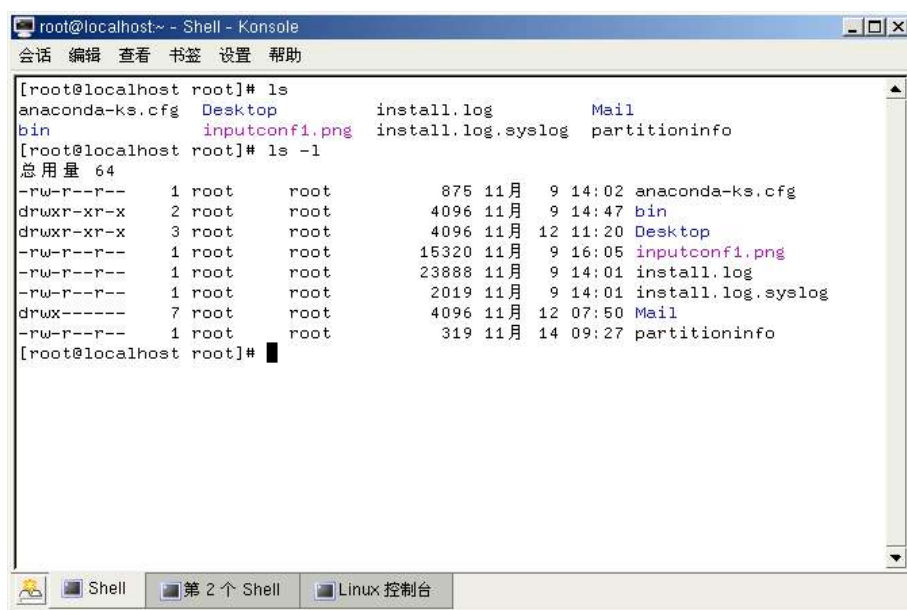
如果要回到图形界面，则按如下组合键：

`<Ctrl> + <Alt> + <F7>`

另外，在图形桌面环境下运行“系统终端”也可以执行 Shell 命令，与用组合键切换出来的命令行界面是等效的。“系统终端”启动后是一个命令行操作窗口，可以随时放大缩小，随时关闭，比较方便，推荐使用。启动“系统终端”的方法是：

【开始】→【应用程序】→【附件】→【系统终端】

如图 9—1，就是系统终端的界面。该软件允许建立多个 Shell 客户端，它们相互独立，可以通过标签在彼此之间进行切换。



```
root@localhost~ - Shell - Konsole
会话 编辑 查看 书签 设置 帮助

[root@localhost root]# ls
anaconda-ks.cfg  Desktop      install.log  Mail
bin              inputconf1.png  install.log.syslog  partitioninfo
[root@localhost root]# ls -l
总用量 64
-rw-r--r-- 1 root root      875 11月  9 14:02 anaconda-ks.cfg
drwxr-xr-x 2 root root     4096 11月  9 14:47 bin
drwxr-xr-x 3 root root     4096 11月 12 11:20 Desktop
-rw-r--r-- 1 root root    15320 11月  9 16:05 inputconf1.png
-rw-r--r-- 1 root root   23888 11月  9 14:01 install.log
-rw-r--r-- 1 root root     2019 11月  9 14:01 install.log.syslog
drwx----- 7 root root     4096 11月 12 07:50 Mail
-rw-r--r-- 1 root root      319 11月 14 09:27 partitioninfo
[root@localhost root]#
```

图 9—1. 系统终端

Shell 命令基本规则

一般格式

Shell 命令的一般格式如下：

命令名【选项】【参数1】【参数2】...

【选项】是对命令的特别定义，以减号(-)开始，多个选项可以用一个 减号(-)连起来，如 `ls -l -a` 与 `ls -la` 相同。

【参数】提供命令运行的信息，或者是命令执行过程中所使用的文件名。

使用分号(;)可以将两个命令隔开，这样可以实现一行中输入多个命令。命令的执行顺序和输入的顺序相同。

命令补全

在送入命令的任何时刻，可以按<Tab>键，当这样做时，系统将试图补全此时已输入的命令。如果已经输入的字符串不足以唯一地确定它应该使用的命令，系统将发出警告声。再次按<Tab>键，系统则会给出可用来补全的字符串清单。使用命令

补全功能，可以提高使用长命令或操作较长名字的文件或文件夹的都是非常有意义的。

历史命令

系统会把过去输入过的命令记忆下来，只要按方向键中的上下箭头，就可以选择以前输入过的命令了。

联机帮助

可以通过 man 命令来查看任何命令的联机帮助信息。它将命令名作为参数，该命令的语法格式为：

man 【命令名】

在任何命令之后给--help 参数，也可以显示该命令的帮助信息。

常用的 Shell 命令

目录和文件操作

查看当前目录相对于根目录的位置

pwd

查看当前目录内容

ls

以长格式查看当前目录内容。对应每个文件的条目将包括连接数目、所有者、大小、最后修改时间、权限等内容

ls -l

改变当前目录。目的目录名可用相对路径表示，也可以用绝对路径表示。

cd [目的目录名]

转移到上一级目录

```
cd ..
```

创建目录

```
mkdir [新目录名]
```

例：

```
mkdir /var/ftp
```

删除空目录

```
rm [目录名]
```

例：

```
rm /var/ftp
```

递归删除一个目录中的所有文件和该目录本身。

```
rm -r [目录名]
```

复制文件

```
cp [源文件名] [目标文件名]
```

重命名文件

```
mv [源文件名] [目标文件名]
```

例：

```
mv /etc/rc.d/rc3.d/K50xinetd /etc/rc.d/rc3.d/S50xinetd
```

删除文件

```
rm [文件名]
```

不需确认地删除多个文件

```
rm -f [带通配符的文件名]
```

硬链接文件。不能对目录文件做硬链接，不能在不同的文件系统之间做硬链接。

`ln [源文件名] [目标文件名]`

例：

`mv /etc/rc.d/rc3.d/K50xinetd /etc/rc.d/rc3.d/S50xinetd`

软链接文件。也就是符号链接。可用此法创建文件的快捷方式。

`ln -s [源文件或文件夹名] [目标名]`

按文件名查找文件。

`find / -name nametofind -print`

改变文件所有者。

例：

`chown workman.workgroup /data/pub/upload`

改变文件访问权限

例：

`chmod -R 755 /usr/local/LumaQQ`

查看一个文件有多少行

`wc -l usr.bin`

查看一个文件有多少字节

`wc -c usr.bin`

查看文本文件的内容

`cat usr.bin`

用户管理

改变身份。如果不指定用户名，则默认将用户身份换至 root。从 root 身份切换到其他任何身份都不需要口令。

```
su [username]
```

添加用户

例：

```
adduser -c "MyW" -d /home/workman -g root workman
```

设置用户口令

例：

```
passwd workman
```

安装和卸载软件

安装 rpm 软件

例：

```
rpm -Uvh kdevelope-3.1-1.i386.rpm
```

卸载 rpm 软件

例：

```
rpm -e kdevelope
```

配置、编译、安装、卸载源码发布的软件包。

```
./configure
```

```
make
```

```
make install
```

```
make clean
```

卸载源码发布的软件包

```
make uninstall
```

文件的打包和解包²

解 tar 包

例：

```
tar -xvzf gaim-0.77.tar.gz
```

```
tar -jxvf XXX.tar.bz2
```

生成 tar 包

例：

```
tar cvf - /etc | gzip-9c > backup.tar.gz
```

生成 rar 包³

```
rar a test.rar file1.txt
```

```
rar a test.rar dir1
```

解 rar 包

```
rar x test.rar
```

解 zip 包

```
unzip lumaqq_p3.zip
```

挂装一个 iso 文件

```
mout -t iso9660 -o loop,icharset=cp936 /data/myarc.iso /mnt/iso
```

²除 iso 文件外，其他各类型压缩包均可以在图形界面下使用“压缩工具”进行管理。从开始菜单的启动“压缩工具”的方法是：【开始】→【应用程序】→【附件】→【压缩工具】

³需要安装 RAR for Linux 才能处理 RAR 格式的压缩文件，参见“软件的安装和升级”一章。

卸装 iso 文件

```
umount /mnt/iso
```

生成一个 iso 文件

```
cp /dev/cdrom /home/gsh/TomJerry1.iso
```

远程 Shell 登录⁴

ssh 远程登录进行命令行操作

```
ssh -l workman 192.168.0.101
```

ssh 远程登录进行文件的传送

```
sftp workman@192.168.0.101
```

磁盘信息的低级复制

把 Linux 分区引导扇区的信息提取出来生成一个文件 bootsect.lnx，放到 Windows 的 C 盘。

```
dd if=/dev/hda2 of=bootsect.lnx bs=512 count=1
```

把 bootdisk.img 写到软盘上。

```
dd if=bootdisk.img of=/dev/fd0
```

使用缩略符号

波浪线 (~)代表家目录。如果现在位于/home/workman/MyOSS-Stuff/IOSN，那么只需执行“cd ~”就可以回到家目录/home/workman而不必输入“cd /home/workman”。

一个点(.)代表当前目录，而“../”代表上一级目录。“../”可以嵌套使用，比如“../..”。

⁴被登陆的目标主机应该开启安全 Shell 服务。参见“远程操作”

输出重定向

在默认的情况下，Linux 从键盘接受输入，并将命令的输出送到屏幕。在有时候，这样做并不方便。比如，在一个目录里有很多文件，如果只用简单的 `ls` 命令，在屏幕上显示的输出结果可能上千行！为了得到我们需要的信息。我们或许需要把这些结果存储到一个文件中然后再查看这个文件，这就要用到系统的输出重定向功能。

输出重定向的操作符为 `>` 或 `>>`。

单个大于号 (`>`) 后面紧跟文件名。如果指定的文件不存在，将建立这一文件。如果指定的文件存在，则文件原有的内容将被覆盖。如果使用两个大于号 (`>>`) 则会把输出内容追加到原来文件里面。下面是一个例子：

```
1.[-(/tmp)> ls /usr/bin >> usr.bin
```

```
2.[-(/tmp)> wc -l usr.bin
```

```
3.2171 usr.bin
```

```
4.[-(/tmp)> ls /usr/bin >> usr.bin
```

```
5.[-(/tmp)> wc -l usr.bin
```

```
6.4342 usr.bin
```

```
7.[-(/tmp)> ls /usr/bin > usr.bin
```

```
8.[-(/tmp)> wc -l usr.bin
```

```
9.2171 usr.bin
```

注：行号是为了行文方便加入的，并不是 shell 的输出。

第 1 行，`/usr/bin` 中的文件列表被写入文件 `usr.bin`，由于这个文件原来并不存在，系统会自动创建。从第 3 行可以看出 `usr.bin` 的行数。在第 4 行又重复了第 1 行的命令，由于使用的是 `>>`，`usr/bin` 中的文件列表被追加写入文件，`usr.bin` 的行数也就增加了一倍。在第 7 行我们使用了 `>`，`usr/bin` 中的文件列表被写入文件，`usr.bin` 中原有的内容被覆盖了，行数也变成了新的值。

管道

将一个程序的标准输出写道一个文件中，再将这个文件的内容作为另一个命令的标准输入，等效于通过临时文件将两个命令结合起来。这种情况很普遍，需要 Linux 系统提供一种功能：它不需要或不使用临时文件，就能将两条命令结合在一起。这种功能就是管道。

管道的操作符是一个竖杠 “|”。管道是可以嵌套使用的，因此可以把多个命令结合在一起。

接上例，如果执行下面的命令将直接返回/usr/bin 中的文件列表的行数，而不是列表的内容。

```
ls /usr/bin | wc -l
```

练习

从桌面上启动 shell 终端，执行如下操作：

- 1.查看当前所在的目录
- 2.显示当前目录中的内容
- 3.显示/usr/bin 目录中的内容
- 4.转移到家目录，并创建一个子目录，命名为 Temp11，然后转移到该目录中
- 5.从/etc 中复制如下目录到 Temp11 目录中： services, motd, fstab, hosts
- 6.查看 services 内容的行数
- 7.删除 Temp11 中列出的四个文件

远程 Shell 服务

支持远程操作是 Linux 的一个非常重要的特点。利用此功能，用户从另一台计算机⁵远程登录上来，进行 Shell 命令的操作。

红旗 Linux4.1 桌面版已经安装了支持远程操作的安全 Shell 服务软件 OpenSSH。在默认情况下，此软件并没有运行起来。可以通过服务控制启动 OpenSSH 来支持远程操作。下面介绍一下操作方法。

首先，从开始菜单启动服务管理工具，方法是：

【开始】→【设置】→【控制面板】→【服务】

从服务管理工具中找到 sshd(安全 Shell 服务)，如图 9-2。选中它，从操作菜单中就可以启动安全 Shell 服务了：

【操作】→【启动】

如果希望，每次启动计算机时，自动启动安全 Shell 服务，可以选中它，修改它的属性：

⁵可以是运行 Linux 操作系统的计算机，也可以是运行 Windows 操作系统的计算机

【操作】→【属性】

修改属性的界面如图 9-3，把启动类别改成自动即可。

提示：如果想用一台装有 windows 系统的电脑进行远程操作，要在这台电脑上安装 ssh 客户端程序。可以使用运行于 Windows 上的专用客户端程序 SSHSecureShellClient。另外使用 putty 等通用的 ssh 客户端软件也能进行远程操作。



图 9-2 服务管理工具



图 9—3 服务的属性

求赠打印件

如果在使用本章时恰好把它打印了出来(打印时请选用 B5 纸型),那么在用完之后而打算把它当成废纸扔掉之前,请考虑将其赠给本书作者。设想,如果读者们都能把自己不再需要的打印件赠送给作者,那么就有可能汇编成册;如果能汇编成册,用处肯定会很大。请授权作者自由处置。

截止日期:2005 年 11 月 30 日。过了这个时间,就不要再寄了。

在邮寄前,不要忘记签名。如果对 Linux 的应用和普及有所感悟,请作简短留言。

简短留言:

赠送者签名:

日期:

——剪切线 ————下面是作者的通信地址。由此剪下,贴到信封上寄出即可——剪切线 ——

150080

黑龙江省哈尔滨市南岗区延兴路 45 号 518 信箱

(黑龙江省通信技术支持中心 交换部)

郭守华 收