



普通高等教育“十一五”国家级规划教材

边干边学 (第二版)

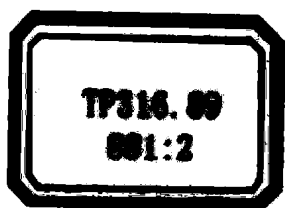
—— LINUX 内核指导

LEARNING LINUX KERNEL BY PRACTICE

李善平 季江民 尹康凯 等 编著
胡志刚 主审



ZHEJIANG UNIVERSITY PRESS
浙江大学出版社



Learning Linux Kernel by Practice

边 干 边 学

——Linux 内核指导

(第二版)

李善平 季江民 尹康凯 等 编著

胡志刚 主审



ZHEJIANG UNIVERSITY PRESS

浙江大學出版社

图书在版编目 (CIP) 数据

边干边学: LINUX 内核指导 / 李善平, 季江民, 尹康凯等编著.
—杭州: 浙江大学出版社, 2002.8 (2008 重印)
ISBN 978-7-308-03073-1

I. 边… II. ①李…②季… III. Linux 操作系统—基本知识
IV.TP316.89

中国版本图书馆 CIP 数据核字 (2002) 第 051037 号

边干边学——LINUX 内核指导

李善平 季江民 尹康凯 等 编著

责任编辑: 杜希武

封面设计: 刘依群

出版发行: 浙江大学出版社

(杭州天目山路 148 号 邮政编码: 310028)

(Email: zupress@mail.hz.zj.cn)

(网址: <http://www.zjupress.com>

<http://www.press.zju.edu.cn>)

电话: 0571—88925592, 88273066 (传真)

排 版: 浙江大学出版社电脑排版中心

印 刷: 杭州浙大同心教育彩印有限公司

印 张: 41.75

开 本: 787mm×1092mm 1/16

字 数: 1016 千

版 印 次: 2002 年 8 月第 1 版 2008 年 4 月第 2 版 2008 年 4 月第 7 次印刷

印 数: 14001—16000

书 号: ISBN 978-7-308-03073-1

定 价: 79.00 元

版权所有 翻印必究 印装差错 负责调换

浙江大学出版社发行部邮购电话 (0571) 88072522

内容简介

这是一本学习 Linux 内核的指导教材。根据作者在浙江大学计算机学院多年的教学实践,教材内容选定了 Linux 环境, shell 编程, 时钟与定时器, 系统调用, 进程创建, proc 文件系统, 进程同步, 内存管理, 共享内存, 文件系统, 内核模块等 18 章内容, 其相关源代码贯穿整个 Linux 内核。

学习操作系统必须动手实践。本书的特点就是边干边学。为此在每章里都安排了实验内容, 章节围绕实验目的展开, 以求在实验中掌握 Linux 内核, 在实验中融会贯通。

本书可以作为计算机专业本科生、研究生的操作系统实验教材, 也可以作为 Linux 系统管理员, 嵌入式系统研发人员的参考书。

前 言

学计算机专业知识必须动手实践，这点应该不会有异议！

那么学操作系统怎么实践呢？尤其是我们近年来承担了浙江大学计算机学院操作系统类课程的教学，更需认真回答这个问题了。

首先，选择实验对象和环境。这一点已经找到答案了，那就是 Linux 操作系统内核。我们在 Linux 内核分析方面小有成就，也编写了几本教材。

其次，搜集指导材料。我们自己在学习 Linux 过程中，耗费了许多精力和时间搜集资料，又耗费了许多精力和时间分析、整理这些资料。如果你不是天才，同样也要耗费精力和时间再去做这些。何必呢！我们共享吧。

对了！这就是我们费劲写这本书的起因。我们希望奉上这些心得，帮助你在学习 Linux 内核时事半功倍。

本书的第一版自 2002 年出版后，得到了读者的大量反馈，在此表示衷心感谢！但是，毕竟过去了 5 年，Linux 内核、Linux 世界变化巨大，早该出第二版了。

现在你看到的第二版，主要有三个方面的更新：

- 第一版里的错误得到纠正；读者的意见得以体现。
- 第一版针对当时流行的内核版本 2.4.18。这次的第二版则针对内核版本 2.6.15。2.6 版相对于 2.4 版，内核的变化是根本性的。

● 为照顾许多读者在 Linux/Unix 基础知识方面的欠缺，特别增加了 8 章关于 Linux 操作环境的原理和实验（统一编入第一部分）。原有的 Linux 内核方面的内容，全部在第二部分。所以，如果你了解 Linux 常识，那么，直接进入第二部分学习吧！

另外，你会发现，第二版在格式上也有些调整，希望方便你阅读。而且我们在许多章的末尾，列出了参考文献，给学有余力的读者加点料。当然，不用说的是，最主要、最重要的参考资料还是内核源代码本身。

本书第一版当时由李善平、陈文智共同负责，尹康凯等 10 余位浙江大学计算机学院的研究生一起参加编写。

第二版里，尹康凯编写第 11，12，15，16，17，18 章，季江民编写第 1，2，3，4，5，6，7，8 章，李善平编写第 9，10，13，14 章。

一如既往地，我们欢迎你的宝贵意见。

目 录

第一部分 Linux 操作环境

第 1 章 Linux 基础	1
1.1 登录和退出	1
1.2 Linux 的 shell	4
1.3 shell 的一些基本命令	9
第 2 章 文本编辑	20
2.1 vi 文本编辑器	20
2.2 emacs 文本编辑器	25
第 3 章 Linux 文件系统操作	29
3.1 文件类型	29
3.2 文件系统目录结构	31
3.3 目录操作的基本命令	35
3.4 文件操作的基本命令	39
3.5 显示字符串 echo	46
3.6 命令行中使用扩展符	46
第 4 章 文件权限与文件共享	52
4.1 存取权限	52
4.2 改变文件的存取权限	55
4.3 特殊权限位 SUID、SGID、Sticky	57
4.4 硬链接	58
4.5 符号链接	60
第 5 章 文件管理工具	63
5.1 正则表达式	63
5.2 排序文件	65
5.3 查找文件	66
5.4 搜索文件内容	68
5.5 命令记录	69

5.6 压缩文件、解压缩文件与打包文件	70
5.7 RPM 包管理	72
第 6 章 Linux 进程、管道和重定向	75
6.1 进程的属性	76
6.2 进程的终止	78
6.3 进程和作业控制	79
6.4 命令行中使用操作符	82
6.5 Linux 系统启动和进程层次结构	83
6.6 系统启动和关机	84
6.7 输入、输出重定向	85
6.8 使用文件描述符	86
6.9 输入和输出重定向的组合使用	86
6.10 管道 (pipe)	87
第 7 章 C 语言开发工具	90
7.1 编写程序的工具	90
7.2 编译 C 语言程序	92
7.3 make 工具	96
7.4 gdb 调试工具	100
第 8 章 Bourne Again Shell 编程	106
8.1 bash 脚本的建立和运行	107
8.2 shell 的变量	108
8.3 shell 脚本位置参数的传递	113
8.4 控制结构语句	115
8.5 其他几个有用的语句	123
8.6 数值处理	126
8.7 数组	130
8.8 函数	131
8.9 here 文档	134
8.10 exec 命令	136
8.11 trap 命令	139
8.12 调试脚本程序	141

第二部分 Linux 内核分析与实践

第 9 章 编译 Linux 内核	143
9.1 Linux 内核	143
9.2 查找并且下载一份内核源代码	146
9.3 部署内核源代码	147
9.4 配置内核	147
9.5 编译内核和模块	148
9.6 了解 Linux 内核的启动	149
9.7 应用 grub 配置启动文件	151
9.8 编写制作 Linux 启动盘的 shell 脚本程序	153
9.9 Linux 源程序的目录分布	156
9.10 学习 Linux 的常用工具	159
9.11 查看 Linux 内核状况	162
9.12 编程序检查系统状况	163
9.13 Linux 编程环境	167
第 10 章 系统调用	174
10.1 一个简单的例子	174
10.2 系统调用基础知识	175
第 11 章 进程创建	211
11.1 进程是什么	211
11.2 进程的产生	229
11.3 进程的消亡 / 退出	262
11.4 实验一：用 fork（）系统调用	264
11.5 实验二：使用 clone（）系统调用	266
11.6 实验三：使用 kernel thread	269
第 12 章 /proc 文件系统	279
12.1 /proc 文件系统	279
12.2 现有 proc 文件系统中各个文件的含义	280
12.3 怎样使用/proc 文件系统	285
12.4 seq_file	289
12.5 proc 文件系统的内部实现机制	293
12.6 实验一：使用 proc 文件系统的一个简单例子	325

12.7 实验二：利用/proc 文件系统显示缺页状态	330
12.8 实验三：seq_file 使用例子	330
第 13 章 内核模块	337
13.1 什么是内核模块	337
13.2 模块实现机制	340
13.3 使用内核模块	350
13.4 实例	352
第 14 章 内存管理	356
14.1 虚拟内存管理	356
14.2 Linux 虚拟内存管理	363
14.3 实例	391
14.4 综合实验的原理	394
14.5 综合实验的实施	395
第 15 章 内核时钟与定时器	400
15.1 关于时钟和定时器	400
15.2 Linux 系统时钟	405
15.3 Linux 系统定时器	421
15.4 时钟命令	432
15.5 实验一：一个应用定时器的简单例子	435
15.6 实验二：统计关于进程的时间	437
15.7 实验三：更进一步的进程时间统计	440
第 16 章 共享内存	450
16.1 进程间通信和共享内存	450
16.2 共享内存的 API	452
16.3 共享内存存在 Linux 中的实现	463
第 17 章 同步机制	494
17.1 同步机制	494
17.2 Linux 中几种同步机制的实现	508
17.3 设计我们自己的同步机制	553
第 18 章 文件系统	561
18.1 文件系统基本概念	561
18.2 文件系统的抽象	565
18.3 VFS 文件系统	566
18.4 ext2 文件系统	600

18.5 对文件的操作	608
18.6 块读写与页缓存	624
18.7 本章总结	632
18.8 实验：添加一个文件系统	632
18.9 附录：优秀的日志文件系统——ext3	648

第1章 Linux 基础

实验目的

- 学习登录和退出 Linux 系统。
- 了解 Linux shell。
- 熟悉 Linux 系统的元字符。
- 初步了解 Linux 的命令格式。
- 学会如何得到帮助信息。
- 掌握一些常用的 Linux 命令。

实验指导

Linux 操作系统是目前最流行的操作系统之一。作为 Internet 的产物, Linux 操作系统是由全世界许多人共同合作开发的, 它是一个自由 (free) 的操作系统。

shell 是用户和操作系统之间接口的一个程序。shell 的主要功能是解释命令, shell 也被称为 Linux 的命令解释器 (command interpreter)。在各种各样的 shell 中, Bourne Again shell (bash), TC shell (tcsh) 和 Korn shell (ksh) 是最为流行的。

shell 命令可以被分为内部 (内置) 命令和外部命令。内部命令是其程序代码包含在 shell 程序中的一些命令, 这些内部命令的代码是整个 shell 代码的一个组成部分; 而外部命令的代码是存放在磁盘上的二进制可执行文件或者 shell 脚本文件。

本章我们简要描述 Linux 操作系统的结构和主要组成部分。然后我们介绍登录和退出过程。介绍 Linux shell, shell 的元字符, 内部命令和外部命令。并且介绍一些简单常用的 Linux 命令。

1.1 登录和退出

Linux 系统是多用户、多任务和交互式的计算环境。多任务表示计算机系统同时可以运行多个计算进程或程序。多用户表示多个用户可以同时使用同一个计算机系统。这就说明登录操作和退出操作都是必要的, 每个使用计算机系统的用户需要使用各自的用户名和密码进入到 Linux 系统。

有基于文本界面的登录和退出 Linux 系统的方式及基于图形用户界面 (GUI) 的登录和退出 Linux 系统的方式。

基于文本界面方式登录后, 进入到文本界面 (也称命令行方式), 这是 Unix 传统界面, 这种方式下只能使用键盘操作, 不能使用鼠标。在基于图形用户界面 (GUI) 的登录后,

你可以使用鼠标操作。

1.1.1 登录 Linux 系统

1. 基于文本界面的登录

有多种登录到 Linux 系统的方法。在 WINDOWS 操作系统中通过因特网连接到 Linux 系统和通过局域网连接到 Linux 系统；如果你的计算机上已安装好 Linux 系统，就可以很方便地使用文本界面登录了。

(1) 通过因特网连接

这种方式连接，通常是在 WINDOWS 操作系统下连接到 Linux 操作系统。一般使用远程登录软件连接到因特网的开放 Linux 服务器上。因特网中的开放 Linux 服务器很多，如：lab.lupa.cn。在 WINDOWS 操作系统下使用的远程登录软件连接 Linux 服务器，远程登录软件有各种版本软件，如 SSH、telnet、putty 等。你可以在 <http://www.ssh.com/> 下载 SSH 软件。安装 SSH 软件后，启动“SSH Secure Shell Client”程序，选择 Edit→Settings 菜单，输入 Linux 服务器的 ip 地址或主机名及用户名。SSH 的设置为：Host 为 lab.lupa.cn，User 为 lab，Port 为 22，如图 1-1 所示。对于一个 Linux 服务器，SSH 设置一次就可以了。

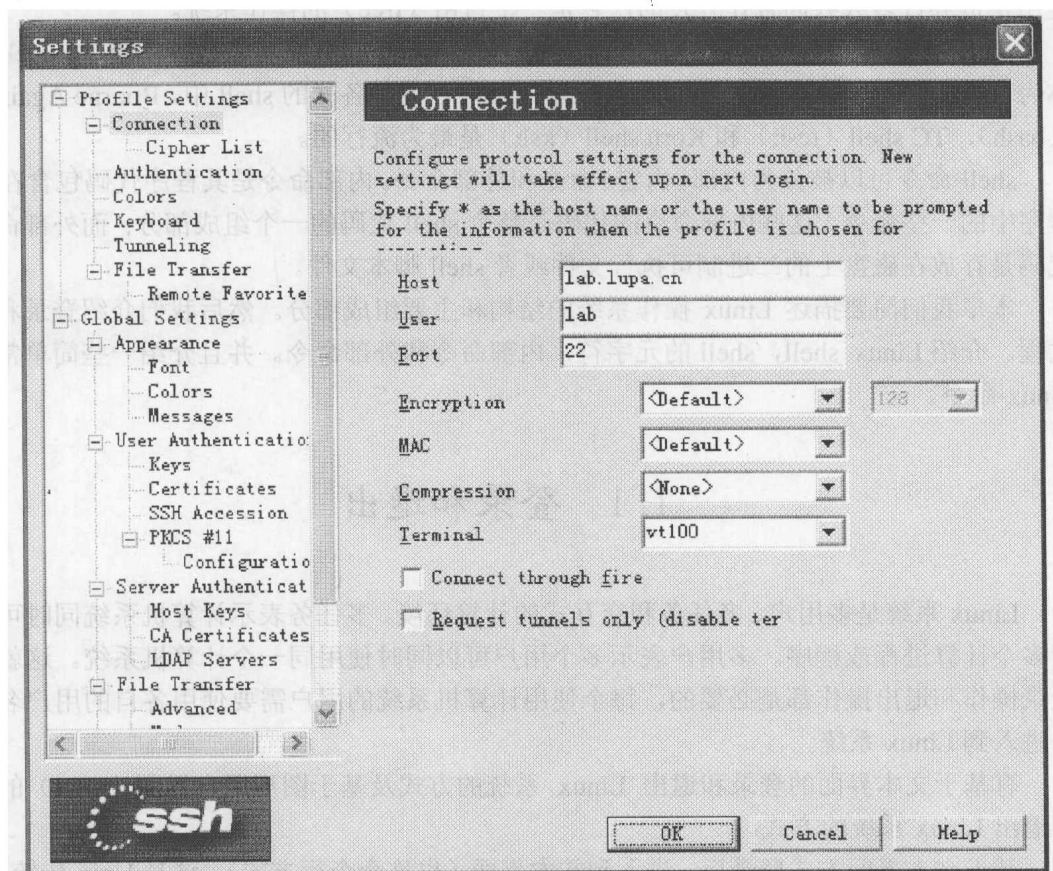


图 1-1 设置 SSH 连接参数

SSH 设置完成后，你就可以连接 Linux 服务器了。每一次连接 Linux 服务器，只要这样操作就可以了：选择 File→Quick Connect 菜单；屏幕显示“Connect to Remote Host”对话框后，单击“Connect”按钮；最后输入登录密码，密码检查通过后，屏幕显示如图 1-2 所示。这样你就可以使用 Linux 系统的各种命令了。

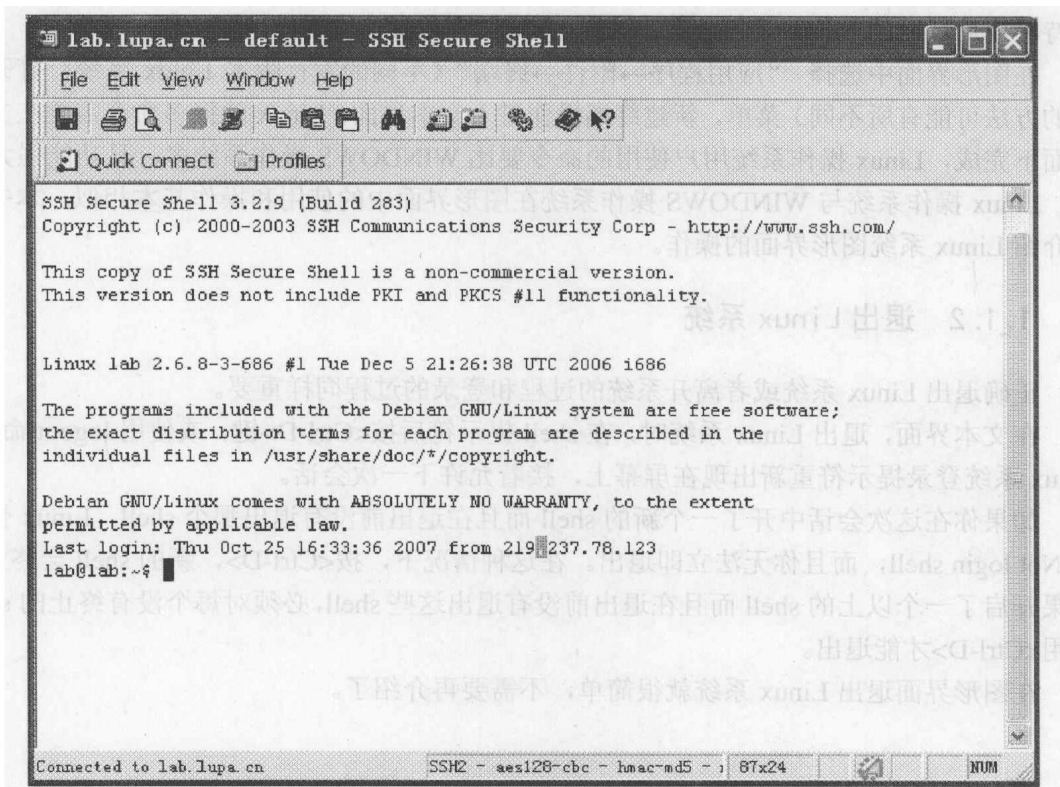


图 1-2 登录 Linux 系统

(2) 局域网连接

局域网连接方法与因特网相似，不同的是你所连接的 Linux 服务器在你计算机系统的局域网中。

(3) 独立连接

你使用的是一台装有 Linux 系统的计算机，这台计算机的 Linux 系统没有安装 GNOME 或 KDE 等桌面系统。在系统启动后，需要向系统输入正确的用户名和登录密码。

Login : your username

Password: your password

在成功登录到 Linux 系统后，屏幕会出现一个诸如“\$”（作者的 fedora core 的命令提示符是“\$”，你的 Linux 系统有可能是其他的符号）的 shell 提示符。接着用户可以使用各种各样的 Linux 命令了。

2. 基于图形用户界面

你使用的计算机系统是安装了 GNOME 或 KDE 图形桌面的 Linux 系统, 可以使用图形界面的登录窗口登录到 Linux 系统中。使用图形界面登录也需要输入用户名和密码。

在 WINDOWS 操作系统下, 你可以使用 Xmanager 或 X-Win32 等客户端软件, 连接到因特网或局域网的 Linux 服务器上, 这样也可以使用 Linux 的图形界面。

在图形界面中选择 “应用程序→附件→终端”(不同的发行版本 Linux 系统, 打开终端的方法可能有所不同) 菜单, 新建终端操作进入文本界面。Linux 的很多命令需要在文本界面下完成, Linux 操作系统用户使用的命令要比 WINDOWS 操作系统多, 且功能强大。

Linux 操作系统与 WINDOWS 操作系统在图形界面中的使用和操作基本相似, 本书不再介绍 Linux 系统图形界面的操作。

1.1.2 退出 Linux 系统

正确退出 Linux 系统或者离开系统的过程和登录的过程同样重要。

在文本界面, 退出 Linux 系统时, 在 shell 提示符后按<Ctrl-D>键, 或使用 logout 命令。Linux 系统登录提示符重新出现在屏幕上, 接着允许下一次会话。

如果你在这次会话中开了一个新的 shell 而且在退出前没有退出那个 shell, Linux 会提示 Not login shell, 而且你无法立即退出。在这种情况下, 按<Ctrl-D>, 新的 shell 会终止。如果开启了一个以上的 shell 而且在退出前没有退出这些 shell, 必须对每个没有终止的 shell 使用<Ctrl-D>才能退出。

在图形界面退出 Linux 系统很简单, 不需要再介绍了。

1.2 Linux 的 shell

每个 Linux 系统发行版本中都包含了多种 shell。目前使用的最为广泛的 shell 包括 bash, TC shell 和 Korn shell。你进入系统后开始运行的那种 shell 就是你的登录 shell (login shell)。通常默认的登录 shell 是 bash。系统管理员可以为你指定使用哪种 shell 作为登录 shell。当然你也可以通过命令来改变自己的默认登录 shell。比如说, 如果你的默认登录 shell 是 bash, 但是你更喜欢用 TC shell, 你就可以通过命令 tcsh 或者 chsh 来改变默认登录 shell。

1.2.1 常用 shell 程序

Linux 提供多种 shell 程序, 用户可以根据 shell 的功能、特性来选择符合自己使用习惯的 shell。各种发行版本的 Linux 系统中并不一定把所有的 shell 都安装在系统中, 作者使用的 fedora core 系统中最常用的几种 shell 如表 1-1 中所示。各种 shell 程序放在/bin/目录下。

表 1-1 常用 shell 程序

shell 名称	存放的位置	程序名
Bourne shell	/bin/sh->bash	bash
Bourne Again shell	/bin/bash	bash
C shell	/bin/csh->tcsh	tcsh
TC shell	/bin/tcsh	tcsh
Korn shell	/bin/ksh	ksh

Bourne shell 是标准的 UNIX shell，它是由 AT&T（贝尔实验室）的 Steve Bourne 开发并以之命名。它的特点是简单、高效，但缺乏交互性的功能（如别名、任务控制等）。

Bourne-again shell 是 GNU shell，所有 Linux 系统都提供这种 shell。它遵循 POSIX，POSIX 指的是可移植的操作系统接口（Portable Operating System Interface），POSIX 是一系列由 IEEE 制定的标准。Bourne-again shell 同时也保留了与 Bourne shell 的兼容性。在 Linux 这样的基于 GNU 的系统中取代了 Bourne Shell。Linux 的默认 shell 是 Bourne-again shell。事实上，/bin/sh 是链接到/bin/bash。

C shell 由 Bill Joy 在加州大学伯克利分校开发，所有 BSDUNIX 版本都提供这种 shell。另外，AT&T 的系统 V/386 R3.2 和 SVR4 也提供 C shell。C shell 是在第六版 shell 而非 Bourne shell 的基础上构造的。其控制流类似于 C 语言，它支持 Bourne shell 没有的某些特色功能，例如作业控制、历史记忆机制以及命令行编辑等。在 Linux 系统中与 C shell 对应的是 TC shell。

Korn shell 由 AT&T 的 David Korn 在 80 年代中期开发并成为 SVR4 UNIX 的正式部分。它实际上是 Bourne shell 的一个超集，增加了一些新的特性并与 Bourne shell 相兼容。

1.2.2 改变 shell 的方法

系统管理员在建立用户账户时，将确定你第一次登录系统或打开 GUI 环境下模拟终端窗口时使用的 shell，你的账户可能使用 bash 作为登录 shell。你可以自己决定运行哪个 shell。可以通过三种不同方法更改 shell：

- (1) 为以后登录的所有会话更改默认 shell；
- (2) 创建一个运行在默认 shell 之上，或者和默认 shell 并行运行的新的 shell 会话；
- (3) 只改变当前会话的 shell。如在 bash 下运行 tcsh。

命令 chsh 能够改变用户默认的登录 shell。带选项 chsh -l 命令可以显示系统中有哪些可用的 shell。通过输入想要使用的 shell 的完整的路径，就可以完成默认 shell 的更改。比如，在出现提示后，输入/bin/tcsh 就可以把默认 shell 改为 tcsh。

如果想在默认 shell 上运行另一个新的 shell，只需要在命令行输入 shell 程序的名字（如表 1-1 所示）。在下面的例子中，默认 shell 是以 \$ 为提示符的 Bourne • again shell，环境变量 SHELL 中存放当前 shell 的程序路径。运行 tcsh 程序后把当前 shell 改变为 TC shell，TC shell 提示符为 %。

```
$ echo $SHELL
/bin/bash
$ tcsh
```

```
% ps
PID    TTY    TIME    CMD
899    pts/0  00:00:01  bash
1515   pts/0  00:00:00  tcsh
1535   pts/0  00:00:00  ps
```

要中止或者离开这个新的临时的 shell，返回到默认登录 shell，输入<Ctrl-D>命令。如果这种方法不起作用，你还可以通过输入 exit 命令返回到上一个 shell。

只改变当前登录会话的 shell：若你要在 bash 下运行 TC shell 的脚本程序 cshpr，可以在 bash 提示符下输入 tcsh cshpr。

1.2.3 shell 的启动文件

当启动 shell 时，它将运行启动文件初始化自己。每个 Linux 系统都有一个系统启动文件，通常是/etc/profile。这个文件中包含了 shell 和其他一些实用工具的重要环境变量的初始化设置。而且，对于某些 shell，系统中还有一些隐藏文件，或者叫做点(.)文件，这种文件的文件名的首个字符是小数点。在你运行相应的 shell 时，这些文件会被执行。这些被称为 shell 启动文件的文件通常存放在主目录（用~表示）中，比如~/.profile，或者以特定 shell 的 profile 和登录文件的形式存放在用户的主目录中。比如 bash 的 profile 和登录文件通常被命名为~/.bash_profile 和~/.bash_login。这些隐藏文件最初是由系统管理员根据用户安全使用系统的要求进行配置的。

如果登录 shell 是 bash，当你登录时，如果系统中有/etc/profile，这个文件中的命令首先被执行。然后按照~/.bash_profile，~/.bash_login 或者~/.profile 的顺序搜索并执行第一个被找到而且可读的文件。当你注销时，bash 执行文件~/.bash_logout 中的命令。

如果你启动一个交互的 bash shell，~/.bashrc 文件存在且可读，这个文件就会被执行。如果以非交互的形式运行一个 shell 脚本，bash 会搜索环境变量 BASH_ENV 来确定将要执行的文件的名字。

尽管有很多种启动文件和 shell，但用户通常只需要主目录下的.bash_profile 和.bashrc 文件。.bash_profile 中类似下面的命令将为登录 shell 时执行.bashrc（如果该文件存在）中的命令。

```
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi
```

条件[-f ~/.bashrc]测试主目录下是否存在名为.bashrc 的文件。关于[]以及与之同义 test，我们将在第 9 章中详细介绍。

你可以在.bash_profile 设置 PATH，PATH 为环境变量，该变量存放程序或命令的所有目录，在运行程序或命令时要根据这些目录进行搜索。因为每次执行 bash 程序都要运行.bashrc 中的命令，因此.bashrc 中的命令可能执行多次，并且因为子 shell 继承那些被输出的变量，所以最好将那些附加已有变量的命令放置到.bash_profile 文件中。例如，下面的命令将你的主目录的子目录 bin 添加到 PATH 中，应该将其放置到.bash_profile 文件中：

```
PATH=$PATH:$HOME/bin
```

1.2.4 shell 的环境变量

shell 环境变量具有特殊的意义，它们的名字一般比较短，bash 的环境变量名通常由大写英文字母组成。当用户启动 shell 的时候（比如登录），shell 将从环境中继承几个环境变量。HOME 和 PATH 就属于这样的变量。其中，HOME 表示用户的主目录。而当用户输入命令时，PATH 决定了 shell 在哪些目录下搜索该命令，同时还决定了搜索时的顺序。当 shell 启动之后，它再创建和（用默认值）初始化其他环境变量。而对于其他变量，除非用户设置，否则它们都是不存在的。

用户在任何时候都可以更改大多数 shell 环境变量的值，但通常没有必要修改系统初始化文件/etc/profile 和/etc/csh.cshrc 中初始化的环境变量的值。如果需要修改 bash 环境变量的值，就在初始化文件中进行修改。用户可以将用户创建的变量变成全局变量，同样，可以将环境变量变成全局变量，这个工作一般在初始化文件中自动完成。用户还可以将环境变量变为只读。

在下面的这个例子中，我们在搜索路径中增加了两个目录，~/bin 和（“.”表示当前目录）。而且使~/bin 成为最先被搜索的目录，而当前目录则成为最后被搜索的目录。

```
# PATH=~/bin:$PATH?
```

我们在表 1-2 中列出了部分环境变量，更多的环境变量将在第 9 章中给出。

表 1-2 部分 bash 环境变量

环境变量名	含义
CDPATH	cd 命令访问的目录的别名
EDITOR	用户在程序中使用的默认的编辑器
ENV	Linux 查找配置文件的路径
HOME	主目录的名字
PATH	存放搜索命令或者程序的所有目录
PS1	shell 提示符
PS2	shell 的二级提示符
PWD	当前工作目录的名字
TERM	用户使用的控制台终端的类型

1.2.5 shell 元字符

除了字母和数字，很多其他字符对于 shell 都有特殊的含义。这些字符被称为 shell 元字符（shell metacharacters），也称为特殊字符。如果不以特殊方式指明，在 shell 命令中，这些字符不能作为文本字符使用。所以，不要在文件名中使用这些字符。而且在命令中使用这些字符时，不需要在它们的前面或者后面加上空格。当然，为了清楚起见，你也可以在这些元字符的前面或者后面加上空格。表 1-3 列出了 shell 元字符及其作用。