



红旗 Linux 系统 应用编程

北京中科红旗软件技术有限公司 编著

石油工业出版社

红旗 Linux 系统应用编程

北京中科红旗软件技术有限公司 编著

石油工业出版社

内 容 提 要

本书主要介绍了在 Linux 系统平台上的程序开发技术, 内容涉及 Linux 系统的基本开发环境、基础开发技术、Qt 编程、脚本编程, 以及代码的组织、管理与发布等方面。本书不是编程语言和软件设计原理丛书, 而是针对已经具备软件编程基本知识的人员, 用于学习和掌握在 Linux 系统平台的编程技术与开发经验, 重点面向应用类开发领域的技术需求。

本书主要面向软件产业领域的技术人员和学生。随着“核心电子器件、高端通用芯片及基础软件产品”国家重大专项的开展, 以及移动互联网应用的普及, 基于 Linux 系统平台的软件产业也将迅猛发展, 本书将帮助在此领域的软件开发人员和学生, 尽快熟悉和掌握 Linux 系统上的软件开发技术。

图书在版编目 (CIP) 数据

红旗 Linux 系统应用编程/北京中科红旗软件技术有限公司编著.
北京: 石油工业出版社, 2012. 12
ISBN 978 - 7 - 5021 - 9389 - 8

I. 红…

II. 北…

III. Linux 操作系统-程序设计

IV. TP316. 89

中国版本图书馆 CIP 数据核字 (2012) 第 293276 号

出版发行: 石油工业出版社

(北京安定门外安华里 2 区 1 号 100011)

网 址: www.petropub.com.cn

编辑部: (010) 64255590 发行部: (010) 64523620

经 销: 全国新华书店

印 刷: 北京中石油彩色印刷有限责任公司

2012 年 12 月第 1 版 2012 年 12 月第 1 次印刷

787 × 1092 毫米 开本: 1/16 印张: 18.25

字数: 467 千字

定价: 48.00 元

(如出现印装质量问题, 我社发行部负责调换)

版权所有, 翻印必究

前 言

Linux 系统是运行在个人计算机和工作站上的 Unix 操作系统，它继承了 Unix 的全部优点，是真正的多用户、多任务、多平台的操作系统。在个人计算机和工作站上使用 Linux 系统，能充分发挥硬件的功能，使个人计算机能够作为工作站和服务器使用，提高工作站效率；Linux 系统符合 POSIX 标准，能同时支持 i386、MIPS、SPARC 和 PowerPC 等处理器；对于 Unix 系统上运行的软件，不用改动或稍加改动就可以运行在 Linux 操作系统上。

同 Unix 系统相比，Linux 系统要灵活得多。Linux 系统的用户界面非常友好，用户使用很方便。基于开源文化技术理念的开发与交流方式，使得众多经验与知识可以得到充分的交流。Linux 系统及其众多应用程序的源代码都是向用户公开的，允许用户使用 Linux 系统自带的 C/C++ 语言编译器修改并编译自己喜欢的程序，同时也可以 GNU 的许可下使用这些源程序的部分或全部；这种利用 Linux 源程序的可能性大大激发了世界范围内热衷于计算机事业的人们的兴趣和创造力，吸引了众多程序员为 Linux 系统工作。在基于不同文化背景需求的 Linux 系统发布版中，同时集成了众多编程语言的编译程序、数据库管理系统、图形用户界面、通信联网工具以及大量的其他应用软件。可见，Linux 系统已经是一个相当优秀的应用软件开发平台。

此外，强大的网络与集群功能，使 Linux 系统在目前基于云计算的应用架构中起到了重要的作用。因此，Linux 系统也将激发基于集群服务架构的网络应用开发技术的运用。未来的 Linux 系统应用将不再是以往单机运行的主导模式，基于强大高速网络平台的应用技术将会快速发展。

为适应软件开发产业的变革以及促进 Linux 系统在我国推广应用，针对院校人才培养市场的需要，并配合“国家 Linux 技术培训与推广中心”建设以及“核心电子器件、高端通用芯片及基础软件产品”国家重大专项的需要，红旗教育学院基于红旗软件研发体系的工作经验，组织经验丰富的 Linux 编程人员，参

考国际经典 Linux、Unix 编程书籍，编写了本书，为开源文化理念下的软件技术开发教学提供经验与技术的交流。

本书由北京中科红旗软件技术有限公司红旗教育学院组织编写，对参与本书编写的人员，在此表示深深的感谢。由于本书涉及的内容丰富，加之篇幅、时间有限，书中难免存在纰漏和不足，失误之处敬请谅解和指正。

北京中科红旗软件技术有限公司

2012. 7

目 录

1 基本开发环境	1
1.1 Linux 编程综述	1
1.1.1 Linux 编程特点	1
1.1.2 Linux 编程与 Windows 编程的主要区别	1
1.1.3 Linux 编程与 Unix 编程的主要区别	2
1.2 Linux 编程环境	2
1.2.1 编程系统安装	2
1.2.2 Linux 系统文件分布和组织	5
1.2.3 Linux 系统命令操控技术	7
1.2.4 文本编辑器使用技术	18
1.2.5 源程序代码的获取与阅读方法	27
练习思考题	27
2 基础开发技术	29
2.1 编译和链接	29
2.1.1 GNU 编译器简介	29
2.1.2 GCC 编译过程	29
2.1.3 GCC 警告选项	36
2.1.4 GCC 优化选项	38
2.1.5 GCC 调试选项	41
练习思考题	41
参考资料	41
2.2 程序库的使用和开发	41
2.2.1 程序库历史	41
2.2.2 静态库	42
2.2.3 动态库	43
2.2.4 常用 C 语言库	48
练习思考题	51
参考资料	51
2.3 程序调试	51
2.3.1 基本 GDB 命令	52

2.3.2 高级 GDB 命令	62
2.3.3 图形化调试工具	67
练习思考题	68
2.4 进程和线程	68
2.4.1 Linux 进程模型	68
2.4.2 进程创建和退出	73
2.4.3 信号	78
2.4.4 进程间通信	82
2.4.5 线程	87
练习思考题	93
2.5 内存管理	93
2.5.1 进程虚拟地址空间	93
2.5.2 内存的分配、释放和映射	96
练习思考题	99
3 Linux 下的 Qt 编程	100
3.1 Qt 编程综述	100
3.1.1 XWindow 编程综述和 Qt 的基本概念	100
3.1.2 Qt 简介	100
3.1.3 “Hello world!” in Qt	101
练习思考题	107
3.1.4 Qt 库的类层次结构	107
3.2 Qt 的基础类	109
3.2.1 QString	109
练习思考题	112
3.2.2 QWidget	112
练习思考题	121
3.2.3 QApplication 和 Qt 的国际化	121
练习思考题	129
小结	129
3.3 signal 和 slot	130
3.3.1 signal 和 slot 的概念	130
3.3.2 signal 和 slot 的使用	131
3.3.3 signal 和 slot 及其相关技术的本质	135
练习思考题	136
小结	136
3.4 Qt 的基本控件	136

3.4.1	QPushButton, QLabel, QLineEdit	136
	练习思考题	143
3.4.2	QComboBox, QListBox, QSlider, QProgress	143
3.4.3	使用 QT Designer 设计界面	146
	练习思考题	155
3.5	系统相关程序的开发	155
3.5.1	文件系统的交互	155
	练习思考题	161
3.5.2	和终端程序的交互	161
	练习思考题	168
	小结	168
	参考资料	168
4	脚本编程	169
4.1	Shell 脚本编程	169
4.1.1	Shell 介绍、定义和功能	169
4.1.2	Linux Shell 编程工具箱	172
4.1.3	Shell 脚本编程	180
	练习思考题	191
4.2	Python 脚本编程	191
4.2.1	Python 语言概述	192
4.2.2	Python 编程基本概念	194
4.2.3	控制流	201
4.2.4	函数	206
4.2.5	高级数据结构	212
	练习思考题	220
5	代码组织、管理与发布	221
5.1	使用 GNU make 管理多文件项目	221
5.1.1	多文件项目架构分析	222
5.1.2	Makefile 结构分析	223
5.1.3	Makefile 的规则	225
5.1.4	伪目标技术	235
5.1.5	Makefile 中的变量	238
5.1.6	Makefile 的隐含规则	249
5.1.7	Makefile 中的函数	257
5.1.8	执行 make	260
5.2	版本控制工具 CVS	265

5.2.1 版本控制简介	265
5.2.2 版本控制工具 CVS 的使用	265
5.3 使用 autoconf 和 automake 开发自配置软件	273
5.3.1 什么是 autoconf	273
5.3.2 autoconf 是如何工作的	273
5.3.3 使用 autoconf 开发	273
5.3.4 使用 automake 维护 Makefile	279
练习思考题	284

1 基本开发环境

本章主要介绍 Linux 编程的特点、如何搭建 Linux 的基本开发环境以及 Linux 的基本命令和编辑工具的使用。了解了这些内容，就可以开始 Linux 平台上的开发之旅。

通过对本章的学习，将能够：

- (1) 了解 Linux 编程和 Windows 编程的主要区别；
- (2) 了解 Linux 基本编程环境的安装；
- (3) 了解 Linux 下基本命令和编辑工具的使用。

1.1 Linux 编程综述

1.1.1 Linux 编程特点

Linux 作为一个遵循 GNU 的操作系统，源代码的开放性已成为其最大的特点，可以方便地得到 Linux 系统及其应用软件的源代码。Linux 编程就是在这样的环境下对程序进行编写或修改工作的。

1.1.2 Linux 编程与 Windows 编程的主要区别

Windows 一面以最友好的界面展现给程序员，一面却严格限制程序员对其进行透彻地研究，把自己用华丽的外表包装起来了。而 Linux 则一直以真面目示人，虽然相对有些神秘、费解，但内心是对刻苦钻研者敞开的。

在 Linux 系统的程序设计中，有一些重要特性要特别注意：

(1) Linux 是多任务的。可以同时创建多个进程或者线程。绝不能认为自己的程序是系统唯一的运行实例；同一个用户或者不同的用户可以运行该程序的其他实例。因此，在某些情况下必须小心对待同步的问题。

(2) Linux 是个真正的多用户系统。这就意味着存在某种安全机制，以确保把不同用户的文件隔开。用户的程序不能像在其他操作系统中那样，可以修改或者移动任何文件，除非它是 root。

(3) Linux 有分时机制。分时的意思是允许多个用户同时登录，或者一个用户登录多次。用户可以使用诸如 telnet、ssh 或者 X 技术远程登录。因此必须知道你的程序可能同时被许多用户使用。

还有，与 Windows 相比较而言，Linux 要便宜得多。在 Linux 上做开发，基本上不需要再另外安装开发工具，各个 Linux 发布版中都会集成很多开发工具，且不需要另外购买。

1.1.3 Linux 编程与 Unix 编程的主要区别

Unix 系统并非完全开放的，商业 Unix 发布版价格昂贵；且各个 Unix 发布版之间也不是完全一致，有可能在一个 Unix 发布版上开发的程序到了另外一个 Unix 发布版中还需要很大的修改才可编译执行，也就是说在哪个 Unix 发布版上学习的就只能在哪个发布版上使用。

而 Linux 的各个发布版之间的区别较小，在一个发布版上开发的源程序基本上不需要修改就可以在另外一个发布版上编译执行，所以掌握一个 Linux 发布版的编程方法就等于学会了 Linux 的系统编程。

⑬ 这里所讲的发布版本指的是同一时间段的发布版本，也就是说核心以及系统库均差别不大，假如把在一个 3 年前的发布版上开发的软件拿到一个今天的发布版本上编译，那么有极大可能是无法编译通过的。

1.2 Linux 编程环境

本节以红旗 Linux 系统为例，介绍 Linux 的编程环境，包括编程环境的安装以及基本工具的使用。

1.2.1 编程系统安装

Linux 各个发布版本中都集成了很多的开发工具，一个较为完整的操作系统，除了包含主要的系统核心以及常规管理和系统控制模块之外，还常会集合开发编程和常规应用等工具。按照各系统发布商的不同文化理念，其模块的组合与安装过程也会略有不同。在红旗的系统中需要在完成基本系统安装之后，另行选择在需要的时候通过配套的开发工具盘来完成其简单智能的安装过程。在开源文化技术领域，也可以随时通过相关社区下载不同功能的开发工具自行手工配装。

1.2.1.1 基本 Linux 系统安装技术

在商业发布的操作系统中，基本系统的安装均可通过人机交互的智能界面简单完成，相关操作技术以及所涉及的知识，可参考对应产品技术手册或相关的教程。

1.2.1.2 开发工具集的安装

完成红旗 Linux 的基本系统安装后，在基本系统中已经包含了开发使用的 gcc 编译器和常用的各种开发库，具备了基本的开发环境，但是一些图形的集成开发工具和调试工具还没有包括在里面。红旗 Linux 系统把一些常用的图形开发工具和调试工具单独放在了开发工具盘中，可跟随安装向导完成安装；也可以通过 YUM 库随时选择需要的工具集。

准备好红旗 Linux 系统的开发工具盘，进入系统，启动图形界面，将开发工具盘放入光驱中。红旗开发工具包安装程序将被自动引导，在欢迎界面后，点击【下一步】按钮就可

以开始安装了。首先进入的是“选择软件包”界面，在这里，可以选择需要安装的软件包。基于希望拥有一个完全的开发环境的原因，推荐全部安装这些软件包（图 1-1）。

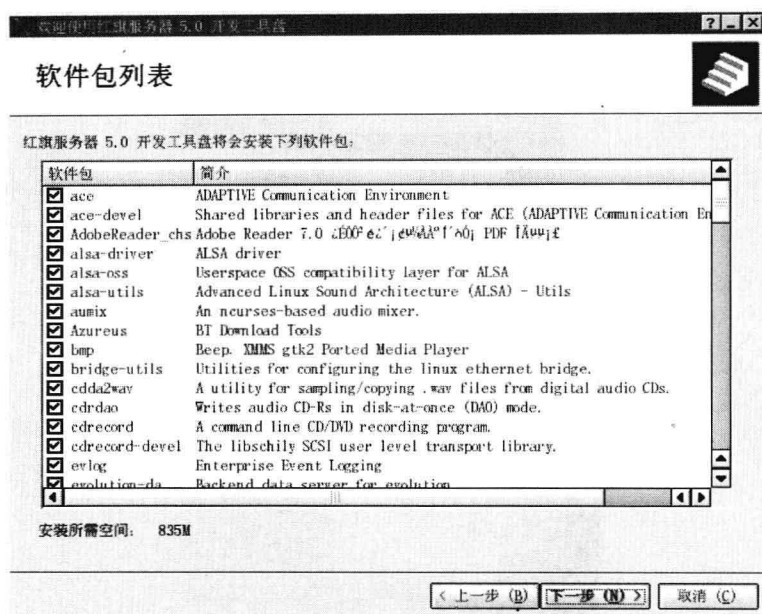


图 1-1 选择软件包界面

点击【下一步】按钮将开始安装软件包（图 1-2）。

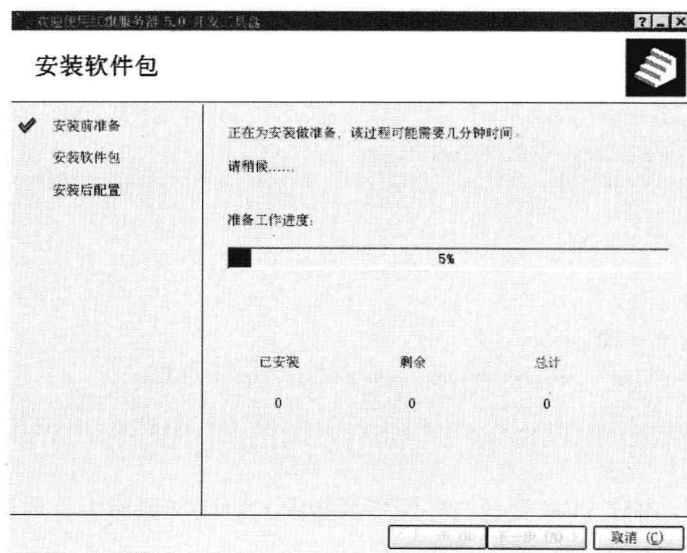


图 1-2 开始安装软件包

安装完成后，安装程序会自动执行安装后的配置，然后显示完成界面，点击【结束】按钮就可以结束开发工具盘的安装。

对于红旗 Linux 桌面系统来说,如果需要配置基本开发环境,除了使用全光盘安装之外,也可在基本系统环境下,通过 YUM 库更新系统,如图 1-3 所示。在“开始”菜单中,选择“更新”,在“软件管理”模块中,即可自动根据系统所设置的“软件源”,通过网络或安装光盘来完成软件的添加、删除等管理工作。

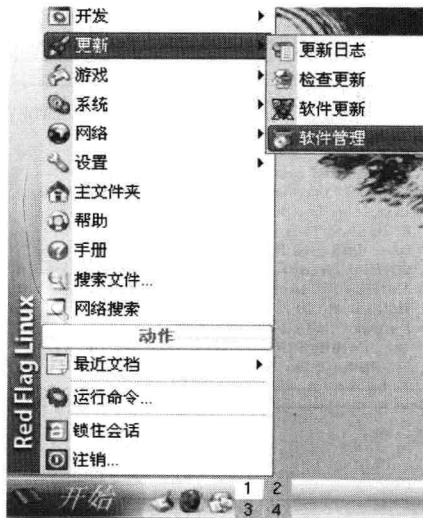


图 1-3 YUM 软件更新

以红旗 Linux 桌面系统的 YUM 管理体系为例,“软件源”的配置文件在“/etc/yum.repos.d/”中的,例如,redflag-dt6sp3.repo 中的配置指向。

```
[DT60SP3_Repository_Local]
name = RedFlag Desktop 6.0 SP3 inter
baseurl = http://172.16.82.48/yum/RedFlag/RPMS/
gpgcheck = 0
enabled = 1

[DT60SP3_Repository]
name = RedFlag Desktop 6.0 SP3 www
baseurl = http://www.redflag-linux.com/source/yum/dt60sp3/Update/RPMS
enabled = 1
gpgcheck = 0
```

这里分别给出了内部 YUM 资源与外网资源指向,对应资源库中,均提供了由红旗匹配好的相关工具。

如果系统配置的软件源有效,则可在分类列表中选择“开发”类,并跟随向导,完成所匹配开发工具的安装,如图 1-4 所示。

到此为止,一个基本的开发环境就构造好了。

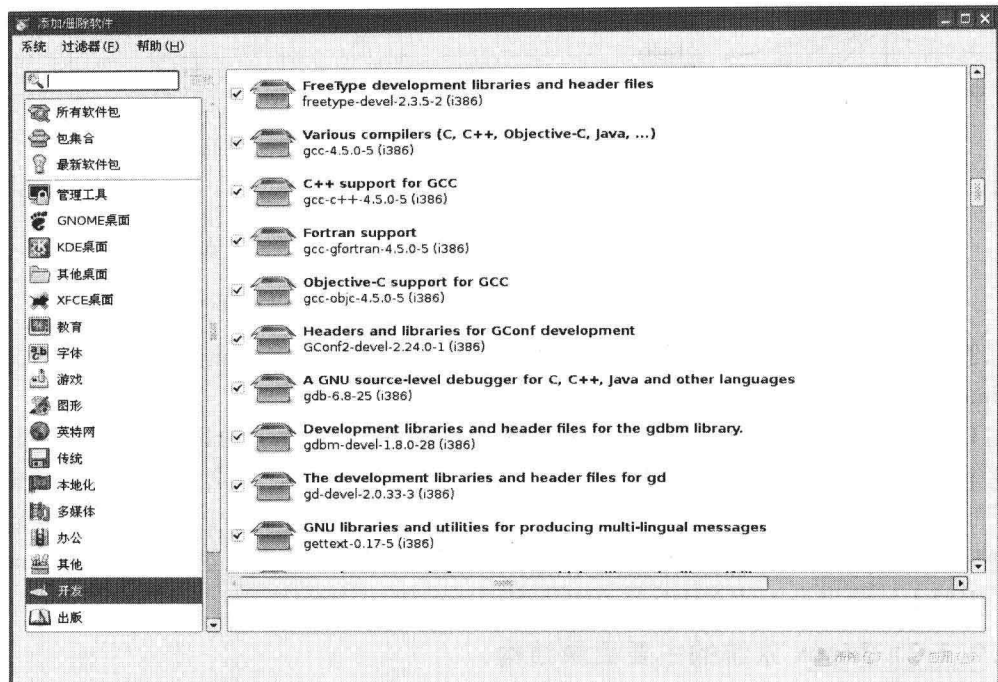


图 1-4 软件仓库安装选择

1.2.2 Linux 系统文件分布和组织

1.2.2.1 Linux 的文件分布结构

很多熟悉 Windows 的用户刚接触 Linux 时，对 Linux 的磁盘管理感到最头疼，因为在 Linux 系统中不会看到 C 盘、D 盘这样的东西，而只能看到一些目录，那么 Linux 是如何管理磁盘的呢？在 Linux 系统中如何访问不同的磁盘分区呢？

Linux 的文件系统没有 Windows 下的盘符的概念，Linux 系统以文件目录的方式来组织和管理系统中的所有文件。所谓文件目录就是将所有文件的说明信息采用树型结构组织起来——即常说的目录。也就是说，整个文件系统有一个“根”（root），然后在根上分“权”（directory），任何一个分权上都可以再分权，权上也可以长出“叶子”。“根”和“权”在 Linux 中被称为是“目录”或“文件夹”，而“叶子”则是一个个的文件。

Linux 系统通过目录将系统中所有的文件分级、分层组织在一起，形成了 Linux 文件系统的树型层次结构（图 1-5）。以根目录为起点，所有其他的目录都由根目录派生而来，用户可以浏览整个系统，可以进入任何一个已授权进入的目录，访问那里的文件。

Windows 的目录结构也是树型结构，不过 Windows 中这样的树型结构的根是磁盘分区的盘符，有几个分区就有几个树型结构，他们之间的关系是并列的（图 1-6）。但是在 Linux 中，无论操作系统管理几个磁盘分区，这样的目录树只有一个。

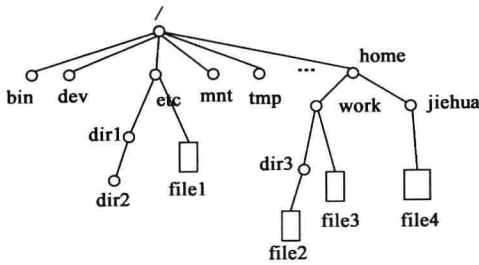


图 1-5 Linux 树型目录结构

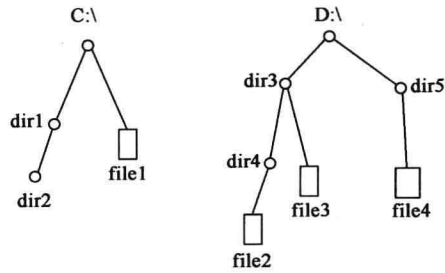


图 1-6 Windows 树型目录结构

对于只有一个磁盘分区的文件系统还好理解，那么如果是多个磁盘分区组成的文件系统，Linux 如何管理不同的分区呢？

在 Linux 系统中有一个挂载（mount）的概念，可以把一个磁盘分区挂载到一个目录下，使得这个磁盘分区成为整个文件系统树的一部分，当挂载成功后，原来挂载目录（也称挂载点）下的文件就消失了，取而代之的是该磁盘分区上的内容。通过这种方式就可以在一个树型目录结构中访问所有的磁盘分区。在访问结束后，可以将磁盘分区卸载（umount）下来，则原来挂载目录下的文件就又可以访问了。

1.2.2.2 Linux 系统的主要目录结构

（1）Linux 文件系统中各主要目录的内容和用途。

Linux 目录树的主要部分有/、/usr、/var、/home 等，下面是一个典型的 Linux 目录结构（表 1-1）。

表 1-1 Linux 目录结构

目录	内容和用途	目录	内容和用途
/	根目录	/proc	存放存储进程和系统信息
/bin	存放必要的命令	/root	超级用户的主目录
/etc	存放系统的配置文件	/sbin	存放系统管理程序
/dev	存放设备文件	/tmp	存放临时文件的目录
/boot	存放内核以及启动所需的文件等	/usr	包含了一般不需要修改的应用程序，命令程序文件、程序库、手册和其他文档
/home	用户文件的主目录，用户数据存放在其主目录中	/mnt	存放临时的映射文件系统，常把软驱和光驱挂装在这里的 floppy 和 cdrom 子目录下
/lib	存放必要的运行库	/var	包含系统产生的经常变化的文件，如打印机、邮件、新闻等假脱机目录、日志文件、格式化后的手册页以及一些应用程序的数据文件等

（2）典型的/usr 目录下的内容如表 1-2 所示。

表 1-2 /usr 目录结构

目录	内容和用途	目录	内容和用途
/usr/X11R6	存放 X window 系统	/usr/lib	存放库文件
/usr/bin	存放增加的用户程序	/usr/local	存放本地产生的增加的应用程序
/usr/doc	存放追加的文档	/usr/man	存放在线帮助文件
/usr/etc	存放配置文件	/usr/sbin	存放增加的管理程序
/usr/games	存放游戏和教学文件	/usr/share	存放结构独立的数据
/usr/include	存放 C 开发工具的头文件	/usr/src	存放程序的源代码

1.2.3 Linux 系统命令操控技术

随着 Linux 桌面技术的飞速发展，现在，在 Linux 上也可以像在 Windows 系统上一样进行图形操作了，但是，由于 Linux 上的命令功能强大、灵活、方便，很多熟悉 Linux 的人还是热衷于使用命令来操作 Linux，所以还是有必要简单介绍一下 Linux 系统的常用命令。

1.2.3.1 Shell 简介

用户在命令行下工作时，不是直接同操作系统内核打交道，而是由命令解释器接收命令，分析后再传给相关的程序。进入 Linux 环境时系统将自动启动相应的 Shell，Shell 是一种命令行解释程序，它提供用户与操作系统之间的接口。红旗 Linux 系统默认的 Shell 是 bash。

bash 命令的基本格式为：

```
命令名 [选项] <参数1> [参数2] ...
```

其中方括号括起的部分表明该项对命令而言是可选的；尖括号括起的部分表示该项对命令而言是必选的。

(1) [选项]：对命令有特别定义，一般以“-”开始，多个选项可用一个“-”连起来，如 ls -l -a 与 ls -la 相同；

(2) [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。

⊗ 输入用户名、口令与文件名、命令名时，一定要区分大小写，因为大小写字母在 Linux 系统中代表不同的含义。

在命令、选项和参数之间要用空格隔开。连续的空格会被 Shell 解释为单个空格。

在 Shell 提示符下输入相应的命令，然后按回车键确认，Shell 会读取该命令并执行。如果系统找不到输入的命令，则会显示“Command not Found”，这时需要检查键入命令的拼写及大小写是否正确。

使用分号(;)可以将两个命令隔开，这样可以实现在一行中输入多个命令。命令的执行顺序与输入的顺序相同。

1.2.3.2 常用命令介绍

由于本书重点在于介绍 Linux 编程方面的知识，所以这里只是先介绍一下 Linux 的常用

命令, 更详细的命令介绍请参考相关 Linux 系统使用方面的书籍。

1. 进入与退出系统

(1) 进入系统。

Linux 系统正常启动之后, 会进入系统登录界面, 登录界面分两种, 文本登录 (图 1-7) 和图形登录 (图 1-8)。具体启动哪种登录界面是在安装程序中设定的, 并可根据需要重新设定。



图 1-7 文本登录界面

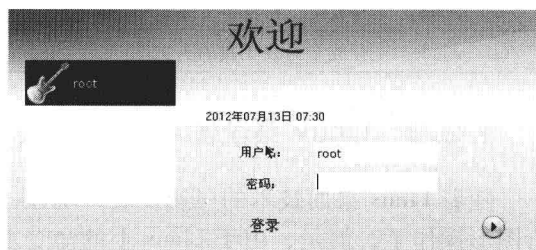


图 1-8 图形登录界面

必须要输入正确的用户名和密码才能进入 Linux 系统, 如在安装 Linux 系统的过程中, 设置了超级用户密码, 这时, 就可以使用超级用户登录 Linux 系统。

Linux 系统的超级用户是 root, 填入 root 这个用户名 (文本登录时直接在 login: 后面写, 然后回车确认), 然后输入在安装系统时设置好的密码, 确认后就可登录进入系统。

一般的 Linux 使用者均为普通用户, 而系统管理员一般使用超级用户账户完成一些系统管理的工作。如果只需要完成一些由普通用户就能完成的任务, 建议不要使用超级用户, 以免无意中破坏系统, 影响系统的正常运行。

当用户正确地输入用户名和口令后, 就能合法地进入系统。文本登录时屏幕会显示 shell 的提示符:

```
[root@localhost ~] #
```

(2) 增加一般用户。

进入系统之后, 首先的任务是建立一个用于日常工作的一般用户, 可以使用命令 useradd 来增加用户:

```
useradd <用户名>
```

然后使用 passwd 命令来为这个用户设置密码:

```
passwd <用户名>
```

它将提示用户输入新口令, 之后还要求用户再次确认新口令, 以避免用户无意中按错键。设置完成后就可以使用这个用户登录系统了。注意, 这里是用超级用户来执行 passwd 命令为一般用户设置密码, 如果用户自己修改密码也是运行这个命令, 但是该命令会首先要求用户输入正确的旧密码, 否则不能修改密码。

如果使用一般用户登录系统, shell 的提示符是 “\$”, 而不再是 “#”:

```
[jiejia@localhost ~] $
```