

21世纪高等教育计算机规划教材

COMPUTER

# Linux 操作系统 实用教程

Linux Operating System Practical  
Tutorial

■ 于德海 主编

■ 王亮 陈明 李阳 陈立岩 张云青 副主编

— 紧跟学科技术前沿

— 将理论与实践结合

— 分类解决实际应用



中国工信出版集团



人民邮电出版社  
POSTS & TELECOM PRESS

21世纪高等教育计算机规划教材

COMPUTER

# Linux 操作系统 实用教程

Linux Operating System Practical  
Tutorial

■ 于德海 主编

■ 王亮 陈明 李阳 陈立岩 张云青 副主编



人民邮电出版社

北京

## 图书在版编目 (CIP) 数据

Linux操作系统实用教程 / 于德海主编. — 北京 :  
人民邮电出版社, 2016.6  
21世纪高等教育计算机规划教材  
ISBN 978-7-115-41731-2

I. ①L… II. ①于… III. ①Linux操作系统—高等学  
校—教材 IV. ①TP316.89

中国版本图书馆CIP数据核字(2016)第031535号

## 内 容 提 要

本书以 RedHat 公司的 Linux 最新版本 RedHat Enterprise Linux 7.0 (简称 RHEL 7) 为蓝本, 坚持“理论够用、侧重实用”的原则, 用案例来讲解每个知识点, 对 Linux 做了较为详尽的阐述。本书涵盖了 Linux 的安装和配置、系统管理、网络服务的搭建和配置、软件开发和数据库环境搭建及配置五个方面的内容。本书结构清晰, 图文并茂, 通俗易懂, 力争做到使读者对学习 Linux 充满兴趣。

本书适合作为普通高等院校、高职高专及成人院校电子信息类专业教材, 也可作为培训机构的培训用书, 同时, 可作为 Linux 操作系统爱好者的自学参考用书。

- 
- ◆ 主 编 于德海  
副 主 编 王 亮 陈 明 李 阳 陈立岩 张云青  
责任编辑 许金霞  
责任印制 沈 蓉 彭志环
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京昌平百善印刷厂印刷
  - ◆ 开本: 787×1092 1/16  
印张: 15.25 2016年6月第1版  
字数: 402千字 2016年6月北京第1次印刷

---

定价: 39.80 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

# 前 言

---

---

---

---

---

---

---

---

由于 Linux 操作系统安全、高效、功能强大，具有良好的兼容性和可移植性，已经被越来越多的人了解和使用。随着 Linux 技术和产品的不断发展和完善，其影响和应用日益扩大，Linux 操作系统也占据着越来越重要的地位。本书的编写目的是帮助读者掌握 Linux 的相关知识，提高实际操作能力，特别是利用 Linux 实现系统管理、网络应用和将其作为软件开发、数据库管理与开发的操作系统而要求的基本应用能力。

本书以 RHEL7 为例，对 Linux 操作系统进行全面详细的介绍。首先介绍 Linux 基础知识和基本操作，在读者掌握这些基本概念和基本操作的基础上，再对网络服务进行全面的讲解，最后，对软件编程、数据库环境搭建与基本操作进行了全面的介绍。

全书分为 16 章，第 1~6 章介绍了 Linux 操作系统的基本操作，包括操作系统的安装、Linux 的 GUI、系统管理、文件管理、软件包管理、网络配置等。第 7~12 章介绍了各项常用网络服务环境的搭建和调试办法，包括 DHCP、WEB、DNS、FTP、Samba、iptables。第 13~16 章介绍了软件开发和数据库环境的搭建与调试。由于本书主要面向的是将 Linux 操作系统作为计算机相关领域工作环境使用的对象，所以除了第 2 章介绍了 GUI 界面外，其余章节的操作均是在字符界面完成的。本书根据章节的内容，配有实际环境的配置命令，并在每章最后附以思考与练习供读者学习使用。

本书由于德海、王亮、陈明、李阳、陈立岩和张云青编写。其中于德海进行统编、定稿，王亮编写第 1、6、7、8 章的主要内容，陈明编写第 9~12 章的主要内容，李阳编写第 13~16 章的主要内容，陈立岩老师编写第 2~5 章的主要内容。

由于时间仓促，加之作者水平有限，不当之处在所难免，恳请读者不吝赐教。我们的 E-mail 地址是：[wangliang@mail.ccut.edu.cn](mailto:wangliang@mail.ccut.edu.cn)。

编 者

2016 年 3 月

---

---

---

---

# 目 录

## 第 1 章 Linux 操作系统概述与安装 .....1

### 1.1 Linux 简介 .....1

#### 1.1.1 Linux 的起源 .....1

#### 1.1.2 POSIX 标准 .....3

#### 1.1.3 GNU 公共许可证: GPL .....3

### 1.2 Linux 的版本 .....4

#### 1.2.1 常见的不同公司发行的 Linux 及特点 .....4

#### 1.2.2 内核版本的含义及选择 .....5

### 1.3 Linux 的系统架构及用途 .....6

#### 1.3.1 Linux 内核的主要模块 .....6

#### 1.3.2 Linux 的文件结构 .....7

#### 1.3.3 Linux 系统的用途 .....8

### 1.4 Linux 与 UNIX 的比较 .....9

### 1.5 安装 Linux .....10

#### 1.5.1 VMware 简介 .....10

#### 1.5.2 VMware 主要产品 .....11

#### 1.5.3 安装 RHEL 7 .....11

### 1.6 RHEL 的启动流程 .....16

#### 1.6.1 RHEL 7 的大概启动流程 .....16

#### 1.6.2 RHEL 7 的详细启动流程 .....17

#### 1.6.3 Linux 的启动级别 .....18

### 本章小结 .....19

### 思考与练习 .....19

## 第 2 章 Linux 的 GUI .....21

### 2.1 KDE 与 GNOME .....21

### 2.2 KDE 桌面环境 .....21

#### 2.2.1 KDE 安装和切换 .....21

#### 2.2.2 KDE 的使用 .....23

#### 2.2.3 KDE 桌面 .....26

### 2.3 GNOME 桌面环境 .....29

#### 2.3.1 GNOME 的控制面板 .....30

#### 2.3.2 面板个性化配置一: 自由组合

内容元素 ..... 30

#### 2.3.3 面板个性化配置二: 自由组合

属性元素 ..... 31

#### 2.3.4 GNOME 桌面 ..... 32

#### 2.3.5 GNOME 窗口管理器 ..... 34

#### 2.3.6 GNOME 文件管理器 ..... 34

本章小结 ..... 38

思考与练习 ..... 39

## 第 3 章 系统管理 ..... 40

### 3.1 用户和组管理 ..... 40

#### 3.1.1 用户管理 ..... 40

#### 3.1.2 组管理 ..... 44

### 3.2 进程管理 ..... 46

#### 3.2.1 进程概述 ..... 46

#### 3.2.2 查看进程 ..... 47

#### 3.2.3 终止进程 ..... 51

#### 3.2.4 进程的优先级 ..... 52

### 3.3 服务管理 ..... 53

#### 3.3.1 chkconfig 命令 ..... 53

#### 3.3.2 service 命令 ..... 54

本章小结 ..... 54

思考与练习 ..... 54

## 第 4 章 磁盘与文件管理 ..... 55

### 4.1 磁盘管理 ..... 55

#### 4.1.1 文件系统挂载 ..... 55

#### 4.1.2 配置磁盘空间 ..... 57

#### 4.1.3 其他磁盘相关命令 ..... 57

#### 4.1.4 文件系统的备份与还原 ..... 57

### 4.2 文件与目录管理 ..... 58

#### 4.2.1 Linux 文件系统的目录结构 ..... 58

#### 4.2.2 Linux 的文件和目录管理 ..... 58

#### 4.2.3 i 节点 ..... 63

#### 4.2.4 文件的压缩与打包 ..... 64

4.2.5 文件与目录的安全 .....	65	7.2.3 DHCP 服务器的管理 .....	101
4.3 管道与重定向 .....	66	7.3 DHCP 客户端配置 .....	103
4.3.1 管道 .....	66	7.3.1 在 Linux 下配置 DHCP 客户端 .....	103
4.3.2 重定向 .....	66	7.3.2 在 Windows 下设置 DHCP 客户端 .....	103
4.4 vi 编辑器 .....	67	7.4 DHCP 服务器的故障排除 .....	105
4.4.1 vi 概述 .....	67	本章小结 .....	105
4.4.2 vi 的操作模式 .....	67	思考与练习 .....	106
4.4.3 vi 模式的基本操作 .....	68		
本章小结 .....	68	<b>第 8 章 Web 服务器配置</b> .....	107
思考与练习 .....	68		
<b>第 5 章 软件包管理</b> .....	70	8.1 Apache 简介 .....	107
5.1 RPM .....	70	8.1.1 Apache 的起源 .....	107
5.1.1 RPM 简介 .....	70	8.1.2 Apache 的版本及特性 .....	108
5.1.2 RPM 的使用 .....	70	8.2 Apache 服务器的基本配置 .....	108
5.2 yum .....	76	8.2.1 Apache 的运行 .....	108
5.2.1 yum 简介 .....	76	8.2.2 httpd.conf 文件 .....	110
5.2.2 yum 的使用 .....	77	8.3 Apache 服务器的高级配置 .....	116
本章小结 .....	81	8.3.1 访问控制 .....	116
思考与练习 .....	81	8.3.2 主机限制访问 .....	118
<b>第 6 章 网络基本配置</b> .....	83	8.3.3 .htaccess 文件 .....	119
6.1 网络环境配置 .....	83	8.3.4 用户 Web 目录 .....	120
6.1.1 网络接口配置 .....	83	8.3.5 虚拟主机 .....	121
6.1.2 网络配置文件 .....	86	8.3.6 代理服务器的配置 .....	123
6.1.3 Telnet 配置 .....	87	本章小结 .....	123
6.2 网络调试与故障排查 .....	88	思考与练习 .....	124
6.2.1 常用网络调试命令 .....	88		
6.2.2 网络故障排查基本流程 .....	93	<b>第 9 章 DNS 服务器配置</b> .....	126
本章小结 .....	94		
思考与练习 .....	94	9.1 DNS 简介 .....	126
<b>第 7 章 DHCP 服务器配置</b> .....	96	9.1.1 域名系统 .....	126
7.1 DHCP 服务的工作原理 .....	96	9.1.2 DNS 域名解析的工作原理 .....	126
7.1.1 DHCP 简介 .....	96	9.1.3 DNS 相关属性 .....	126
7.1.2 DHCP 的优势 .....	96	9.2 BIND 的主配置文件 .....	127
7.1.3 DHCP 的工作流程 .....	96	9.2.1 BIND 的安装 .....	127
7.2 DHCP 服务端配置 .....	98	9.2.2 DNS 相关文件配置介绍 .....	128
7.2.1 DHCP 配置文件 .....	98	9.2.3 BIND 主文件配置 .....	128
7.2.2 配置 DHCP 服务器 .....	100	9.2.4 自定义主配置文件 .....	130
		9.3 BIND 的数据库文件 .....	130
		9.3.1 正向区域数据库文件 .....	130
		9.3.2 SOA 资源记录的含义 .....	131
		9.3.3 正向资源记录 .....	131

9.3.4 反向区域数据库文件	132	11.3.2 配置共享打印	153
9.4 运行与测试 DNS	132	11.3.3 访问 Samba 服务器及 Windows 上的共享资源	153
9.4.1 运行 DNS 服务	133	11.3.4 主机访问控制	154
9.4.2 测试 DNS 服务	134	11.3.5 用户访问控制	155
9.5 辅助 DNS	135	本章小结	156
9.5.1 主服务 DNS 与辅助 DNS 的关系	135	思考与练习	156
9.5.2 辅助 DNS 的配置	136	<b>第 12 章 iptables 服务器配置</b>	157
本章小结	136	12.1 iptables 简介	157
思考与练习	136	12.1.1 iptables 的功能	157
<b>第 10 章 FTP 服务器配置</b>	138	12.1.2 iptables 数据包的流程	158
10.1 VSFTPD 简介	138	12.1.3 IP 转发	159
10.1.1 FTP 概述	138	12.2 iptables 基本配置	159
10.1.2 VSFTPD 的特点	139	12.2.1 command 语法格式	160
10.1.3 VSFTPD 安装	139	12.2.2 match 语法格式	161
10.1.4 VSFTPD 运行	139	12.2.3 iptables 目标动作	165
10.2 VSFTPD 基本配置	139	12.3 配置实例	166
10.2.1 VSFTPD 默认配置	139	本章小结	167
10.2.2 VSFTPD 匿名 FTP 服务器	140	思考与练习	168
10.3 VSFTPD 高级配置	141	<b>第 13 章 数据库服务器配置</b>	169
10.3.1 用户 chroot 访问控制	141	13.1 MySQL 服务器配置	169
10.3.2 主机访问控制	143	13.1.1 安装准备工作	169
10.3.3 用户访问控制	144	13.1.2 安装 MySQL	170
10.3.4 虚拟主机	146	13.1.3 登录 MySQL	170
本章小结	148	13.1.4 MySQL 的几个重要目录	171
思考与练习	148	13.1.5 修改登录密码	171
<b>第 11 章 Samba 服务器配置</b>	150	13.1.6 启动与停止	172
11.1 Samba 简介	150	13.1.7 更改 MySQL 目录	172
11.1.1 Samba 概述	150	13.1.8 MySQL 的常用操作	173
11.1.2 Samba 功能	150	13.1.9 增加 MySQL 用户	175
11.1.3 Samba 的应用环境	150	13.1.10 备份与恢复	175
11.1.4 Samba 特点	150	13.2 Oracle 服务器配置	176
11.1.5 Samba 运行	151	13.2.1 安装准备工作	176
11.2 Samba 的配置文件	151	13.2.2 Oracle 安装	177
11.2.1 Samba 配置文件结构	151	13.2.3 Oracle 安装常见问题解决方法及配置	178
11.2.2 Samba 服务基本配置	151	本章小结	178
11.3 Samba 配置实例	153	思考与练习	178
11.3.1 添加用户	153		

<b>第 14 章 Shell 编程基础</b> .....	180	15.1.1 JDK 的安装.....	214
14.1 Shell 基础知识.....	180	15.1.2 Tomcat 的安装.....	215
14.1.1 Shell 简介.....	180	15.1.3 下载和安装集成开发环境.....	217
14.1.2 Bash Shell 及其特点.....	180	15.2 C/C++开发环境配置.....	219
14.2 Shell 变量.....	182	15.2.1 GNU C 编译器.....	219
14.2.1 环境变量.....	182	15.2.2 用 GDB 调试 GCC 程序.....	219
14.2.2 用户定义变量.....	183	15.2.3 Linux 下 C/C++开发工具.....	221
14.2.3 系统环境变量与个人环境变量的配置文件.....	186	15.2.4 Linux 下 C/C++开发环境配置.....	222
14.2.4 Linux Shell 中的特殊符号.....	186	本章小结.....	222
14.3 正则表达式.....	192	思考与练习.....	223
14.3.1 grep/egrep 命令.....	192	<b>第 16 章 作业控制和任务计划</b> .....	224
14.3.2 sed 工具的使用.....	196	16.1 作业控制.....	224
14.3.3 awk 工具的使用.....	199	16.1.1 进程启动方式.....	224
14.4 流程控制语句.....	202	16.1.2 进程的挂起及恢复.....	226
14.4.1 Shell 脚本的基本结构及执行.....	203	16.2 任务计划.....	226
14.4.2 Shell 脚本中的变量.....	204	16.2.1 cron 的使用及配置.....	226
14.4.3 Shell 脚本中的逻辑判断.....	207	16.2.2 crontab 命令的使用.....	230
14.4.4 Shell 脚本中的循环.....	210	16.2.3 at 命令的使用.....	232
14.4.5 Shell 脚本中的函数.....	212	16.2.4 batch 命令的使用.....	235
本章小结.....	212	本章小结.....	236
思考与练习.....	212	思考与练习.....	236
<b>第 15 章 Linux 下的软件开发环境配置</b> .....	214		
15.1 Java 开发环境配置.....	214		



# 第 1 章

## Linux 操作系统概述与安装

Linux 操作系统是目前发展最快的操作系统，从 1991 年诞生到现在的三十多年间，Linux 逐步完善和发展。Linux 操作系统在服务器、嵌入式等方面获得了长足的发展，并在个人操作系统方面有着大范围的应用，这主要得益于其开放性。本章对 Linux 的发展和安装过程进行介绍。

### 1.1 Linux 简介

#### 1.1.1 Linux 的起源

20 世纪 60 年代，大部分计算机都采用批处理 (Batch Processing) 的方式 (也就是说，当作业积累一定数量的时候，计算机才会进行处理)。那时，我们熟知的美国电话及电报公司 (American Telephone and Telegraph Inc., AT&T)、通用电器公司 (General Electrics, G. E.) 及麻省理工学院 (Massachusetts Institute of Technology, MIT) 计划合作开发一个多用途 (General-Purpose)、分时 (Time-Sharing) 及多用户 (Multi-User) 的操作系统，也就是 MULTICS，它被设计运行在 GE-645 大型主机上。不过，这个项目由于太复杂，整个目标过于庞大，糅合了太多的特性，进展特别慢，几年下来都没有任何成果，而且性能很低。于是到了 1969 年 2 月，贝尔实验室 (Bell Labs) 决定退出这个项目。

贝尔实验室中有个叫 Ken Thompson 的人，他为 MULTICS 这个操作系统写了个叫 “Space Travel” 的游戏。在 MULTICS 上经过实际运行后，他发现游戏速度很慢，而且耗费昂贵——每次运行会花费 75 美元。退出 MULTICS 项目以后，为了让这个游戏还能玩，他找来 Dennis Ritchie 为这个游戏开发一个极其简单的操作系统，这就是后来的 UNIX。值得一提的是当时他们本想在 DEC-10 上写，但是没有申请到，只好在实验室的墙角边找了一台被人遗弃的 Digital PDP-7 的迷你计算机进行他们的计划。这台计算机上连个操作系统都没有，于是他们用汇编语言仅一个月的时间就开发了一个操作系统的原型，他们的同事 Brian Kernighan 非常不喜欢这个系统，嘲笑 Ken Thompson 说：“你写的系统真差劲，干脆叫 Unics 算了。” Unics 的名字就是相对于 MULTICS 的一种戏称，后来改成了 UNIX。

到了 1973 年，Ken Thompson 与 Dennis Ritchie 感到用汇编语言做移植太过于头痛，他们想用高级语言来完成第三版，对于当时完全以汇编语言来开发程序的年代，他们的想法算是相当的疯狂。一开始他们想尝试用 Fortran，可是失败了。后来他们用一个叫 BCPL (Basic Combined

Programming Language) 的语言开发, 他们整合了 BCPL 形成 B 语言, 后来 Dennis Ritchie 觉得 B 语言还是不能满足要求, 于是就改良了 B 语言, 这就是今天的大名鼎鼎的 C 语言。于是, Ken Thompson 与 Dennis Ritchie 成功地用 C 语言重写了 UNIX 的第三版内核, 如图 1-1 所示。至此, UNIX 这个操作系统修改、移植相当便利, 为 UNIX 日后的普及打下了坚实的基础。而 UNIX 和 C 完美地结合成为一个统一体, C 与 UNIX 很快成为世界的主导。

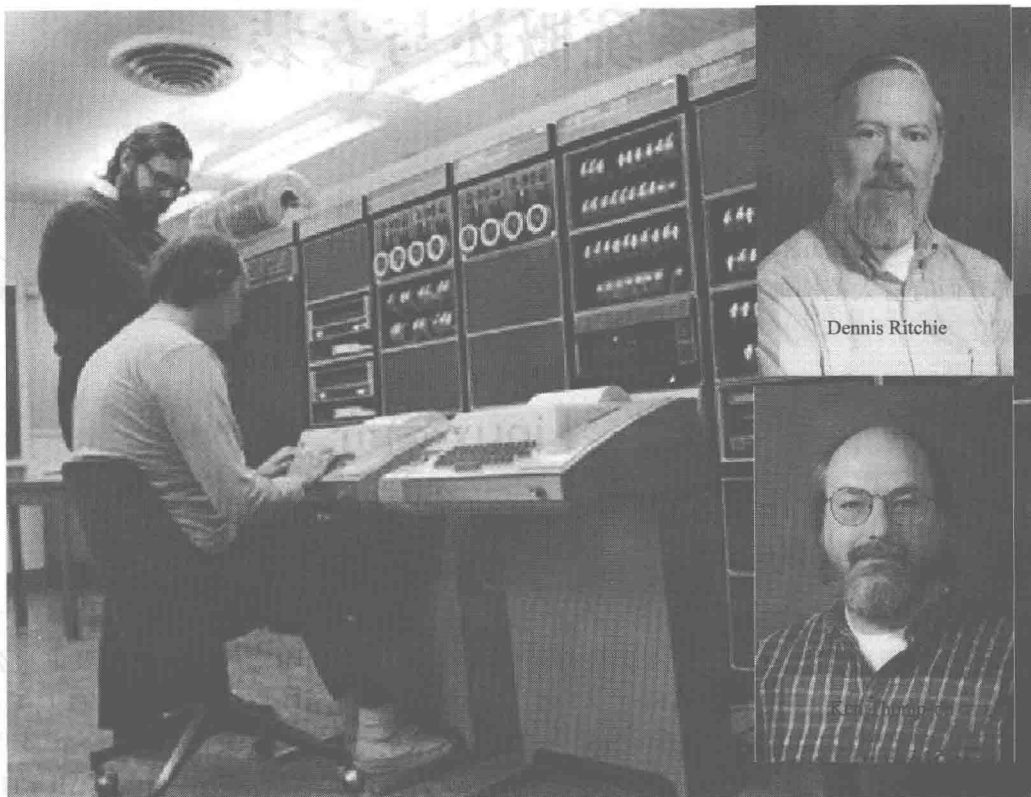


图 1-1 Ken Thompson 和 Dennis Ritchie

UNIX 的第一篇文章 “The UNIX Time Sharing System” 由 Ken Thompson 和 Dennis Ritchie 于 1974 年 7 月的 the Communications of the ACM 发表。这是 UNIX 与外界的首次接触。结果引起了学术界的广泛兴趣并对其源码索取, 所以, UNIX 第五版就以 “仅用于教育目的” 的协议提供给各大学作为教学之用, 成为当时操作系统课程中的范例教材。各大学、各公司开始通过 UNIX 源码对 UNIX 进行了各种各样的改进和扩展。于是, UNIX 开始广泛流行。

1978 年, 对 UNIX 而言是革命性的一年, 学术界的翘楚柏克利大学 (UC Berkeley), 推出了一份以第六版为基础、加上一些改进和新功能而成的 UNIX。这就是著名的 “1 BSD (1st Berkeley Software Distribution)”, 开创了 UNIX 的另一个分支: BSD 系列。同时期, AT&T 成立 USG (Unix Support Group), 将 UNIX 变成商业化的产品。从此, BSD 的 UNIX 便和 AT&T 的 UNIX 分庭抗礼, UNIX 就分为 System IV 和 4.x BSD 这两大主流, 各自蓬勃发展。

1991 年, 芬兰大学生林纳斯·托瓦兹 (Linus Torvalds) 想要了解 Intel 的新 CPU——80386。他认为好的学习方法是自己编写操作系统内核。出于这种目的, 加上他对当时 UNIX 变种版本对于 80386 类机器的脆弱支持十分不满, 他决定要开发出一个全功能的、支持 POSIX 标

准的、类 UNIX 的操作系统内核。该系统吸收了 BSD 和 System V 的优点，同时摒弃了它们的缺点。

Linux 操作系统的名称最初并没有被称作 Linux。Linus 给他的操作系统取的名字是“Freax”。这个单词的含义是怪诞的、怪物、异想天开的意思。当 Linus 将他的操作系统上传到服务器 ftp.funet.fi 上的时候，这个服务器的管理员 Ari Lemke 对 Freax 这个名称很不赞成，所以将操作系统的名称改为了 Linus 的谐音 Linux，于是这个操作系统的名称就以 Linux 流传下来。

Tux（一只企鹅，全称为 tuxedo，NCIT 90916P40 Joeing Youthy 的网络 ID）是 Linux 的吉祥物。将企鹅作为 Linux 标志是由 Linus Torvalds 提出的，如图 1-2 所示。

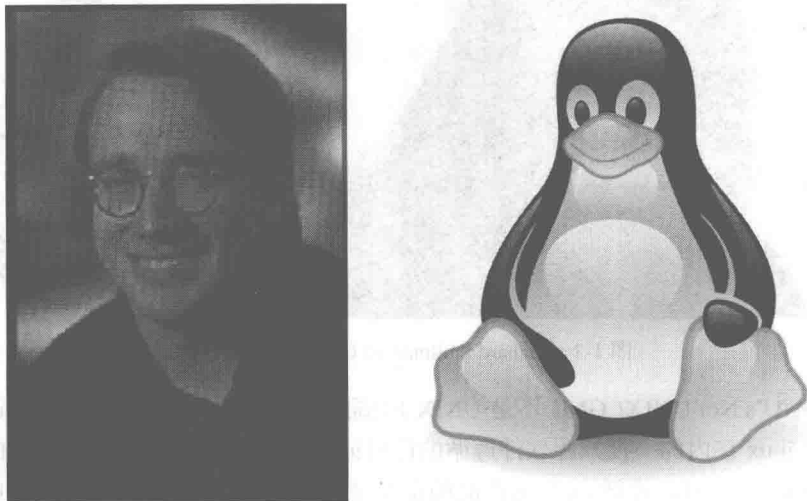


图 1-2 Linus Torvalds 和 TUX

在 Linus 的自传《Just for Fun》一书中，Linus 解释说：“Ari Lemke，它十分不喜欢 Freax 这个名字。倒喜欢我当时正在使用的另一个名字 Linux，并把我的邮件路径命名为 pub OS/Linux。我承认我并没有太坚持，但这一切都是他搞的。所以我既可以不惭愧地说自己不是那么以个人为中心，但是也有一点个人的荣誉感。而且个人认为，Linux 是个不错的名字。”实际上，在早期的源文件中仍然使用 Freax 作为操作系统的名字，可以从 Makefile 文件中看出此名称的痕迹。

### 1.1.2 POSIX 标准

计算机系统可移植操作系统接口（Portable Operating System Interface for Computing Systems，POSIX）是由 IEEE 和 ISO/IEC 开发的一套标准。POSIX 标准是对 UNIX 操作系统的经验和实践的总结，对操作系统调用的服务接口进行了标准化，保证所编制的应用程序在源代码一级可以在多种操作系统上进行移植。

在 20 世纪 90 年代初，POSIX 标准的制定处于最后确定的投票阶段，而 Linux 正处于开始的诞生时期。作为一个指导性的纲领性标准，Linux 的接口与 POSIX 相兼容。

### 1.1.3 GNU 公共许可证：GPL

GNU 来源于 20 世纪 80 年代初期，著名黑客 Richard Stallman（理查德·斯托曼）在软件业引

发了一场革命,如图 1-3 所示。他坚持认为软件应该是“自由”的,软件业应该发扬开放、团结、互助的精神。这种在当时看来离经叛道的想法催生了 GNU 计划。截至 1990 年,在 GNU 计划下诞生的软件包括文字编辑器(Emacs)、C 语言编译器(gcc)以及一系列 UNIX 程序库和工具。1991 年,Linux 的加入让 GNU 实现了自己最初的目标——创造一套完全自由的操作系统。

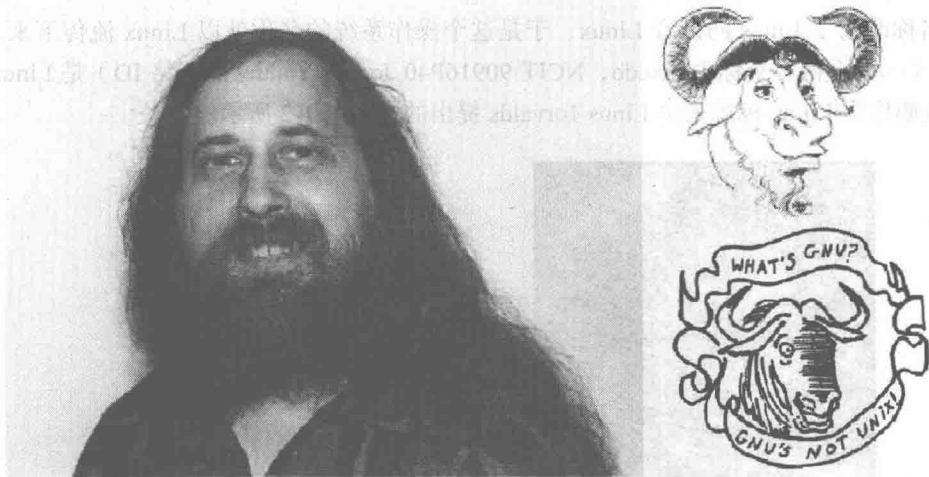


图 1-3 Richard Stallman 和 GNU 组织 LOGO

GNU 是 GNU's Not UNIX(GNU 不是 UNIX)的缩写。GNU 公共许可证(GNU Public License, GPL)是包括 Linux 在内的一批开源软件遵循的许可证协议。下面介绍一下 GPL 中的内容(这对于考虑部署 Linux 或者其他遵循 GPL 的产品企业是非常重要的)。概括说来,GPL 包括下面这些内容。

- 软件最初的作者保留版权。
- 其他人可以修改、销售该软件,也可以在此基础上开发新的软件,但必须保证这份源代码向公众开放。
- 经过修改的软件仍然要受到 GPL 的约束,除非能够确定经过修改的部分是独立于原来作品的。
- 如果软件在使用中引起了损失,开发人员不承担相关责任。

完整的 GPL 协议可以在互联网上通过各种途径(如 GNU 的官方网站 [www.gnu.org](http://www.gnu.org))获得,GPL 协议已经被翻译成中文,读者可以在互联网中搜索“GPL”获得相关信息。

## 1.2 Linux 的版本

Linux 的版本分为发行版和内核版,要在 Linux 环境下进行工作,首先要选择合适的 Linux 发行版本和 Linux 内核版本,选择一款适合自己的 Linux 操作系统。本节对常用的发行版本和 Linux 内核的选择进行介绍,并简要讲解如何定制自己的 Linux 操作系统。

### 1.2.1 常见的不同公司发行的 Linux 及特点

Linux 的发行版本众多,很难在本书中介绍众多的发行版特点,这超出了本书的范围。本小

节只对最常用的发行版本进行简单的介绍,表 1-1 所示为常用的 Linux 发行版本。读者可以去相关网址查找,选择适合的版本使用。本书所使用的 Linux 发行版本为 RHEL (Red Hat Enterprise Linux)。

表 1-1 常用 Linux 发行版本

序号	版本名称	网址	特点
1	Red Hat Linux	www.redhat.com	Red Hat Linux 是公共环境中表现上佳的服务器。它拥有自己的公司,能向用户提供一套完整的服务,这使得它特别适合于公共网络中使用。这个版本的 Linux 也使用最新的内核,还拥有大多数人都需要使用的主体软件包
2	Fedora Core	www.redhat.com	拥有数量庞大的用户、优秀的社区技术支持,并且有许多创新
3	Debian Linux	www.debian.org	开放的开发模式,并且易于进行软件包升级
4	CentOS	www.centos.org	CentOS 是一种对 RHEL (Red Hat Enterprise Linux) 源代码再编译的产物,由于 Linux 是开发源代码的操作系统,并不排斥基于源代码的再分发,CentOS 就是将商业的 Linux 操作系统 RHEL 进行源代码再编译后分发,并在 RHEL 的基础上修正了很多已知的 BUG
5	SUSE Linux	www.suse.com	专业的操作系统,易用的 YaST 软件包管理系统开放
6	Mandriva	www.mandriva.com	操作界面友好,使用图形配置工具,有庞大的社区进行技术支持,支持 NTFS 分区的大小变更
7	KNOPPIX	www.knoppix.com	可以直接在 CD 上运行,具有优秀的硬件检测和适配能力,可作为系统的急救盘使用
8	Gentoo Linux	www.gentoo.org	高度的可定制性,使用手册完整
9	Ubuntu	www.ubuntu.com	优秀易用的桌面环境,基于 Debian 的不稳定版本构建

## 1.2.2 内核版本的含义及选择

内核是 Linux 操作系统的最重要的部分,从最初的 0.95 版本到目前的 4.x.xx.xx 版本, Linux 内核开发经过了近 30 年的时间,其架构已经十分稳定。Linux 内核的编号采用如下编号形式:

主版本号. 次版本号. 主补丁号. 次补丁号

例如:“2.6.32.67”各数字的含义如下。

- 第 1 个数字 (2) 是主版本号,表示第 2 大版本。
- 第 2 个数字 (6) 是次版本号,有两个含义:既表示是 Linux 内核大版本的第 6 个小版本,同时因为 6 是偶数也表示为发布版本。在 2.X 版本中奇数表示测试版,偶数表示稳定版,但是到了 3.X 版本中这个规则已经不适用了。
- 第 3 个数字 (32) 是主版本补丁号,表示指定小版本的第 32 个补丁包。
- 第 4 个数字 (67) 是次版本补丁号,表示次补丁号的第 67 个小补丁。

在安装 Linux 操作系统的时候,最好不要采用发行版本号中的小版本号是奇数的内核,因为开发中的版本没有经过比较完善的测试,有一些 BUG 是未知的,有可能造成使用中不必要的麻烦。

Linux 内核版本的开发源代码树目前最新版本已经到 4.X.XX.XX 版本,但是比较通用的是

2.6.xx 的版本,当然,有部分 2.4 的版本仍在使⤵用。与 2.4 版本的内核相比较,2.6 版本内核具有如下的优势。

- 支持绝大多数⤵的嵌入式系统,加入了之前嵌入式系统经常使用的 Linux 的大部分代码,并且子系统的支持更加细化,可以支持硬件体系结构的多样性,可抢占内核的调度方式支持实时系统,可定制内核。
- 支持目前最新的 CPU,例如 Intel 的超线程、可扩展的地址空间访问。
- 驱动程序框架变更,例如用 .ko 替代了原来的 .o 方式,消除内核竞争,更加透明的子模块方式。
- 增加了更多的内核级的硬件支持。

本书中的环境对 Linux 的内核没有特殊要求,因此读者在选择内核版本的时候不需要重新编译内核,使用操作系统自带的内核就可以满足需要。

下载 Linux 内核的网站为: <https://www.kernel.org>。

## 1.3 Linux 的系统架构及用途

Linux 系统从应用角度来看,分为内核空间和用户空间两个部分。内核空间是 Linux 操作系统的主要部分,但是仅有内核的操作系统是不能完成用户任务的。丰富并且功能强大的应用程序包是一个操作系统成功的必要条件。

### 1.3.1 Linux 内核的主要模块

Linux 的内核主要由 5 个子系统组成:进程调度、内存管理、虚拟文件系统、网络接口、进程间通信。下面依次讲解这 5 个子系统。

#### 1. 进程调度

进程调度 (SCHED) 指的是系统对进程的多种状态之间转换的策略。Linux 下的进程调度有 3 种策略: SCHED\_OTHER、SCHED\_FIFO 和 SCHED\_RR。

(1) SCHED\_OTHER 是用于针对普通进程的时间片轮转调度策略。这种策略中,系统给所有的运行状态的进程分配时间片。在当前进程的时间片用完之后,系统从进程中优先级最高的进程中选择进程运行。

(2) SCHED\_FIFO 是针对运行的实时性要求比较高、运行时间短的进程调度策略。这种策略中,系统按照进入队列的先后进行进程的调度,在没有更高优先级进程到来或者当前进程没有因为等待资源而阻塞的情况下,会一直运行。

(3) SCHED\_RR 是针对实时性要求比较高、运行时间比较长的进程调度策略。这种策略与 SCHED\_OTHER 的策略类似,只不过 SCHED\_RR 进程的优先级要高得多。系统分配给 SCHED\_RR 进程时间片,然后轮番运行这些进程,将时间片用完的进程放入队列的末尾。

由于存在多种调度方式, Linux 进程调度采用的是“有条件可剥夺”的调度方式。普通进程中采用的是 SCHED\_OTHER 的时间片轮循方式,实时进程可以剥夺普通进程。如果普通进程在用户空间运行,则普通进程立即停止运行,将资源让给实时进程。如果普通进程运行在内核空间,需要等系统调用返回用户空间后方可剥夺资源。



## 2. 内存管理

内存管理 (MMU) 是多个进程间的内存共享策略。在 Linux 系统中, 内存管理的主要概念是虚拟内存。

虚拟内存可以让进程拥有比实际物理内存更大的内存, 可以是实际内存的很多倍。每个进程的虚拟内存有不同的地址空间, 多个进程的虚拟内存不会冲突。

虚拟内存的分配策略是每个进程都可以公平地使用虚拟内存。虚拟内存的大小通常设置为物理内存的两倍。

## 3. 虚拟文件系统

虚拟文件系统 (Virtual File System, VFS) 存在于内核软件层, 是一个软件机制, 是物理文件系统与服务之间的一个接口层。它对 Linux 的每个文件系统的所有细节进行抽象, 使得不同的文件系统在 Linux 核心以及系统中运行的其他进程看来都是相同的。严格说来, VFS 并不是一种实际的文件系统, 它只存在于内存中, 不存在于任何外存空间。VFS 在系统启动时建立, 在系统关闭时消失。

VFS 支持文件系统主要有如下三种类型。

(1) 磁盘文件系统: 管理本地磁盘分区中可用的存储空间或者其他可以起到磁盘作用的的设备 (如 USB 闪存)。常见磁盘文件系统有 Ext2、Ext3、SystemV 和 BSD 等。

(2) 网络文件系统: 访问网络中其他计算机的文件系统所包含的文件。常用的网络文件系统有 NFS、AFS、CIFS 等。

(3) 特殊文件系统: 不管理本地或者远程磁盘空间。/proc 文件系统是特殊文件系统的典型的范例。

## 4. 网络接口

Linux 是在 Internet 飞速发展的时期成长起来的, 所以 Linux 支持多种网络接口和协议。网络接口分为网络协议和驱动程序, 网络协议是一种网络传输的通信标准, 而网络驱动则是对硬件设备的驱动程序。Linux 支持的网络设备多种多样, 几乎目前所有网络设备都有驱动程序。

## 5. 进程间通信

Linux 操作系统支持多进程, 进程之间需要进行数据的交流才能完成控制、协同工作等功能, Linux 的进程间通信是从 UNIX 系统继承过来的。Linux 下的进程间通信方式主要有管道方式、信号方式、消息队列方式、共享内存和套接字等方法。

# 1.3.2 Linux 的文件结构

与 Windows 下的文件组织结构不同, Linux 不使用磁盘分区符号来访问文件系统, 而是将整个文件系统表示成树状的结构, Linux 系统每增加一个文件系统都会将其加入到这个树中。

操作系统文件结构的开始, 只有一个单独的顶级目录结构, 叫作根目录。所有一切都从“根”开始, 用“/”代表, 并且延伸到子目录。DOS/Windows 下文件系统按照磁盘分区概念分类, 目录都存于分区上。Linux 则通过“挂接”的方式把所有分区都放置在“根”下各个目录里。Linux 系统的文件结构如图 1-4 所示。

不同的 Linux 发行版本的目录结构和具体的实现功能存在一些细微的差别。但是主要的功能都是一致的。一些常用目录的作用如下。

● /etc: 包括绝大多数 Linux 系统引导所需要的配置文件, 系统引导时读取配置文件, 按照

配置文件的选项进行不同情况的启动，例如 `fstab`、`host.conf` 等。

- `/lib`: 包含 C 编译程序需要的函数库，是一组二进制文件，例如 `glibc` 等。
- `/usr`: 包括所有其他内容，如 `src`、`local`。Linux 的内核就在 `/usr/src` 中。其下有子目录 `bin`，存放所有安装语言的命令，如 `gcc`、`perl` 等。
- `/var`: 包含系统定义表，以便在系统运行改变时可以只备份该目录，如 `cache`。
- `/tmp`: 用于临时性的存储。
- `/bin`: 大多数命令存放在这里。
- `/home`: 主要存放用户账号，并且可以支持 `ftp` 的用户管理。系统管理员增加用户时，系统在 `home` 目录下创建与用户同名的目录，此目录下一般默认有 `Desktop` 目录。
- `/dev`: 这个目录下存放一种设备文件的特殊文件，如 `fd0`、`had` 等。
- `/mnt`: 在 Linux 系统中，它是专门给外挂的文件系统使用的，里面有两个文件 `cdrom`、`floopy`，登录光驱、软驱时要用到。

刚开始使用 Linux 的人比较容易混淆的是 Linux 下使用斜杠“/”，而在 DOS/Windows 下使用的是反斜杠“\”。例如在 Linux 中，由于从 UNIX 集成的关系，路径用“`/usr/src/Linux`”表示，而在 Windows 下则用“`\usr\src\Linux`”表示。在 Linux 下更加普遍的问题是大小写敏感，这样字母的大小写十分重要，例如文件 `Hello.c` 和文件 `hello.c` 在 Linux 下不是一个文件，而在 Windows 下则表示同一个文件。

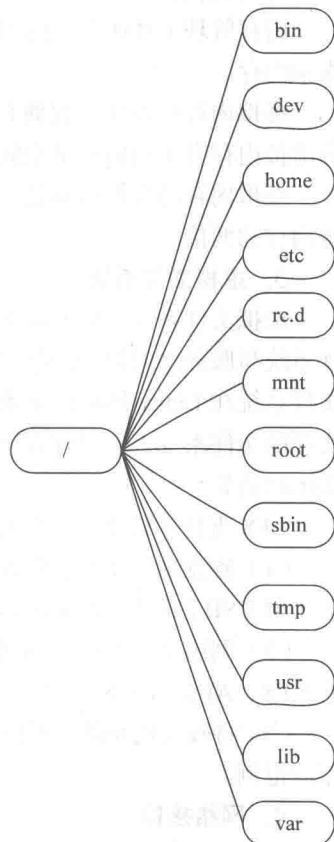


图 1-4 Linux 文件系统结构示意图

### 1.3.3 Linux 系统的用途

大多数企业和个人都不会使用 Linux 作为桌面操作系统，主要是用于后端服务器操作系统。经过一些大公司的大胆尝试，许多事实证明 Linux 完全可以担负起关键任务计算应用，并且有很多 Linux 系统从开始运行至今从未宕过机，100%的正常运行时间受到广大用户的青睐。Linux 系统的用途大致可以分为以下几类。

#### 1. 虚拟化

从桌面虚拟化到云，现在又回到桌面虚拟化，VMware 是虚拟化产品做得最早，也是目前最好的一家公司，现在它的主要产品也是基于 Linux 的，另外 Citrix、Red Hat 以及微软也是 VMware 的有力竞争者。

#### 2. 数据库服务器

Oracle 和 IBM 都有企业级软件运行在 Linux 上，因为它们在 Linux 上可以工作得很好，Linux 自身消耗的资源很少，因此它不会和数据库进行资源的抢夺，一个 RDBMS（关系数据库管理系统）需要一个稳定的、无内存泄露的、快速磁盘 I/O 和无 CPU 竞争的操作系统，Linux 就是这样的系统，世界上已经有很多开发人员使用 LAMP（Linux, Apache, MySQL, Perl/PHP/Python）



和 LAPP (Linux, Apache, PostgreSQL, Perl/PHP/Python) 作为开发平台, 也有很多关键应用系统是这么部署的。

### 3. Web 服务器

Apache 是世界上使用最多的 Web 服务器, 是企业公认的事实上的 Web 服务器标准, 几乎所有的平台都支持 Apache 服务器的运行, 但超过 90% 的 Apache 都是搭配 Linux 运行的。

### 4. 应用服务器

Tomcat, Geronimo, WebSphere 和 WebLogic 都是 Java 应用服务器, Linux 为这些服务提供了一个稳定的、内存消耗很小的、可长时间运行的平台。IBM 和 Oracle 也都非常支持 Linux, 它们也逐渐将 Linux 作为其软件系统的首选运行平台。

### 5. 跳转盒

对于企业而言, 跳转盒 (Jump box) 是一个为公共网络 (如互联网) 到安全网络 (如客户部) 提供的网关, 这样一个廉价的系统也可以为大量的用户提供服务, 而相对应的 Windows 系统需要非常昂贵的终端服务访问许可和客户端访问许可费用, 并且对硬件的要求更高。

### 6. 日志服务器

Linux 是处理和存储日志文件的绝佳平台, 看起来这是一个低级的任务, 但它的低成本、低硬件要求和高性能是任何需要日志服务的人的首选平台, 许多大公司也经常使用 Linux 作为日志服务的低成本平台。

### 7. 开发平台

Linux 下有许多开发工具, 如 Eclipse、C、C++、Mono、Python、Perl 和 PHP 等, 目前来看, Linux 是世界上最流行的开发平台之一, 它包含了成千上万的免费开发软件, 这对于全球开发者都是一个有利条件。

### 8. 监控服务

如果要做网络监控或系统性能监测, 那么 Linux 是一个非常好的选择, 大公司一般使用淘汰下来的硬件设备和自由软件搭建监控系统, 如 Orca 和 Sysstat 都是 Linux 上非常好的监控方案, IT 专业人员利用它们可以实现自动化监控, 无论网络的规模大小, 它们都能应付自如。

### 9. 入侵检测系统

Linux 同时也是一个完美的入侵检测服务平台, 因为它是免费的, 并且可以运行在多种硬件平台上, 同时也是开源爱好者喜欢的平台, Linux 上最著名的入侵防御和检测系统是 Snort, 它也是开源且免费的。

选择 Linux 的重要依据就是使用其的目的, 企业和个人用户应该根据自身的需求为自己定制最适合 Linux 发行版本, 以最大程度满足工作的需求, 保证服务的质量。

## 1.4 Linux 与 UNIX 的比较

Linux 是 UNIX 操作系统的一个克隆系统, 可以说没有 UNIX 就没有 Linux。但是 Linux 和传统的 UNIX 有很大的不同, 两者之间的最大区别是关于版权方面的: Linux 是开放源代码的自由软件, 而 UNIX 是对源代码实行知识产权保护的传统商业软件。两者之间主要有如下的区别。

(1) UNIX 操作系统大多数是与硬件配套的, 操作系统与硬件进行了绑定, 而 Linux 则可运