

高等学校应用型本科创新人才培养计划指定教材
高等学校计算机类专业“十三五”课改规划教材



Linux C

■ 程序设计

青岛农业大学
青岛英谷教育科技股份有限公司 编著



西安电子科技大学出版社
<http://www.xduph.com>

高等学校应用型本科创新人才培养计划指定教

高等学校计算机类专业“十三五”课改规划教

Linux C 程序设计

青岛农业大学

编著

青岛英谷教育科技股份有限公司

西安电子科技大学出版社

内 容 简 介

本书从 Linux 系统出发,以 CentOS 系统为背景详细介绍了 Linux 系统开发的环境及编程接口。全书共分为 7 章,分别介绍了 Linux 系统概述、开发环境搭建、文件编程、进程编程、线程编程、网络编程以及数据库编程等内容。

本书重点突出、应用性较强、内容简练、题材新颖、案例详实,理论性与实践性并重,在结合大量实例的基础上对 Linux 系统编程接口进行了细致的讲解与剖析,既涉及操作系统基本原理,又涉及操作系统接口实现,使读者能够迅速理解并掌握相关知识,全面提高动手能力。

本书可作为高校计算机相关专业的教材使用,也可作为有志于从事 Linux 系统开发工作的研究人员和相关工作者提供理论参考。

图书在版编目(CIP)数据

Linux C 程序设计 / 青岛农业大学, 青岛英谷教育科技股份有限公司编著.

—西安: 西安电子科技大学出版社, 2017.2

高等学校计算机类专业“十三五”课改规划教材

ISBN 978-7-5606-4422-6

I. ① L… II. ① 青… ② 青… III. ① Linux 操作系统—程序设计 IV. ① TP316.85

中国版本图书馆 CIP 数据核字(2016)第 316861 号

策 划 毛红兵

责任编辑 毛红兵 刘炳桢

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2017 年 2 月第 1 版 2017 年 2 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 16.75

字 数 392 千字

印 数 1~3000 册

定 价 42.00 元

ISBN 978-7-5606-4422-6 / TP

XDUP 4714001-1

如有印装问题可调换

高等学校计算机专业 “十三五”课改规划教材编委会

主编 王 蕊

编委 王 燕 于明进 柳永亮

李长明 赵长利 杨和利

唐述宏 侯崇升 张玉涛

◆◆◆ 前 言 ◆◆◆

随着计算机行业的迅猛发展，企业对应用型人才的需求越来越大。“全面贴近企业需求，无缝打造专业实用人才”是目前高校计算机专业教育的革新方向。

本书是面向高等院校计算机相关专业的标准化教材，涵盖 Linux 系统的基本概念、国际标准及编程接口等多方面内容。教材编写充分结合了 Linux 系统编程接口，经过了成熟的调研和论证，并参照多所高校一线专家的意见，具有系统性、实用性等特点。本书旨在使读者在系统掌握 Linux 系统基本概念的同时，着重培养他们解决实际问题的能力。

本书内容以培养读者对 Linux 系统的兴趣，使其熟悉 Linux 开发环境、掌握 Linux 编程接口为目标，在原有体制教育的基础上对课程进行改革，重点加强对 Linux 系统编程接口的学习。读者经过系统的学习后，可以了解车载操作系统的核心技术，掌握车载系统开发环境以及系统核心编程接口，具备投身于车载操作系统研发工作的专业能力，以及对前沿科技发展趋势的敏锐洞察力。

全书共分为 7 章，内容安排如下：

第 1 章简要阐述了 UNIX 系统、Linux 系统和 GNU 开源文化的关系，并对 Linux 系统的标准化、库函数以及系统调用进行了概述，为后面具体技术的学习奠定了基础。

第 2 章详述了 CentOS 系统的安装与配置步骤，系统讲解了 VIM 文本编辑器、GCC 程序编译器、GDB 程序调试器以及 Make 工程管理器的使用，简述了 Qt 和 Eclipse 两款图形化程序开发环境的搭建，为后续的编程学习搭建了必要的开发环境。

第 3 章详细比较了文件 IO 和标准 IO 的异同，并详细讲解了二者的使用及接口函数，详述了目录文件、链接文件等特殊文件的操作，对于编程中经常使用的临时文件也有所涉及。

第 4 章简要阐述了进程的基本概念、运行状态以及内存空间布局和进程创建、进程加载以及进程资源回收的实现，多方位讲解了进程间的通信方式，主要有管道（有名和无名）、信号、信号量、消息队列、共享内存以及内存映射等。

第 5 章简述了线程的基本概念，并分析比较了与进程的异同，详述了线程创建、系统终止和线程销毁等核心函数，并进一步挖掘了线程同步技术，包括互斥量、信号量、条件变量等。

第 6 章简述了计算机网络的基本概念和分类，详细剖析了 OSI 参考模型和 TCP/IP 实现模型，重点讲解了 Socket 网络编程模型，并区分了四种不同使用场景：UNIX Domain 报文、UNIX Domain 字节流、TCP 以及 UDP。

第 7 章简述了数据库的基本概念以及相关专业术语，详细讲解了 MySQL 数据库的安装与配置、MySQL 基本数据类型、SQL 基本语句、MySQL C 开发的函数接口以及编程步骤等。

本书的知识点合理分布于整套教材中，章节间衔接流畅，由浅入深，由总括到细分再

总揽全局，理论结合实际，充分满足各类读者的学习需求。同时，为了符合教学要求，本书在结构编排上进行了精心的设计：每章开始前设有学习目标，让读者可以有针对性地学习；同时，每章结束后还有小结和习题，可以加深读者对相关内容的理解和掌握。

本书由青岛农业大学与青岛英谷教育科技有限公司编写，参与本书编写的人员有：王蕊、卢玉强、宋乃华、邵舟、邓宇、王燕、宁维巍等。另外，在编写期间得到了青岛农业大学、潍坊学院、曲阜师范大学、济宁学院、济宁医学院等合作院校的专家及一线教师的大力支持和协作。在本书出版之际，特别感谢合作院校的师生给予我们的支持和鼓励，感谢开发团队每一位成员所付出的艰辛劳动与努力。

由于编者水平有限，书中难免有不妥之处，读者在阅读过程中如有发现，可以通过邮箱(yinggu@121ugrow.com)联系我们，以期不断完善。

本书编委会
2016年10月

◆◆◆ 目 录 ◆◆◆

第 1 章 Linux 系统概述	1
1.1 Linux 系统基本概念	2
1.1.1 Linux 系统特点	2
1.1.2 Linux 系统架构	3
1.1.3 Linux 系统的起源与发展	4
1.1.4 Linux 系统内核与发行版本	6
1.1.5 Linux 系统市场占有率分析	8
1.2 Linux 系统与 UNIX 系统的历史渊源	10
1.2.1 UNIX 系统基本概念	10
1.2.2 UNIX 系统起源与发展	11
1.2.3 UNIX 系统肩上的 Linux 系统	12
1.3 Linux 系统与 GNU 开源文化	13
1.3.1 GNU 基本概念	13
1.3.2 Linux 系统与 GNU 相辅相成	14
1.4 Linux 系统程序设计规范	15
1.4.1 Linux 系统标准化	15
1.4.2 系统调用和库函数	15
1.4.3 程序设计索引	16
小结	17
习题	18
第 2 章 开发环境搭建	19
2.1 CentOS 操作系统	20
2.1.1 CentOS 系统定制安装	20
2.1.2 CentOS 首次登录配置	27
2.1.3 CentOS 桌面环境介绍	30
2.1.4 CentOS 开发工具安装	30
2.2 VIM 文本编辑器	34
2.2.1 VIM 工作模式	35
2.2.2 VIM 操作流程	35
2.2.3 VIM 常用命令	36
2.3 GCC 程序编译器	38
2.3.1 GCC 文件类型	38
2.3.2 GCC 编译选项	38

2.4 GDB 程序调试器	40
2.4.1 GDB 调试命令	41
2.4.2 GDB 调试步骤	42
2.5 Make 工程管理器	43
2.5.1 Make 工具使用	43
2.5.2 Makefile 语法规则	43
2.6 Linux C 集成开发环境	44
2.6.1 Qt 集成开发环境	44
2.6.2 Eclipse 集成开发环境	53
小结	57
习题	58
第 3 章 文件编程	59
3.1 Linux 文件系统概述	60
3.1.1 文件管理系统	60
3.1.2 文件 IO 和标准 IO	61
3.1.3 文件描述符	61
3.1.4 流和 FILE 对象	62
3.1.5 缓冲机制	63
3.2 文件 IO	64
3.2.1 函数 open()	64
3.2.2 函数 close()	66
3.2.3 函数 read()	66
3.2.4 函数 write()	69
3.2.5 函数 lseek()	71
3.2.6 函数 fcntl()	72
3.2.7 函数 stat()	75
3.2.8 函数 access()	77
3.3 标准 IO	77
3.3.1 函数 fopen()	78
3.3.2 函数 fclose()	78
3.3.3 函数 fgetc()	79
3.3.4 函数 fputc()	79
3.3.5 函数 fgets()	80

3.3.6 函数 fputs()	82	4.3.6 共享内存	146
3.3.7 二进制文件	82	4.3.7 内存映射	151
3.3.8 文件流定位	84	小结	155
3.3.9 格式化输入/输出	84	习题	156
3.4 目录文件	85	第5章 线程编程	157
3.4.1 函数 mkdir()	86	5.1 线程概述	158
3.4.2 函数 rmdir()	87	5.1.1 线程的基本概念	158
3.4.3 函数 opendir()	87	5.1.2 线程与进程	159
3.4.4 函数 closedir()	88	5.1.3 Pthreads API 背景	160
3.4.5 函数 readdir()	88	5.2 线程控制	161
3.4.6 函数 getcwd()	90	5.2.1 线程标识	161
3.4.7 函数 chdir()	91	5.2.2 线程创建	162
3.5 链接文件	92	5.2.3 线程终止	164
3.5.1 硬链接	92	5.3 线程同步	168
3.5.2 软链接	97	5.3.1 同步概念	168
3.6 临时文件	99	5.3.2 互斥量	170
小结	103	5.3.3 条件变量	173
习题	104	5.3.4 信号量	177
第4章 进程编程	105	小结	180
4.1 Linux 文件系统概述	106	习题	180
4.1.1 进程的基本概念	106	第6章 网络编程	181
4.1.2 进程运行状态	107	6.1 计算机网络基础	182
4.1.3 进程状态切换	109	6.1.1 计算机网络的作用与意义	182
4.1.4 进程启动	110	6.1.2 计算机网络的起源与发展	182
4.1.5 进程终止	110	6.1.3 计算机网络的覆盖范围	184
4.1.6 程序存储空间布局	111	6.1.4 计算机网络的拓扑结构	185
4.2 进程控制	112	6.1.5 OSI 参考模型	188
4.2.1 进程标识	112	6.2 TCP/IP 协议基础	190
4.2.2 进程创建	115	6.2.1 TCP/IP 基本概念	190
4.2.3 父子进程	116	6.2.2 TCP/IP 网络模型	192
4.2.4 进程资源回收	119	6.2.3 TCP 和 UDP 协议	193
4.2.5 进程体替换	119	6.2.4 IP 协议	195
4.2.6 调用命令行	122	6.2.5 字节序	198
4.3 进程间通信	124	6.3 Socket(套接字)编程基础	198
4.3.1 管道	124	6.3.1 Socket 地址结构	199
4.3.2 FIFO	126	6.3.2 Socket 地址转换	200
4.3.3 信号	131	6.3.3 Socket 基本属性	202
4.3.4 消息队列	134	6.3.4 Socket 系统调用	203
4.3.5 信号量	140	6.4 UNIX Domain	206

6.4.1 报文 Socket 编程模型	206	7.1.1 MySQL 专业术语概述	230
6.4.2 流式 Socket 编程模型	209	7.1.2 MySQL 基本数据类型	231
6.5 Internet Domain	212	7.1.3 MySQL 开发环境搭建	234
6.5.1 UDP 编程模型	212	7.1.4 MySQL 常用 SQL 语句	237
6.5.2 TCP 编程模型	216	7.2 MySQL 的 C 编程接口	243
6.6 网络编程实例	220	7.2.1 执行查询语句	243
6.6.1 基于 TCP 的文件接收服务器	220	7.2.2 提取查询结果	247
6.6.2 基于 TCP 的文件发送客户端	224	小结	256
小结	227	习题	256
习题	228	参考文献	258
第 7 章 数据库编程	229		
7.1 MySQL 开发基础	230		

第1章 Linux 系统概述



本章目标

- 了解 Linux 系统基本概念及其特点
- 掌握 Linux 系统内核版本命名规则
- 了解 Linux 系统常见发行版本
- 了解 Linux 系统、UNIX 系统和 GNU 计划的基本关系
- 掌握系统调用和库函数的基本关系
- 掌握 C 语言标准头文件和库文件的存储路径



1.1 Linux 系统基本概念

随着开源软件在世界范围内影响力的日益增强，Linux 系统在服务器、桌面计算机、行业定制等领域获得了长足发展，尤其在服务器领域，已经取得了令人瞩目的成就。据统计，全球 TOP500 超级计算机中，超过 97% 的超级计算机运行在 Linux 操作系统之上。

1.1.1 Linux 系统特点

Linux 系统是一套免费使用和自由传播的开源操作系统，是一个基于 POSIX 和 UNIX 系统的多用户、多任务、多线程和多 CPU 的操作系统。Linux 系统支持 32 位和 64 位硬件，同时能兼容绝大多数的 UNIX 系统软件工具、应用程序和网络协议。Linux 系统始终坚持以网络为核心的设计思想，是一个性能稳定的多用户网络操作系统。

Linux 系统的基本设计思想有两点：

第一，一切皆文件。系统中一切都归结为一个特定的文件，包括命令、硬件/软件设备、OS、进程等，对于操作系统内核而言，都被视为拥有各自特性或类型的文件。

第二，每个软件都有确定的用途，同时它们都应该尽可能被编写得更好。

Linux 系统的作者是 Linus Torvalds，如图 1-1 所示。Linux 系统诞生至今已有 20 多年，之所以得到如此长远的发展，与其自身的特点是息息相关的：

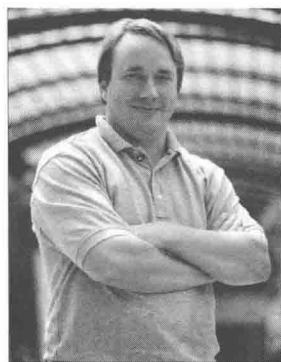


图 1-1 Linus Torvalds

- ✧ 完全免费。Linux 系统是一款免费的操作系统，用户可以通过网络或其他途径免费获得，并可以任意修改其源代码，这是其他操作系统做不到的。正是由于这一点，来自全世界的无数程序员参与了 Linux 系统的修改、编写工作。程序员可以根据自己的兴趣和灵感对 Linux 系统进行改变，这使其吸收了无数程序员的精华创作，自身不断壮大。
- ✧ 完全兼容 POSIX 标准。这说明可以在 Linux 系统下通过相应的模拟器运行常见的 DOS、Windows 的程序。这为用户从使用 Windows 系统转到使用 Linux 系统奠定了基础。
- ✧ 支持多用户、多任务。Linux 系统支持多用户，各个用户对于自己的文件设备有自己特殊的权利，保证了各用户之间互不影响；多任务则是现代计算机最主要的一个特点，Linux 系统可以使多个程序同时并独立地运行。
- ✧ 拥有良好的界面。Linux 系统同时具有字符界面和图形界面。在字符界面用户可以通过键盘输入相应的指令来进行操作。Linux 系统同时也提供了类似 Windows 图形界面的 X-Window 系统，用户可以使用鼠标对其进行操作。
- ✧ 具有丰富的网络功能。Linux 系统中，用户可以轻松实现网页浏览、文件传输、



远程登录等网络工作。此外，Linux 系统还可作为服务器提供 WWW、FTP、E-mail 等服务。

- ✧ 具有可靠、安全的稳定性能。Linux 系统采取了多种安全技术措施，其中包括对读/写权限的控制、审计跟踪、核心授权等技术。Linux 系统在稳定性方面做得十分出色，完全可以胜任服务器对稳定性的需求。
- ✧ 支持多种平台。Linux 系统兼容多种计算机硬件架构，如 x86、PowerPC、Alpha 等处理器。此外，Linux 系统作为嵌入式操作系统还兼容 ARM、MIPS 等处理器架构，可以运行在手机、平板、机顶盒等设备上。
- ✧ 具有设备独立性。Linux 系统将所有外部设备统一当成文件来处理，只要安装设备驱动程序，任何用户都可以像使用文件一样操纵这些设备，而不必知道设备的具体存在形式。

1.1.2 Linux 系统架构

从应用角度来看，Linux 系统分为内核空间和用户空间两部分，如图 1-2 所示。用户空间主要由丰富且功能强大的应用程序和底层 C 函数库构成；作为 Linux 系统的主要部分，内核空间主要由进程调度、内存管理、文件系统、网络接口、进程间通信五个子系统构成，如图 1-3 所示。

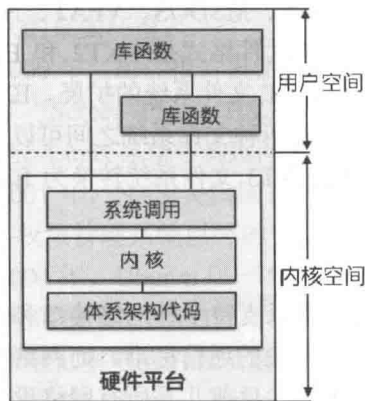


图 1-2 Linux 系统内核空间与用户空间

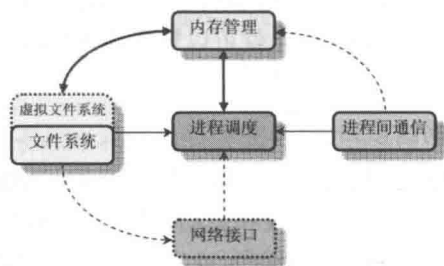


图 1-3 Linux 内核模块结构和依赖关系

1. 进程调度

进程调度指的是系统对进程的多种状态之间转换的策略。Linux 系统下的进程调度有三种策略：SCHED_OTHER、SCHED_FIFO 和 SCHED_RR。

- ✧ SCHED_OTHER 是针对普通进程的时间片轮转调度策略。在这种策略中，系统给所有处于“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，系统从进程中选择优先级最高的进程进行运行。
- ✧ SCHED_FIFO 是针对实时性要求比较高、运行时间短的进程的调度策略。在这种策略中，系统会按照进入队列的先后顺序进行进程的调度。当前进程在没有更高优先级进程到来或者没有其他等待资源阻塞的情况下，会一直运行



下去。

- ◇ **SCHED_RR** 是针对实时性要求比较高、运行时间比较长的进程的调度策略。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，然后轮循运行这些进程，将时间片用完的进程放入队列的末尾。

Linux 进程调度采用的是“有条件可剥夺”的调度方式。普通进程采用的是 **SCHED_OTHER** 的时间片轮循方式，实时进程可以剥夺普通进程的资源。如在用户空间，普通进程被迫立即停止运行，将资源让给实时进程；在内核空间，实时进程需要等系统调用返回用户空间后方可剥夺普通进程的资源。

2. 内存管理

内存管理是指多个进程间的内存共享策略。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. 文件系统

Linux 支持多种文件系统，如 **EXT2/EXT3/EXT4**、**XFS**、**MSDOS**、**VFAT**、**NTFS**、**PROC**、**TMPFS**、**ISO9660** 等。目前 Linux 系统下最常用的文件格式是 **EXT2** 和 **EXT3**。**EXT2** 文件系统用于固定文件系统和可活动文件系统，是 **EXT** 文件系统的扩展。**EXT3** 文件系统是在 **EXT2** 上增加日志功能后的扩展，兼容 **EXT2**。两种文件系统之间可以互相转换，**EXT2** 不用格式化就可以转换为 **EXT3** 文件系统，而 **EXT3** 文件系统转换为 **EXT2** 文件系统也不会丢失数据。

4. 网络接口

Linux 系统是在互联网飞速发展的时期成长起来的，所以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，目前几乎所有网络设备都有驱动程序。

5. 进程间通信

Linux 系统支持多进程，进程之间需要进行数据的交流才能实现控制、协同工作等功能。Linux 的进程间通信方式是从 **UNIX** 系统继承过来的，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。

1.1.3 Linux 系统的起源与发展

1987 年，**MINIX** 系统发布，并将全部源代码免费开放给大学教学和研究工作。这在软件极其昂贵的当时得到了广泛的关注和传播。



1991 年, Linus Torvalds 在芬兰赫尔辛基大学学习期间, 想通过增加 MINIX 系统的功能来提升性能, 但受限于 Tanenbaum。Linus Torvalds 遂决定写一个内核, 并将这一决定发布在 comp.os.minix 新闻组上, 得到了全世界计算机爱好者的支持。同年 9 月 1 日, Linux v0.01 版本发布。

1992 年, Linux v0.12 版本开始, Linus 使用 GNU GPL 作为 Linux 系统的版权声明, 将 Linux 系统奉献给自由软件, 从而铸就了包括 Linux 系统在内的自由软件今天的辉煌。

1994 年, Linux v1.0 版本发布, 开始支持基于 i386 的单处理器。Red Hat 公司以 49.95 美元的零售价格出售 Red Hat Software Linux 的 CD-ROM 和 30 天的安装支持。截止到 2012 年, Red Hat 公司成为第一家市值达 10 亿美元的开源公司。

1995 年, 召开了第一个专门针对 Linux 的贸易和会议——Linux Expo 的年度盛会, 备受关注。

1996 年, 企鹅 Tux 被选定成为 Linux 系统的吉祥物, 如图 1-4 所示。Linux v2.0 发布, 开始支持多处理器架构。同年, Mattias Ettrich 发起了 KDE 项目, 致力于 Linux 系统图形桌面的开发。

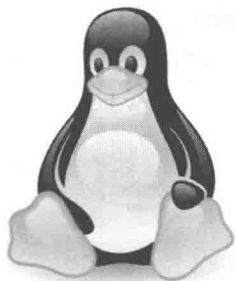


图 1-4 Linux 吉祥物 Tux

1998 年, Google 搜索引擎面世。

1999 年, GNOME 桌面进入 Linux 系统, 很多 Linux 系统发行版都将 GNOME 桌面作为默认的桌面环境。

2000 年, IDC 报告表明 Linux 排在“最受欢迎的服务器操作系统的第 2 位”。爱立信公布了“Screen Phone HS210”, 这是一款基于 Linux 的触屏手机, 具备邮件和网页浏览等功能。Linux 系统咨询顾问 Klaus Knopper 发布了第一个 Linux Live 发行版——Knoppix。

2001 年, NAS(美国国家安全局)以 GPL 许可证发布了 SELinux, SELinux 提供了标准 UNIX 权限管理系统以外的安全检查层。

2004 年, Ubuntu 以一个不同寻常的版本号 4.10 和怪异的版本代号“Warty Warthog”(长满疙瘩的非洲疣猪)发布。Ubuntu 虽然不是内核的主要贡献者, 然而对于 Linux 系统台式机和笔记本电脑的普及起到了重要作用。

2007 年, Linux 基金会由 OSDL(开源发展实验室)和 FSG(自由标准组织)联合成立, 该基金会得到了世界各地开发者及众多计算机公司的支持, 其中包括 Fujitsu、HP、IBM、Intel、NEC、Oracle、Qualcomm、Samsung 等。

2008 年, 在 Google I/O 大会上, Google 提出了 Android HAL 架构图。同年 8 月 18 号, Android 获得了 FCC(美国联邦通信委员会)的批准。当年 9 月份, 谷歌正式发布了基于 Linux 系统内核的 Android 1.0 系统。

2011 年, Google I/O 大会发布了 Chromebook, 这是一款运行着 Chrome OS 云操作系统的笔记本, Chrome OS 是基于 Linux 系统内核的云操作系统。同年 Linus 发布了 Linux v3.0 版本。

2013 年, Linux 系统进入车联网, 运行在许多汽车的中控台上, 各大汽车公司都选择 Linux 系统作为汽车的车载信息娱乐平台。

2015 年, Linus 发布了 Linux v4.0 版本, 开启了 Linux 系统内核 Live Patching(实时补



丁)机制,实现了系统内核在无需重启的情况下自动更新的功能。

1.1.4 Linux 系统内核与发行版本

严格意义上, Linux 这个词本身只表示 Linux 内核,但实际上人们已经习惯了用 Linux 来形容整个基于 Linux 内核并且使用 GNU 工程的各种工具和数据库的操作系统。市面上虽存在着不同版本的 Linux 系统,但它们都是基于 Linux 内核的。Linux 系统可安装在各种计算机硬件设备中,如手机、平板、路由器、视频游戏控制台、计算机、大型机和超级计算机等。

Linux 内核源代码可在官方网站“The Linux Kernel Archives”上免费获取,其具体网址为“www.kernel.org”,网站首页如图 1-5 所示。



图 1-5 Linux 内核官网首页

Linux 内核的版本称为 Linux 内核版本。Linux 内核版本具有一定的命名规则,通常来说,由下列形式构成:

VERSION.PATCHLEVEL.SUBLEVEL.EXTRAVERSION

其中:

- ✧ VERSION 为主版本号,有结构性的变化时才会更改。
- ✧ PATCHLEVEL 为次版本号,有新功能时会更改,偶数为稳定版,奇数为测试版。
- ✧ SUBLEVEL 为修订版本号,表示对此版本的内核的修订次数或补丁包数。
- ✧ EXTRAVERSION 为扩展版本号,表示此版本用于修订最新稳定版出现的问题。

Linux 内核以开源代码形式对外发布,供人们免费下载。各 Linux 发行商在得到



Linux 内核源码之后,自行完成对内核的裁剪优化,搭配特定需求的应用软件,然后对外发行。这种以 Linux 内核为基础,装配了一些特定程序库或工具所发行的系统,就是所谓的 Linux 系统发行版,其中比较典型的几个 Linux 系统发行版如图 1-6 所示。每种 Linux 系统发行版都有自己的设计理念,以及所针对的不同客户群。



图 1-6 典型的 Linux 系统发行版本

1. Ubuntu

Ubuntu 是一款以桌面应用为主的 Linux 系统,默认桌面为 Unity。Ubuntu 名称来自非洲南部祖鲁语,意思是“人性”,是非洲一种传统的价值观。Ubuntu 每六个月发布一个新版本,一般在每年的 4 月份发布测试版系统,10 月份发布与之对应的稳定版。Ubuntu 旨在为用户提供一个最新的、自由的、稳定的操作系统。Ubuntu 具有庞大的社区力量,用户可以方便地从社区获得帮助。2013 年 1 月 3 日,Ubuntu 正式发布面向智能手机的移动操作系统。

中国 CCN 联合实验室以 Ubuntu 为参考,建立和主导了开源项目——Ubuntu Kylin。为了满足广大中文用户的特定需求,该项目将国际化平台与本地化应用相融合,开发出了最具中国特色的桌面系统和操作系统。这一项目得到了来自 Debian、Ubuntu、LUPA 等国内外众多社区爱好者的广泛参与和热情支持。

2. Fedora

Fedora 是 Red Hat 公司推出的社区版 Linux 系统,其被定位为 Red Hat 新技术的实验场,每年发布两个新版本。每一版本的 Fedora 都会引入大量的新技术,待测试稳定之后加入到 Red Hat Enterprise Linux 中。从 21 版本开始,Fedora 分化为三个子版本: Fedora Cloud、Fedora Server 和 Fedora Workstation。

Fedora 默认采用 RPM 软件包管理方式,使得系统在安装、升级、删除和管理软件方面变得十分容易。Fedora 对 Linux 内核、glibc 以及 gcc 所做出的贡献是有目共睹的,其对于 SELinux 功能、虚拟化技术、系统服务管理器、前沿日志文件系统以及其他企业级功能的整合也赢得了公司客户的高度赞赏。

3. openSUSE

openSUSE 前身为 SUSE Linux 和 SUSE Linux Professional,其用户界面非常华丽,而



且性能良好，在欧洲的发行量常年占据第一位。openSUSE 采用 KDE 作为默认桌面环境，其软件包管理系统采用自主开发的 YaST，颇受好评。

openSUSE 不仅是一款优秀的桌面系统，作为中小型企业服务器，它也具有十分明显的优势。YaST2 软件包管理系统可以使服务器的配置更加简单和快捷。作为默认的防入侵系统，AppArmor 从内到外保护操作系统和应用程序。SELinux 安全机制的引入大大增强了系统的安全性。

4. Debian

Debian 最早是由 Ian Murdock 于 1993 年创建并发起的，至今已经发展成为最大的非商业 Linux 发行版。Debian 是一款完全由社区驱动的 Linux 发行版，目前支持的内核有 Linux、FreeBSD 和 Hurd。

Debian 的发行及其软件源有五个分支：oldstable(旧稳定分支)、stable(稳定分支)、testing(测试分支)、unstable(不稳定分支)以及 experimental(实验分支)。Debian 默认采用 apt-get/dpkg 的软件包管理方式：apt-get 工具用于从远程服务器获取软件包以及处理复杂的软件包依赖关系；dpkg 工具用于本地软件包的安装、更新以及删除。这种软件包管理方式使得 Debian 系统在安装、升级、删除和管理软件方面变得十分容易。

5. Red Hat Enterprise Linux

Red Hat Enterprise Linux 是 Red Hat 公司发行的企业级 Linux 系统。该系列有三个版本：Red Hat Enterprise Linux AS、Red Hat Enterprise Linux ES 以及 Red Hat Enterprise Linux WS。目前最新的版本为 Red Hat Enterprise Linux 7，该版本是基于 Fedora 19 发展而来的，并引入了很多新的特性：内核版本升级为 Linux Kernel 3.10；默认文件系统为 XFS 格式；系统及服务方面使用 systemd，取代了传统的 SysV；引入了 Docker 容器等。

Red Hat 公司并不售卖软件，而是依靠订阅红帽服务营利，订阅服务给了用户极大的灵活性。订阅服务尝试给用户提供了高质量的服务，而不是某个产品的正式版本。Red Hat Enterprise Linux 生命周期非常长，Red Hat 公司更是将 5、6、7 三个版本系统的支持年限从原有的 7 年增加到 10 年，为用户长期部署操作系统提供了保障。

6. CentOS

CentOS(Community Enterprise Operating System)来自 Red Hat Enterprise Linux 依照开放源代码规定释出的源代码所编译而成。由于出自同样的源代码，因此有些要求高度稳定性的服务器，会选择使用 CentOS 替代商业版的 Red Hat Enterprise Linux 使用。两者的不同之处在于 CentOS 并不包含封闭源代码软件。

2014 年，Red Hat 公司完成对 CentOS 公司的收购。收购之后，CentOS 和 Red Hat Enterprise Linux 之间的关系保持不变，即社区成员和贡献者继续对 CentOS 所做的贡献是独立于 Red Hat Enterprise Linux 的，并且 CentOS 仍会使用 Red Hat Enterprise Linux 作为上游。

1.1.5 Linux 系统市场占有率分析

经过多年的发展，Linux 系统已经在服务器、桌面计算机、嵌入式等领域取得了令人