



计算机“十二五”规划教材

Linux操作系统 及应用技术

主编◎顾润龙 刘智涛 侯玉香



航空工业出版社

计算机“十二五”规划教材

Linux 操作系统及 应用技术

主编 顾润龙 刘智涛 侯玉香

航空工业出版社

北 京

内 容 提 要

Linux 以基于 Intel x86 架构 CPU 为硬件平台,是与 UNIX 兼容的多用户、多任务操作系统,具有很好的稳定性和安全性,并具有强大的网络服务功能,是目前商用服务器的首选操作系统之一。

本书以 Red Hat 公司最新版本 Red Hat Enterprise Linux 7.2 为平台,对 Linux 的网络服务应用进行了详细的讲解。全书以“理论够用、实践第一”为原则,力求使读者能够快速、轻松地掌握 Linux 技术与应用。本书内容包括 Linux 服务器搭建与测试、DHCP 服务器搭建与应用、DNS 服务器搭建与应用、FTP 服务器搭建与应用、Web 服务器搭建与应用、防火墙服务器搭建与应用等。

本书可作为大中专院校计算机类和信息技术类专业教材,也可作为 Linux 爱好者的参考书和培训教材。

图书在版编目(CIP)数据

Linux 操作系统及应用技术 / 顾润龙, 刘智涛, 侯玉
香主编. -- 北京: 航空工业出版社, 2016.8
ISBN 978-7-5165-1084-1

I. ①L… II. ①顾… ②刘… ③侯… III. ①Linux 操
作系统—程序设计—教材 IV. ①TP316.85

中国版本图书馆 CIP 数据核字(2016)第 206737 号

Linux 操作系统及应用技术 Linux Caozuo Xitong ji Yingyong Jishu

航空工业出版社出版发行

(北京市朝阳区北苑 2 号院 100012)

发行部电话: 010-84936597 010-84936343

三河市祥达印刷包装有限公司印刷

全国各地新华书店经售

2016 年 8 月第 1 版

2016 年 8 月第 1 次印刷

开本: 787×1092

1/16

印张: 17.75

字数: 399 千字

印数: 1—3000

定价: 39.80 元

前 言

Linux 是 1991 年发展起来的与 UNIX 兼容的多用户、多任务分时操作系统,是以 UNIX 操作系统为原型而设计的一种类 UNIX 操作系统,其网络功能十分强大,具有运行稳定、可靠性高、安全性好、开放性和移植性好、支持跨平台运行等优点。

Linux 既可作为客户端的桌面操作系统使用,也可作为服务器操作系统使用。Linux 的网络服务功能十分强大,可成为功能强大的 Web 服务器、FTP 服务器、DNS 服务器、DHCP 服务器、流媒体服务器、代理服务器、防火墙或路由器。Windows 平台的 DNS 服务器无法实现策略 DNS 解析,但利用 Linux 平台的 BIND DNS 服务器软件,通过简单的配置,即可实现功能强大的、具有策略 DNS 解析能力的 DNS 服务器。因此,Linux 常用作网络服务器操作系统。

根据 IDC 的报告统计,全球 Linux 市场的年均增长率为 44.0%。今天,Linux 已进入了企业的关键性业务应用领域,包括数据库、电子邮件、防火墙、应用软件开发、Web 服务等。由于市场需求强烈,很多大中专院校都开设了 Linux 课程,同时在教学过程中更加注重学生岗位能力的培养,以“实用为主、够用为度”有针对性地进行职业技能的培养与训练。

本教材以 Linux 7.2 发行版为例,将 Linux 作为桌面操作系统进行了详细介绍。全书共分 14 章,内容包括 Linux 简介、Linux 操作系统的安装与启动、常用 Linux 命令、vi 编辑器、shell 编程、用户与用户组管理、网络配置、Samba 跨平台资源共享的管理及应用、DHCP 服务器的搭建及应用、BIND 域名解析服务器的搭建及应用、Apache Web 服务器的搭建及应用、MySQL 数据库服务器的搭建及应用、vsftpd FTP 服务器的搭建及应用、防火墙与 Squid 代理服务器的搭建及应用。

本书由顾润龙、刘智涛、侯玉香任主编,刘文、原锦明任副主编。其中,第 1、4、5、6、8 章由顾润龙编写,第 11、12、13、14 章由刘智涛编写,第 2、10 章由侯玉香编写,第 7、9 章由刘文编写,第 3 章由原锦明编写,全书由顾润龙统稿。

本书在编写过程中参考了互联网上公布的一些有关资料,由于互联网上的资料较多,引用复杂,无法一一注明原出处,故在此声明,原文版权属于原作者。其他参考文献在本书后列出。由于作者水平有限,书中难免存在疏漏和错误之处,恳请读者批评指正,以期修订更新。

顾润龙
2016 年 7 月

C 目录

atalog.....

第 1 章 Linux 简介	1	3.2.4 修改命令行输入错误	36
章前导读	1	3.2.5 特殊控制符	36
本章要点	1	3.3 文件操作命令	36
1.1 Linux 发展史	1	3.3.1 文件系统概念	36
1.2 Linux 的特点	4	3.3.2 文件显示命令	39
1.3 Linux 版本编号方式	6	3.3.3 匹配、排序命令	41
1.4 Linux 文件结构	7	3.3.4 比较文件内容的命令	44
1.5 常见的 Linux 发行版本	9	3.3.5 复制、删除和移动 文件的命令	46
第 2 章 Linux 操作系统的 安装与启动	14	3.3.6 文件内容统计命令	48
章前导读	14	3.4 目录及其操作命令	48
本章要点	14	3.4.1 目录结构	48
2.1 安装 Linux 操作系统	14	3.4.2 创建和删除目录的命令	50
2.1.1 光盘安装 Linux 系统	14	3.4.3 改变工作目录和显示 目录内容命令	51
2.1.2 使用虚拟机安装 Linux	24	3.4.4 链接文件的命令	52
2.2 Linux 登录与注销	28	3.4.5 改变文件或目录存 取权限的命令	52
第 3 章 常用 Linux 命令	29	3.4.6 改变用户组和 文件主的命令	54
章前导读	29	3.5 文件压缩命令	55
本章要点	29	3.6 进程控制命令	57
3.1 系统安全性	29	3.6.1 查看系统的进程	57
3.1.1 注册和退出系统	30	3.6.2 进程控制	58
3.1.2 修改口令	32	3.7 网络命令	58
3.1.3 良好的保密习惯	33	3.8 其他命令	63
3.2 命令的使用	34	3.9 联机帮助命令	65
3.2.1 bash 简介	34		
3.2.2 命令格式	35		
3.2.3 键入命令	35		

第 4 章 vi 编辑器	67	5.1 shell 概述	91
章前导读	67	5.2 创建和执行 shell 脚本	93
本章要点	67	5.3 shell 特殊字符	95
4.1 vi 的工作方式	67	5.3.1 引号	95
4.1.1 命令方式	67	5.3.2 输入/输出重定向	98
4.1.2 输入方式	68	5.3.3 注释和管道线	101
4.1.3 ex 转义方式	69	5.3.4 命令执行操作符	101
4.2 vi 的启动和退出	69	5.4 shell 变量	103
4.2.1 启动 vi	69	5.4.1 用户变量	103
4.2.2 退出 vi	70	5.4.2 内部变量	104
4.3 文本输入	71	5.4.3 环境变量	104
4.3.1 插入命令	71	5.4.4 位置参数	105
4.3.2 附加命令	72	5.5 正则表达式与算术运算	108
4.3.3 打开命令	72	5.5.1 正则表达式	108
4.3.4 输入方式下光标移动	73	5.5.2 算术运算	109
4.4 打开文件	74	5.6 控制结构	111
4.4.1 打开一个文件	74	5.6.1 条件语句	111
4.4.2 打开多个文件	75	5.6.2 循环语句	118
4.4.3 用 vi 打开多个窗口	77	5.7 其他语句	122
4.5 保存文件	78	5.7.1 break 语句	122
4.6 移动光标	79	5.7.2 continue 语句	123
4.7 文本修改	80	5.7.3 exit 语句	123
4.7.1 删除与替换	80	5.8 函数	123
4.7.2 查找	82	5.9 调试 shell 脚本	125
4.7.3 复制、粘贴、剪切	83	5.10 实例	126
4.7.4 重复	85	习 题	127
4.7.5 撤销	86	第 6 章 用户与用户组管理	129
4.8 ex 命令	86	章前导读	129
4.8.1 命令定位	87	本章要点	129
4.8.2 常用 ex 命令	87	6.1 Linux 系统用户账号的管理	129
习 题	90	6.1.1 添加新用户	129
第 5 章 shell 编程	91	6.1.2 修改账号	130
章前导读	91	6.1.3 删除帐号	132
本章要点	91	6.1.4 用户口令管理	133

6.2 Linux 系统用户组的管理	134
6.2.1 增加用户组	134
6.2.2 修改用户组属性	135
6.2.3 删除用户组	136
6.2.4 用户组设置	137
6.3 查看用户和用户组状态	138
6.3.1 查看用户	138
6.3.2 who 命令	139
6.3.3 w 命令	139
6.3.4 whoami 命令	140
第 7 章 网络配置	141
章前导读	141
本章要点	141
7.1 网络配置命令	141
7.2 常用网络配置文件	152
第 8 章 Samba 跨平台资源共享的 管理及应用	156
章前导读	156
本章要点	156
8.1 Samba 简介	156
8.2 Samba 服务的安装与 运行控制	159
8.3 认识 Samba 服务的配置文件	160
8.4 配置可匿名访问的文件共享	164
8.5 配置带验证的文件共享	165
8.6 Samba 服务器扩展功能配置	165
8.7 Linux 与 Windows 资源互访	167
8.8 配置 Samba 打印共享	167
第 9 章 DHCP 服务器的 搭建及应用	169
章前导读	169
本章要点	169
9.1 DHCP 服务概述	169
9.1.1 DHCP 服务简介	169
9.1.2 DHCP 工作原理	170
9.2 DHCP 服务器的安装	172
9.3 配置单子网的 DHCP 服务	173
9.4 配置多子网的 DHCP 服务	175
9.5 用中继代理实现跨网段的 DHCP	178
9.5.1 为什么需要 DHCP 中继代理	178
9.5.2 配置 DHCP 中继 代理方式	178
9.5.3 配置 DHCP 中继代理的 步骤	178
第 10 章 BIND 域名解析服务器的 搭建及应用	181
章前导读	181
本章要点	181
10.1 DNS 服务及域名空间	181
10.1.1 什么是 DNS	181
10.1.2 DNS 的结构	182
10.1.3 理解 DNS 的工作方式	183
10.1.4 DNS 服务器的类型	184
10.2 DNS 服务的安装与运行	184
10.2.1 获得 BIND 软件包	184
10.2.2 检查是否已安装 BIND 软件包	185
10.2.3 安装 BIND 软件包	185
10.2.4 DNS 服务的运行管理	185
10.3 配置主 DNS 服务	186
10.4 配置辅助 DNS 服务器	196
10.5 配置纯缓存 DNS 服务器	197
10.6 配置 DNS 服务的转发器	198
第 11 章 Apache Web 服务器的 搭建及应用	199
章前导读	199

本章要点·····	199	13.2 vsftpd 服务器安装与测试·····	229
11.1 Web 服务简介·····	199	13.3 认识 vsftpd 的配置文件·····	233
11.1.1 Web 服务的历史和 工作原理·····	199	13.4 基于匿名用户访问的 FTP 配置·····	241
11.1.2 Apache 简介·····	200	13.5 基于本地用户访问的 FTP 配置·····	244
11.2 Web 服务器安装·····	201	13.6 基于虚拟用户访问的 FTP 配置·····	246
11.2.1 Apache 安装方法·····	201	第 14 章 防火墙与 Squid 代理 服务器的搭建及应用·····	249
11.2.2 认识 Apache 的目录 和文件·····	204	章前导读·····	249
11.3 用虚拟目录为多部门 建子网站·····	213	本章要点·····	249
11.4 使用虚拟主机实现 一机多站·····	214	14.1 防火墙·····	250
11.5 Web 服务的访问控制·····	218	14.1.1 认识防火墙 (firewall)·····	250
11.6 为系统用户建立个人 主页空间·····	220	14.1.2 Linux 防火墙概述·····	252
第 12 章 MySQL 数据库服务器的 搭建及应用·····	221	14.1.3 iptables 规则的 分层结构·····	253
章前导读·····	221	14.1.4 数据包过滤匹配流程·····	254
本章要点·····	221	14.1.5 代理服务器 Squid·····	254
12.1 基本概念·····	221	14.2 iptables 服务的安装·····	255
12.1.1 数据库服务器的 基本概念·····	221	实例 1——管理 icmp·····	261
12.1.2 MySQL 简介·····	222	实例 2——设置远程登录限制·····	261
12.2 MySQL 服务器的安装·····	223	实例 3——作为专门 Web 服务器 终端的配置·····	261
12.3 MySQL 服务器的运行管理·····	224	14.3 使用 iptables 实现 NAT 服务·····	263
12.4 MySQL 的基本操作·····	224	14.4 Squid 服务器的安装·····	271
12.5 用户与权限的管理·····	226	14.5 认识 Squid 配置参数与 初始化·····	271
12.6 数据库的备份与恢复·····	227	14.6 普通代理服务器的配置·····	273
第 13 章 vsftpd FTP 服务器的 搭建及应用·····	228	14.7 透明代理服务器的配置·····	274
章前导读·····	228	14.8 反向代理服务器的配置·····	274
本章要点·····	228	参考文献·····	276
13.1 FTP 服务概述·····	228		

第1章 Linux 简介

章前导读

Linux 是一套免费使用和自由传播的类 UNIX 操作系统。Linux 可安装在各种计算机硬件设备中，比如个人计算机、服务器、路由器、防火墙、小型机、大型机等。对我们个人用户而言，最熟悉的 Linux 应用莫过于今天广泛使用的 Android 手机以及平板电脑等手持终端，Android 正是一款基于 Linux 的开源操作系统。

Linux 到底有多强大呢？

早在 1997 年上映的好莱坞大片《泰坦尼克号》，后期的特效制作所使用的 160 台 Alpha 图形工作站中，就有 105 台采用了 Linux 操作系统。除此之外，后来的《哈利波特》《指环王》《星球大战》等诸多电影的制作过程中都有 Linux 的身影；今天 Google 两万多台服务器组成的集群是由 Linux 支撑的；我们每天都会访问的大型网站如新浪、百度、淘宝等，都在使用 Linux 的服务器；世界上速度最快的计算机运行的就是 Linux/UNIX；全球 95% 的高端服务器运行的也是 Linux/UNIX 操作系统……可见 Linux 对世界、对我们每个人生活的影响是非常广泛的。

本章要点

- ✎ Linux 的发展及特点
- ✎ Linux 的文件结构
- ✎ 了解常见的 Linux 发行版本

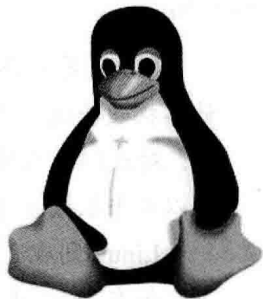
1.1 Linux 发展史

Linux 操作系统是一个类 UNIX 的操作系统。1991 年，芬兰赫尔辛基大学的学生 Linus Torvalds，受 MINIX 系统的启发，推出一个新的 UNIX 的变种，并在新闻组 comp.os.MINIX

发布了大约有一万行代码的最早的 Linux 内核版本 v0.01。1991 年 10 月 5 日 Linus Torvalds 正式向外界宣布 Linux 内核系统诞生,并发布了 v0.02 版内核。从此,10 月 5 日对 Linux 社区来说成为了一个特殊的日子,后来许多 Linux 新版本发布时都选择了这个日子,而 Linus Torvalds 也被称为 Linux 之父。借助于 Internet 网络,经过全世界各地计算机爱好者的共同努力, Linux 现已成为今天世界上最流行的操作系统之一,并且使用人数还在迅猛增长。

“Linux”的读音五花八门版本颇多,根据 Linux 的创始人 Linus Torvalds 的说法, Linux 的发音和“MINIX”是押韵的。“Li”中“i”的发音类似于“MINIX”中“i”的发音,而“nux”中“u”的发音类似于英文单词“profess”中“o”的发音。依照国际音标应该是/linəks/ (类似于“里讷克斯”)。但是,由于 Linus Torvalds 本人是芬兰人,根据当地语言的发音, Linux 读音/liniks/更为贴切。

Linux 的标志和吉祥物是一只名字叫做 Tux 的企鹅,标志的由来是因为 Linus 在澳洲时曾被动物园里的一只企鹅咬了一口,便选择了企鹅作为 Linux 的标志。Linux 的注册商标是 Linus Torvalds 所有的。Linus Torvalds 一再声明 Linux 是免费的,他本人可以卖掉,但 Linux 绝不能卖。现在我们使用的很多 Linux 发行版本是不需要安装的,只需通过 CD 或者可启动的 USB 存储设备就能使用,这种版本称为 LiveCD。



Linux 操作系统的诞生、发展和成长过程始终依赖着以下 5 个重要支柱: UNIX 操作系统、MINIX 操作系统、GNU 计划、POSIX 标准和 Internet 网络。依照这 5 个基本线索我们可简单探究 Linux 最初的发展过程。

1. UNIX 操作系统

Linux 操作系统是 UNIX 操作系统的一个克隆版本。UNIX 操作系统是美国贝尔实验室的 Ken.Thompson 和 Dennis Ritchie 于 1969 年夏在 DEC PDP-7 小型计算机上开发的一个分时操作系统。当时使用的是 BCPL 语言(基本组合编程语言),后经 Dennis Ritchie 于 1972 年用移植性很强的 C 语言进行了改写,使得 UNIX 系统在大专院校得到了推广。

2. MINIX 操作系统

MINIX 系统是由 Andrew S. Tanenbaum (AST) 开发的。AST 在荷兰阿姆斯特丹的自由大学从事数学与计算机科学系统工作,是 ACM 和 IEEE 的资深会员(全世界只有为数不多的人是两会的资深会员)。MINIX 是他在 1987 年编制的,主要用于学生学习操作系统原理。目前主要有两个版本在使用: 1.5 版和 2.0 版,目前都已经是免费的,可以从许多 FTP 上下载。

作为一个操作系统，MINIX 并不是优秀者，但它提供了用 C 语言和汇编语言编写的系统源代码。这是第一次使得有抱负的程序员或 hacker 能够阅读操作系统的源代码，在当时这种源代码是软件商一直小心地守护着的。

对于 Linux 系统，AST 表示对其开发者 Linus 的称赞。但他认为 Linux 的发展有很大原因是由于他为了保持 MINIX 的小型化，能让学生在一个学期内就能学完，而没有接纳全世界许多人对 MINIX 的扩展要求。这激发了 Linus 编写 Linux，Linus 正好抓住了这个好时机。

3. GNU 计划

GNU 计划和自由软件基金会 (the Free Software Foundation, FSF) 是由 Richard M. Stallman 于 1984 年一手创办的，旨在开发一个类似 UNIX 并且是自由软件的完整操作系统：GNU 系统 (GNU 是 “GNU’s Not UNIX” 的递归缩写)。

到 20 世纪 90 年代初，GNU 项目已经开发出许多高质量的免费软件，其中包括有名的 emacs 编辑系统、bash shell 程序、gcc 系列编译程序、gdb 调试程序等。这些软件为 Linux 操作系统的开发创造了一个合适的环境，是 Linux 能够诞生的基础之一。

各种使用 Linux 作为核心的 GNU 操作系统正在被广泛使用，虽然这些系统通常被称作 “Linux”，但是严格地说，我们通常说的 Linux 仅仅是指内核部分，并不能代表 Linux 的全部，而只有内核的操作系统是没有办法使用的，于是人们将 Linux 内核与 GNU 项目开发的各种应用程序结合在一起，就形成了一个完整的操作系统，即基于 Linux 内核的 GNU 系统，所以 Linux 操作系统的完整名称应为 GNU/Linux 系统。

4. POSIX 标准

POSIX (Portable Operating System Interface for Computing Systems) 是由 IEEE 和 ISO/IEC 开发的一簇标准。该标准是基于现有的 UNIX 实践和经验，描述了操作系统的调用服务接口，用于保证编制的应用程序可以在源代码一级上在多种操作系统上移植运行。它是在 1980 年一个 UNIX 用户组 (usr/group) 的早期工作的基础上取得的。该 UNIX 用户组原来试图将 AT&T 的系统 V 和 Berkeley CSRG 的 BSD 系统的调用接口之间的区别重新调和集成，从而于 1984 年产生了 /usr/group 标准。1985 年，IEEE 操作系统技术委员会标准小组委员会 (TCOS-SS) 开始在 ANSI 的支持下责成 IEEE 标准委员会制定有关程序源代码可移植性操作系统服务接口正式标准。到了 1986 年 4 月，IEEE 就制定出了试用标准。第一个正式标准是在 1988 年 9 月份批准的 (IEEE 1003.1-1988)，也就是后来经常提到的 POSIX.1 标准。

1989 年 POSIX 的工作被转移至 ISO/IEC 社团，并由 15 个工作组继续将其制定成 ISO 标准。到 1990 年，POSIX.1 与已经通过的 C 语言标准联合，正式批准为 IEEE 1003.1-1990

(也是 ANSI 标准)和 ISO/IEC 9945-1:1990 标准。POSIX.1 仅规定了系统服务应用程序编程接口(API),仅概括了基本的系统服务标准,因此期望对系统的其他功能也制定出标准。这样 IEEE POSIX 的工作就开始展开了。在 1990 年,刚开始有 10 个批准的计划在进行,有 300 多人参加每季度为期一周的会议。着手的工作有命令与工具标准(POSIX.2)、测试方法标准(POSIX.3)、实时 API(POSIX.4)等。到了 1990 年上半年已经有 25 个计划在进行,并且有 16 个工作组参与其中。与此同时,还有一些组织也在制定类似的标准,如 X/Open, AT&T, OSF 等。

在 90 年代初,POSIX 标准的制定正处在最后投票敲定的时候,那是 1991—1993 年间。此时正是 Linux 刚刚起步的时候,这个 UNIX 标准为 Linux 提供了极为重要的信息,使得 Linux 能够在标准的指导下进行开发,能够与绝大多数 UNIX 系统兼容。在最初的 Linux 内核代码中(0.01 版、0.11 版)就已经为 Linux 与 POSIX 标准的兼容做好了准备工作。

1991 年 7 月 3 日, Linus 在 comp.os.MINIX 上发布的 post 上就已经提到了正在搜集 POSIX 的资料。其中透露了他正在进行 Linux 系统的开发,并且在 Linux 最初的时候已经想到要实现与 POSIX(UNIX 的国际标准)的兼容问题了。

5. Internet

Linux 从诞生之日起就与 Internet 密不可分,支持各种标准的 Internet 网络协议。目前, Linux 几乎支持所有主流的网络硬件、网络协议和文件系统。

由于 Linux 是一款免费的开源操作系统,全世界的用户都可以通过 Internet 或其他途径免费获得,并可以任意修改其源代码,这是其他的操作系统做不到的。正是由于这一点,来自全世界的无数 Linux 爱好者和程序员参与了 Linux 的修改、编写工作,每个人都可以根据自己的兴趣和灵感对其进行改变,这让 Linux 吸取了无数程序员的精华,不断壮大。

正所谓众人拾柴火焰高,通过 Internet 将全世界优秀程序员的智慧都汇集到了一起,这才使得 Linux 成为了今天无论是在安全性亦或是可靠性都更胜一筹的优秀操作系统。

1.2 Linux 的特点

1. 开源、免费

由于 Linux 系统的开发从一开始就与 GNU 项目紧密地结合起来,所以它的大多数组成部分都直接来自 GNU 项目。任何人、任何组织只要遵守 GPL 条款,就可以自由使用 Linux 源代码,为用户提供了最大限度的自由度。这一点也正投嵌入式系统所好,因为嵌入式系统应用千差万别,设计者往往需要针对具体的应用对源码进行修改和优化,所以是

否能获得源代码对于嵌入式系统的开发是至关重要的。

加之 Linux 的软件资源十分丰富，每种通用程序在 Linux 上几乎都可以找到，并且数量还在不断增加。这一切使设计者在其基础之上进行二次开发变得非常容易。另外，由于 Linux 源代码公开，也使用户不用担心有诸如“后门”之类的安全隐患。

同时，源码开放给众多 Linux 的学习和爱好者提供极大的方便，很大程度上促进了 Linux 的学习、推广和应用。

2. 完全兼容 POSIX1.0 标准

对 POSIX1.0 标准的良好兼容，使得用户可以在 Linux 下通过相应的模拟器运行常见的 DOS、Windows 的程序，这为用户从 Windows 转到 Linux 奠定了基础。许多用户在考虑使用 Linux 时，就想到以前在 Windows 下常见的程序是否能正常运行，这一点足以消除他们对此的众多疑虑。

3. 模块化

Linux 的内核设计非常精巧，分成进程调度、内存管理、进程间通信、虚拟文件系统和网络接口五大部分。其独特的模块机制可根据用户的需要，实时地将某些模块插入或从内核中移走，使得 Linux 系统内核可以裁剪得非常小巧，适合于嵌入式系统的需要。

4. 多用户、多任务

Linux 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 可以使多个程序同时并独立地运行。

5. 安全性及可靠性好

Linux 中大量网络管理、网络服务等方面的功能，可使用户很方便地建立高效稳定的防火墙、路由器、工作站、服务器等。为提高安全性，它还提供了大量的网络管理软件、网络分析软件和网络安全软件等。Linux 内核的高效和稳定已在各个领域内得到了大量事实的验证。

6. 良好的界面

Linux 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。它同时也提供了类似 Windows 图形界面的 X-Window 系统，用户可以使用鼠标对其进行操作。在 X-Window 环境中就和在 Windows 中相似，可以说是一个 Linux

版的 Windows。

7. 支持多种平台

Linux 可以运行在多种硬件平台上，如具有 x86、680x0、SPARC、Alpha 等处理器的平台。此外 Linux 还有一种嵌入式操作系统，可以运行在手机、掌上电脑、机顶盒或游戏机上。2001 年 1 月份发布的 Linux 2.4 版内核已经能够完全支持 Intel 64 位芯片架构。同时 Linux 也支持多处理器技术，多个处理器同时工作，使系统性能大大提高。

8. 具有优秀的开发工具

开发嵌入式系统的关键是需要有一套完善的开发和调试工具。传统的嵌入式开发调试工具是在线仿真器（In Circuit Emulator, ICE），它通过取代目标板的微处理器，给目标程序提供一个完整的仿真环境，从而使开发者能非常清楚地了解到程序在目标板上的工作状态，便于监视和调试程序。但是在线仿真器的价格非常高，而且只适合做非常底层的调试。

如果使用的是嵌入式 Linux，一旦软硬件能支持正常的串口功能，即使不用在线仿真器，也可以很好地进行开发和调试工作，从而节省了一笔不小的开发费用。嵌入式 Linux 为开发者提供了一套完整的工具链，能够很方便地实现从操作系统到应用软件各个级别的调试。

1.3 Linux 版本编号方式

Linux 自诞生至今，其内核共有 3 种不同的版本编号方式。

第一种方式用于 1.0 版本之前（包括 1.0）。第一个版本是 0.01，紧接着是 0.02、0.03、0.10、0.11、0.12、0.95、0.96、0.97、0.98、0.99 和之后的 1.0。

第二种方式用于 1.0 之后到 2.6，由 3 部分数字“A.B.C”组成，其中 A 代表主版本号，B 代表副版本号，C 代表较小的末版本号。

只有在内核发生很大变化时（历史上只发生过两次，1994 年的 1.0 和 1996 年的 2.0），A 才变化。通过数字 B 可以判断 Linux 是否稳定，偶数的 B 代表稳定版，奇数的 B 代表开发版。C 代表一些 bug 修复、安全更新、新特性和驱动的次数。

以版本 2.4.0 为例，2 代表主版本号，4 代表次版本号，0 代表改动较小的末版本号。在版本号中，序号的第二位为偶数的版本表明这是一个可以使用的稳定版本，如 2.2.5，而序号的第二位为奇数的版本一般有一些新的东西加入，是不一定很稳定的测试版本，如 2.3.1。稳定版本来源于上一个测试版升级版本号，而一个稳定版本发展到完全成熟后就不再发展。

第三种方式从 2004 年 2.6.0 版本开始，使用一种“time-based”的方式。3.0 版本之前，

是一种“A.B.C.D”的格式。7年里，前两个数字 A.B 即“2.6”保持不变，C 随着新版本的发布而增加，D 代表一些 bug 修复、安全更新、添加新特性和驱动的次数。3.0 版本之后是“A.B.C”格式，B 随着新版本的发布而增加，C 代表一些 bug 修复、安全更新、新特性和驱动的次数。

第三种方式中不再使用偶数代表稳定版，奇数代表开发版这样的命名方式，举个例子：3.7.0 代表的不是开发版，而是稳定版。

1.4 Linux 文件结构

Linux 文件系统采用树型目录结构来组织管理文件，所有文件采取分级分层的方式组织在一起，从而形成一个树型的层次结构。

在 Linux 操作系统的整个树型目录结构中，只有一个根目录（用“/”表示），位于根分区，文件和目录都是建立在根目录之下的。若要查看当前系统的目录结构，可在文本模式或终端窗口的命令提示符下使用 ls 命令来进行查看，命令用法及结果如图 1-1 所示。

```
[root@localhost ~]# ls /
bin  dev  home  lost+found  misc  net  proc  sbin  srv  tftpboot  usr
boot  etc  lib  media  mnt  opt  root  selinux  sys  tmp  var
```

图 1-1 查看当前系统的目录结构

本教程例举的 Red Hat Enterprise Linux 采用标准的 Linux 目录结构，从根目录（/）开始的每个目录都有其特定用途，用于存放某些特定类型的文件。以下对根目录及其下主要目录的功能作简要说明：

/：根目录，所有的目录、文件、设备都在根目录（/）之下，/就是 Linux 文件系统的组织者，也是最上级的领导者。

/bin：bin 是二进制（binary）英文缩写。在一般的系统当中，都可以在这个目录下找到 Linux 常用的命令。系统所需要的那些命令位于此目录。

/boot：Linux 的内核及引导系统程序所需要的文件目录，比如 vmlinuz initrd.img 文件都位于这个目录中。在一般情况下，GRUB 或 LILO 系统引导管理器也位于这个目录下。

/dev：dev 是设备（device）的英文缩写。这个目录对所有的用户都十分重要。因为在这个目录中包含了所有 Linux 系统中使用的外部设备。但是这里并不是存放外部设备驱动程序的。这一点和常用的 Windows、DOS 操作系统不一样，它实际上是一个访问这些外部设备的端口。通过该端口，可以非常方便地去访问这些外部设备，和访问一个文件、一个目录没有任何区别。

/etc：etc 这个目录是 Linux 系统中最重要目录之一。在这个目录下存放了系统管理

时要用到的各种配置文件和子目录。例如，网络配置文件、文件系统、x 系统配置文件、设备配置信息、用户设置信息等都在这个目录下。

/home: 建立普通用户的主目录。如果建立一个用户，用户名是“xx”，那么在/home 目录下就有一个对应的/home/xx 路径，用来存放用户的主目录。

/lib: lib 是库（library）的英文缩写。这个目录是用来存放系统动态链接共享库的。几乎所有应用程序都会用到这个目录下的共享库。因此，千万不要轻易对这个目录进行什么操作，一旦发生问题，系统就不能工作了。

/lost+found: 在 ext2 或 ext3 文件系统中，系统意外崩溃或机器意外关机时，其而产生一些文件碎片放在这里。在系统启动的过程中 fsck 工具会检查这里，并修复已经损坏的文件系统。有时系统发生问题，有很多的文件被移到这个目录中，可能会用手工的方式来修复，或转移文件到原来的位置上。

/mnt: 这个目录一般是用于存放挂载储存设备的挂载目录的，比如 cdrom 等目录。

/media: 有些 Linux 的发行版本使用这个目录来挂载那些 USB 接口的移动硬盘（包括 U 盘）、CD/DVD 驱动器等。

/opt: 这里主要存放那些可选的程序。

/proc: 可以在这个目录下获取系统信息。这些信息是在内存中，由系统自己产生的。

/root: Linux 超级权限用户 root 的主目录。

/sbin: 这个目录用来存放系统管理员的系统管理程序，大多是涉及系统管理命令的存放，是超级权限用户 root 的可执行命令存放地。普通用户无权限执行这个目录下的命令，这个目录和/usr/sbin、/usr/X11R6/sbin 或/usr/local/sbin 目录相似，凡是/sbin 目录中包含的都是 root 权限才能执行的。

/seLinux: 对 SELinux 的一些配置文件目录，SELinux 可以让 Linux 更加安全。

/srv: 服务启动后，所需访问的数据目录，举个例子来说，www 服务启动读取的网页数据就可以放在/srv/www 中。

/tmp: 临时文件目录，用来存放不同程序执行时产生的临时文件。/var/tmp 目录和这个目录相似。

/usr: 这是 Linux 系统中占用硬盘空间最大的目录。用户的很多应用程序和文件都存放在这个目录下。在这个目录下，可以找到那些不适合放在/bin 或/etc 目录下的额外工具。

/usr/local: 这里主要存放那些手动安装的软件。它和/usr 目录具有相类似的目录结构。让软件包管理器来管理/usr 目录，而将自定义的脚本（scripts）放到/usr/local 目录下面。

/usr/share: 系统共用的文件存放地，比如/usr/share/fonts 是字体目录，/usr/share/doc 和/usr/share/man 是帮助文件目录。

/var: 这个目录的内容是经常变动的，看名字就知道，可以理解为 vary 的缩写，/var 下有/var/log，这是存放系统日志的目录。/var/www 是定义 Apache 服务器站点存放目录；/var/lib

用来存放一些库文件，比如 MySQL 数据库。

1.5 常见的 Linux 发行版本

Linux 有两种版本：核心（Kernel）版本和发行（Distribution）版本。核心版本仅仅是指内核的版本，它没有用户界面、应用程序等外围资源，所以单纯的内核并不是一个完整的操作系统。发行版本是每个社区或公司增加了诸多外围资源，比如用户界面、系统安装、应用程序等。所以各发行版的内核可以一致，但发行版采取的各种用户应用方式或资源往往是不同的。

用户可以使用命令 `uname -a` 或者 `uname -r` 查看 Linux 内核的版本号。目前已经有超过 300 个发行版被积极地开发，下面对其中最为著名并被广泛使用的发行版做简单介绍。

1. Fedora Project

Fedora Linux（第七版以前为 Fedora Core）是众多 Linux 发行版之一，它是一套从 Red Hat Linux 发展出来的免费 Linux 系统，可运行的体系结构包括 x86（即 i386-i686），x86_64 和 PowerPC。Fedora 由 Fedora Project 社群开发，这个社区的成员以自己的不懈努力，提供并维护自由、开放源码的软件和开放的标准。Fedora 项目由 Fedora 基金会管理和控制，得到了 Red Hat 的大力支持。它是一个开放、创新和具有前瞻性的 Linux 操作系统和平台，允许任何人自由地使用、修改和重发布，无论现在还是将来。

Fedora Project 主页：<http://fedoraproject.org/>

2. Debian

Debian Project 诞生于 1993 年 8 月 13 日，它的目标是提供一个稳定容错的 Linux 版本。支持 Debian 的不是某家公司，而是许多在其改进过程中投入了大量时间的开发人员，这种改进吸取了早期 Linux 的经验。Debian 以其稳定性著称，虽然它的早期版本 Slink 有一些问题，但是它的现有版本 Potato 已经相当稳定了。

Debian 的安装完全是基于文本的，对于其本身来说这不是一件坏事，但对于初级用户来说却并非这样。因为它仅仅使用 `fdisk` 作为分区工具而没有自动分区功能，所以它的磁盘分区过程令人十分讨厌。磁盘设置完毕后，软件工具包的选择通过一个名为 `dselect` 的工具实现，但它不向用户提供安装基本工具组（如开发工具）的简易设置步骤。最后需要使用 `anXious` 工具配置 X Windows，这个过程与其他版本的 X Windows 配置过程类似。完成这些配置后，Debian 就可以使用了。

Debian 主要通过基于 Web 的论坛和邮件列表来提供技术支持。作为服务器平台，Debian 提供一个稳定的环境。为了保证它的稳定性，开发者不会在其中随意添加新技术，