

Linux系统命令 及Shell脚本 实践指南

王军 著

Linux Commands and Shell Scripting

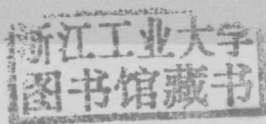
- 采用理论联系实际的方式，从系统管理出发，深入剖析Linux系统的运行原理，介绍Linux系统中的常用命令，理清Bash Shell编程的脉络。
- 结合作者多年的运维诊断经验，提供了大量实用性极强的脚本案例，对于广大Linux系统运维人员来说，可谓“一书在手，运维不愁”。



Linux系统命令 及Shell脚本 实践指南

Linux Commands and Shell Scripting

王军 著



浙江工业大学图书馆



72014574



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

Linux 系统命令及 Shell 脚本实践指南 / 王军著. —北京: 机械工业出版社, 2013.11

ISBN 978-7-111-44503-6

I. L… II. 王… III. ① Linux 操作系统 ② UNIX 操作系统 - 程序设计 IV. TP316.8

中国版本图书馆 CIP 数据核字 (2013) 第 251113 号

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问 北京市展达律师事务所

本书由浅入深、全面、系统地介绍了 Linux 系统的基础命令和 Shell 脚本的开发。作为一本面向 Linux 入门用户的书籍, 本书力求语言通俗易懂, 用例简单明了, 便于读者阅读领会。同时, 全书穿插了不少要求读者动手实践的内容, 不但可加深读者对内容的理解和消化, 同时也可通过动手实践激发读者对 Linux 系统和 Shell 编程的兴趣, 从而更好地驱动读者学习。

全书共计 18 章, 从知识结构上分为三部分。第一部分为前 8 章, 全面介绍了 Linux 的基础知识和常用命令; 第二部分为第 9 章和第 10 章, 主要是 Linux 下交互和非交互编辑器的使用; 第三部分为第 11 ~ 18 章, 主要是 Shell 脚本开发。这三部分内容各有偏重, 读者可以根据实际需要选择阅读。

本书根据作者多年来 Linux 环境运维管理工作的经验编著而成, 内容丰富, 且注重实践讲解, 非常适合 Linux 初学者和基于 Linux 平台的软件开发者学习或查阅。从内容结构上来看, 本书摒弃了各章独立的写作风格, 而是采取了各章有机结合的方式, 自上而下自成一体, 既加深了读者对单个知识点的理解, 也有利于从全局把控 Linux 系统和 Shell 编程的精髓。

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 余洁

三河市杨庄长鸣印刷装订厂印刷

2014 年 1 月第 1 版第 1 次印刷

186mm × 240mm · 19.5 印张

标准书号: ISBN 978-7-111-44503-6

定 价: 59.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzsj@hzbook.com

推荐序 1

认识王军是在 2008 年，还记得当年他说是怀着对 Linux 及开源软件极大的兴趣来参加我的课程。作为一个 Linux 从业 10 年并在开源软件培训业从事教学多年的我而言，说“阅人无数”毫不夸张，也因此对很多学生都没有太深刻的印象。随着课程的开展和对王军逐渐的了解，我渐渐地感受到他好学的精神和他对技术的执着，也逐渐对其心生敬佩，并坚定地认为，只需假以时日他一定能在 Linux 领域有所成绩。

课程结束后至今，我和他一直保持着断断续续的联系，也经常能从其他朋友的口中听到他的消息，事实证明他的努力已经逐渐得到越来越多的认可。在此期间我也在不少技术论坛上看到他的技术帖，其中很多文字表现出他对技术独到的见解。

一个偶然的再次见到王军，并得知他正在写一本关于 Linux 系统管理和 Shell 编程方面的书，当时就希望能在初步成稿后一览为快。拿到初稿后，我花了几个小时仔细通读了一遍，可以肯定这确实是一本凝结了他多年实际工作经验的作品。与很多 Linux 书籍不同，该书尽量避免使用生硬的技术词汇，对知识点的描述都尽量地做到“去技术化”、“去概念化”，同时也能看出各章节前后的衔接顺序包括各个知识点的出场他都做了精心设计，并深入浅出地加以描述，这些都是我认为该书是一本很好的 Linux 系统和 Shell 编程入门书籍的原因，初学者完全可以通过该书迅速入门，但是要想完全吃透该书还需要读者结合实际工作多次研读。

最后希望王军的书能帮助和带领更多的 Linux 系统爱好者进入 Linux 的殿堂，同时也希望王军能再接再厉，再出好书！

上海尚观科技有限公司教学总监、技术总监、首席系统架构师 邹凯 (Kissingwolf)

推荐序 2

2010 年有幸与王军共事，期间相处极为投缘。王军当时在公司主要负责推动平台虚拟化和自动化运维相关工作，在这段时间里王军给我的印象是为人热情，凡事有求必应；另一方面他非常热衷于技术研究，喜欢钻研新技术，喜欢突破创新，在那段时间里王军带着其他同事将公司原来的虚拟化平台进行了一次大规模升级，使公司系统更稳定，遗憾的是后来有了更好的发展机会他离开了 51JOB。

得知王军的书稿已经完成并不日出版，为他高兴的同时也深感不易：他平时很忙且需要经常出差，能在日常繁忙工作的基础上，把一些知识要点记录下来已经是很少有人能做到的事情了，而且还能坚持整理成册，这绝对是需要毅力才能完成的工作。

借此机会分享一次与王军的聊天心得，这也是我们 Linux 服务器运维工程师心灵成长的点滴记录。运维工作已经不是搬搬服务器、扛扛交换机、配配网络的时代了，现在运维工作应该以“降低成本，提升用户体验”为目标。降低成本就无形要求运维技能的提升，如现在流行的“去 IOE”，去掉高端、昂贵小型机服务器就必须用多台廉价的 PC 服务器代替，但又要保证系统稳定、高可用、可扩展性强，这样就要求运维工程师具备过硬的 Linux 技能。在提升用户体验的过程中，有三点极其重要：一是稳定，不能频繁宕机；二是要快，天下武功，唯快不破；三是界面友好，不能半天找不到操作按钮——这些都是与精湛的技术密不可分的。

收到王军的定稿并邀请写序，突然感觉到压力，一是自己很少写序，怕写不好；二是担心影响了王军的努力成果。在仔细阅读后，感觉本书最大特点是结构清晰、各章节前后贯穿、知识框架完整且覆盖全面，伴有大量深入浅出的案例，无疑是一本很好的 Linux 系统运维和 Shell 编程的作品，我很钦佩作者的技术功底。

本书从基础知识入手，系统讲解了 Linux 系统结构、主流服务器搭建及故障排除、用户权限管理、磁盘存储管理、文件系统管理、内存管理、进程管理、Shell 编程等关键技术，同时，王军根据他多年的运维诊断经验，提供了大量实用性极强的脚本案例，对于广大 Linux 服务器运维人员来说，可谓“一书在手，运维不愁”。

前 言

为什么要写这本书

早在我还在大学校园时就对 Linux 产生了极大的兴趣，期间我热衷于下载、安装、体验各种不同的 Linux 发行版，并尝试使用 Linux 作为我的桌面系统。但实际情况是，由于大学中使用群体极小，学校又没有开设直接的 Linux 系统课程，虽然我花了不少的课余时间去研究它，但始终感觉不得其法，难以入门。至今我依然记得当时使用鼠标双击好不容易才复制桌面上的 rpm 包，并抱怨为什么没有出现类似于 Windows 的“安装向导”。所以实际上有很长一段时间，面对 Linux 系统我能做的少之又少。

2006 年大学毕业后，我有幸进入了一直梦寐以求的 IT 行业，并从此正式走上了技术之路。工作中能实际接触到 Linux 系统运维是我在该领域发展的很重要的外部因素，工作的驱使和个人的兴趣成为我每天坚持学习 Linux 的源动力。但当时很尴尬的一个现实是：一方面互联网行业的高速发展促进了 Linux 如火如荼的发展，另一方面又很难找到真正适合“新手”的入门级教材，得到一本简单明了的入门书籍是我当时迫切的愿望。于是在走了不少弯路并感觉自己已经“迷路”之后，我报名参加了 Linux 系统工程师社会培训班，利用工作之余系统并完整地学习了 Linux。事实证明，当时的选择是正确的，这直接影响了我至今的职业发展乃至今后的职业规划。

经历了多年的工作后，我也非常希望能有机会与大家分享自己在 IT 领域的体会，所以也经常在一些技术网站发表技术文章，或是与志同道合的朋友一起举办免费的网络培训班。但是总体来说，所涉及的内容大多零碎、不成体系。筹划本书的初期，我想把重点放在 Linux 系统管理、高性能计算、高可用集群甚至云计算这些“够时髦”的主题上，但是反复思考后觉得，“时髦”的技术永远在变，而且限制了读者范围。但是对我、对很多梦想学习 Linux 的读者来说这更是一个机会：用最简单、最朴素、最基础的语言讲解和描述 Linux 系统以及如何使用它，给更多初学者以“可以学会”的希望和“努力前行”的力量。

出于这样的考虑，我在组织本书的内容时尽量安排书的各个章节以及每章节中的每个小节，甚至是每小节中的知识点的出现顺序符合新手的认知规律，做到从易到难，从基础到提高，以循

序渐进的方式将各类知识点以人物出场、层次推进的方式呈现在读者面前，尽量避免将生僻的术语突然摆在读者面前，造成读者思维上的困扰，并且尽量使用简单明了的文字和浅显易懂的比喻帮助读者理解、消化。尽管如此，我还是希望读者能在此基础上展开阅读，并根据实际需要做必要的深入理解。

总之，这是一本讲解 Linux 系统和 Shell 编程的入门级书籍，内容主要涉及 Linux 的基础命令、编辑器的使用和 Shell 脚本的开发。

读者对象

本书适合以下读者阅读：

- Linux 爱好者
- Linux 初学者
- 希望学习 Shell 编程的读者
- 希望了解系统的网络工程师
- 网站前后台开发人员

如何阅读本书

本书从知识结构上分为三大部分，第一部分（第 1 ~ 8 章）为基础内容，详细地介绍了 Linux 的历史发展、安装使用、用户管理、文件管理、文件系统、字符处理、网络管理、进程管理和软件安装。第二部分（第 9 ~ 10 章）为编辑器部分，内容为 Linux 下常用编辑器 vi 和 vim 的用法和基于流处理的 sed 和 awk 工具，这是管理 Linux 系统的基本技能。第三部分（第 11 ~ 18 章）为 Shell 编程，内容包括 Shell 的安装、使用、语法，其中最后一章是本部分的重点，该章所有脚本在实际应用中有很高的使用率。

如果你是 Linux 的初学者，我建议从第 1 章开始阅读。第二部分的内容涉及 vi 和 vim 编辑器的操作细节，建议读者通读。如果读者已经有一定的基础，希望学习 Shell 脚本开发，可以直接跳至第三部分学习。

勘误和支持

由于作者的水平有限，编写的时间也很仓促，书中难免会出现一些错误或者不准确的地方，恳请读者批评指正。如果你有更多的宝贵意见，欢迎你发送邮件至我的邮箱 johnwang.wangjun@gmail.com，或是关注我的新浪微博 weibo.com/u/1186347743，我很期待能够听到你们的真挚反馈。

致谢

首先，感谢伟大的 Linux 之父 Linus Torvalds，他最初开发的这套 Linux 系统已经改变了整个世界的面貌，也影响了我个人的职业乃至人生发展。

感谢那些无数个为了解决问题而彻夜无眠的夜晚，感谢那些在寒冬凌晨的三四点接到报警电话后集体上阵的兄弟姐妹们，感谢那些年一起扛起 24×7 运维任务的战友们，这一切无疑是我人生中最宝贵的财富。

感谢机械工业出版社华章公司的编辑杨绣国 (Lisa)，感谢她在这一年的时间里始终支持我的写作，她的鼓励和帮助引导我顺利完成全部书稿。

谨以此书，献给我最亲爱的家人，以及众多热爱 Linux 的朋友们。

王军

2013 年 10 月于上海

目 录

推荐序 1

推荐序 2

前 言

第 1 章 Linux 简介 1

1.1 Linux 的发展历史 1

1.2 Linux 的特点 2

1.3 系统安装 3

1.3.1 安装前的规划 3

1.3.2 安装 RedHat 4

1.3.3 安装 CentOS 11

1.4 系统登录 16

1.4.1 第一次登录系统的设置 16

1.4.2 使用图形模式登录 21

1.4.3 使用终端模式登录 23

1.4.4 开始学习使用 Linux 的

命令 24

1.5 系统启动流程 25

1.5.1 系统引导概述 25

1.5.2 系统运行级别 26

1.5.3 服务启动脚本 27

1.5.4 Grub 介绍 29

1.6 获得帮助 31

1.6.1 使用 man page 31

1.6.2 使用 info page 32

1.6.3 其他获得帮助的方式 32

第 2 章 Linux 用户管理 33

2.1 Linux 用户和用户组 33

2.1.1 UID 和 GID 33

2.1.2 /etc/passwd 和 /etc/shadow 34

2.2 Linux 账号管理 36

2.2.1 新增和删除用户 36

2.2.2 新增和删除用户组 39

2.2.3 检查用户信息 39

2.3 切换用户 40

2.3.1 切换成其他用户 40

2.3.2 用其他用户的身份执行

命令: sudo 41

2.4 例行任务管理 43

2.4.1 单一时刻执行一次任务: at 43

2.4.2 周期性执行任务: cron 44

2.4.3 /etc/crontab 的管理 45

第 3 章 Linux 文件管理 46

3.1 文件和目录管理 46

3.1.1 绝对路径和相对路径 47

3.1.2	文件的相关操作	48	4.1.3	ext3 文件系统简介	66
3.1.3	目录的相关操作	51	4.2	磁盘分区、创建文件系统、挂载	66
3.1.4	文件时间戳	53	4.2.1	创建文件系统: fdisk	66
3.2	文件和目录的权限	54	4.2.2	磁盘挂载: mount	69
3.2.1	查看文件或目录的权限:		4.2.3	设置启动自动挂载:	
	ls -al	54		/etc/fstab	70
3.2.2	文件隐藏属性	55	4.2.4	磁盘检验: fsck、badblocks	70
3.2.3	改变文件权限: chmod	56	4.3	Linux 逻辑卷	71
3.2.4	改变文件的拥有者:		4.3.1	什么是逻辑卷	71
	chown	56	4.3.2	如何制作逻辑卷	72
3.2.5	改变文件的拥有组:		4.4	硬链接和软链接	76
	chgrp	57	4.4.1	什么是硬链接	76
3.2.6	文件特殊属性:		4.4.2	什么是软链接	77
	SUID/SGID/Sticky	57	第 5 章 字符处理		79
3.2.7	默认权限和 umask	58	5.1	管道	79
3.2.8	查看文件类型: file	60	5.2	使用 grep 搜索文本	79
3.3	查找文件	60	5.3	使用 sort 排序	80
3.3.1	一般查找: find	61	5.4	使用 uniq 删除重复内容	82
3.3.2	数据库查找: locate	61	5.5	使用 cut 截取文本	82
3.3.3	查找执行文件: which/ whereis	62	5.6	使用 tr 做文本转换	83
3.4	文件压缩和打包	63	5.7	使用 paste 做文本合并	84
3.4.1	gzip/gunzip	63	5.8	使用 split 分割大文件	84
3.4.2	tar	63	第 6 章 网络管理		86
3.4.3	bzip2	63	6.1	网络接口配置	86
3.4.4	cpio	64	6.1.1	使用 ifconfig 检查和 配置网卡	86
第 4 章 Linux 文件系统		65	6.1.2	将 IP 配置信息写入配置 文件	87
4.1	文件系统	65	6.2	路由和网关设置	88
4.1.1	什么是文件系统	65			
4.1.2	ext2 文件系统简介	65			

6.3	DNS 客户端配置	89	8.3.2	使用 yum 安装 Apache	115
6.3.1	/etc/hosts	89	8.3.3	RedHat 使用 yum 的问题	115
6.3.2	/etc/resolv.conf	90	8.3.4	自建本地 yum 源	118
6.4	网络测试工具	90	8.3.5	自建网络 yum 源	121
6.4.1	ping	90	8.4	三种安装方法的比较	123
6.4.2	host	91	8.5	重建 RPM 包	124
6.4.3	traceroute	91	8.5.1	创建重建环境	124
6.4.4	常见网络故障排查	92	8.5.2	快速重建 RPM 包	124
			8.5.3	以 spec 文件重建 RPM 包	125
			8.5.4	spec 文件简介	126
第 7 章	进程管理	94	第 9 章	vi 和 vim 编辑器	129
7.1	什么是进程	94	9.1	vi 和 vim 编辑器简介	129
7.2	进程和程序的区别	95	9.2	vi 编辑器	129
7.3	进程的观察: ps、top	95	9.2.1	模式介绍	129
7.4	进程的终止: kill、killall	98	9.2.2	案例练习	130
7.5	查询进程打开的文件: lsof	99	9.3	vim 编辑器	134
7.6	进程优先级调整: nice、renice	101	9.3.1	多行编辑	134
			9.3.2	多文件编辑	134
			9.3.3	使用 vimtutor 深入学习 vim	136
第 8 章	Linux 下的软件安装	103	9.4	gedit 编辑器	137
8.1	源码包编译安装	103	9.4.1	gedit 编辑器简介	137
8.1.1	编译、安装、打印 HelloWorld 程序	103	9.4.2	启动 gedit 编辑器	138
8.1.2	使用源码包编译安装 Apache	106			
8.2	RPM 安装软件	108	第 10 章	正则表达式	139
8.2.1	什么是 RPM	108	10.1	正则表达式基础	139
8.2.2	RPM 包管理命令: rpm	109	10.1.1	什么是正则表达式	139
8.2.3	包依赖关系	110	10.1.2	基础的正则表达式	140
8.2.4	使用 RPM 包安装 gcc	110	10.1.3	扩展的正则表达式	143
8.2.5	使用 RPM 包安装 Apache	112	10.1.4	通配符	144
8.3	yum 安装软件	114			
8.3.1	yum 命令的基本用法	114			

10.2	正则表达式示例	146	11.2.3	Shell 脚本的排错	166
10.3	文本处理工具 sed	150	11.3	Shell 的内建命令	169
10.3.1	sed 介绍	150	第 12 章 Bash Shell 的安装 184		
10.3.2	删除	151	12.1	确定你的 Shell 版本	184
10.3.3	查找替换	152	12.2	安装 bash	184
10.3.4	字符转换	153	12.3	使用新版本的 Bash Shell	186
10.3.5	插入文本	153	12.4	在 Windows 中安装 bash	187
10.3.6	读入文本	154	第 13 章 Shell 编程基础 191		
10.3.7	打印	154	13.1	变量	191
10.3.8	写文件	154	13.1.1	局部变量	191
10.3.9	sed 脚本	155	13.1.2	环境变量	191
10.3.10	高级替换	155	13.1.3	变量命名	197
10.3.11	sed 总结	156	13.1.4	变量赋值和取值	198
10.4	文本处理工具 awk	157	13.1.5	取消变量	199
10.4.1	打印指定域	158	13.1.6	特殊变量	199
10.4.2	指定打印分隔符	158	13.1.7	数组	200
10.4.3	内部变量 NF	159	13.1.8	只读变量	202
10.4.4	打印固定域	159	13.1.9	变量的作用域	203
10.4.5	截取字符串	159	13.2	转义和引用	205
10.4.6	确定字符串的长度	160	13.2.1	转义	205
10.4.7	使用 awk 求列和	160	13.2.2	引用	206
第 11 章 Shell 编程概述 161			13.2.3	命令替换	207
11.1	Shell 简介	161	13.3	运算符	208
11.1.1	Shell 是什么	161	13.3.1	算术运算符	209
11.1.2	Shell 的历史	163	13.3.2	位运算符	209
11.1.3	Shell 的功能	164	13.3.3	自增自减	211
11.1.4	Shell 编程的优势	164	13.4	其他算术运算	211
11.2	第一个 Shell 脚本	164	13.4.1	使用 \$[] 做运算	211
11.2.1	编辑第一个 Shell 脚本	164	13.4.2	使用 expr 做运算	212
11.2.2	运行脚本	165			

13.4.3	内建运算命令 declare	212	15.2	while 循环	234
13.4.4	算术扩展	213	15.2.1	while 循环的语法	234
13.4.5	使用 bc 做运算	213	15.2.2	使用 while 按行读取文件	235
13.5	特殊字符	215	15.2.3	while 的无限循环	236
13.5.1	通配符	215	15.3	until 循环	237
13.5.2	引号	215	15.3.1	until 循环的语法	237
13.5.3	注释符	215	15.3.2	until 的无限循环	238
13.5.4	大括号	216	15.4	select 循环	238
13.5.5	控制字符	217	15.5	嵌套循环	239
13.5.6	杂项	217	15.6	循环控制	241
第 14 章 测试和判断			15.6.1	break 语句	241
14.1	测试	218	15.6.2	continue 语句	241
14.1.1	测试结构	218	第 16 章 函数		
14.1.2	文件测试	219	16.1	函数的基本知识	243
14.1.3	字符串测试	220	16.1.1	函数的定义和调用	243
14.1.4	整数比较	222	16.1.2	函数的返回值	244
14.1.5	逻辑测试符和逻辑运算符	223	16.2	带参数的函数	246
14.2	判断	224	16.2.1	位置参数	246
14.2.1	if 判断结构	224	16.2.2	指定位置参数值	247
14.2.2	if/else 判断结构	225	16.2.3	移动位置参数	248
14.2.3	if/elif/else 判断结构	226	16.3	函数库	249
14.2.4	case 判断结构	227	16.3.1	自定义函数库	249
第 15 章 循环			16.3.2	函数库 /etc/init.d/functions 简介	249
15.1	for 循环	229	16.4	递归函数	251
15.1.1	带列表的 for 循环	229	第 17 章 重定向		
15.1.2	不带列表的 for 循环	231	17.1	重定向简介	255
15.1.3	类 C 的 for 循环	232	17.1.1	重定向的基本概念	255
15.1.4	for 的无限循环	233			

17.1.2	文件标识符和标准输入 输出	255	18.5	文件安全检测脚本	272
17.2	I/O 重定向	255	18.6	ssh 自动登录备份	273
17.2.1	I/O 重定向符号和用法	255	18.7	使用 rsync 备份	275
17.2.2	使用 exec	260	18.8	使用 netcat 备份	277
17.2.3	Here Document	262	18.9	使用 iptables 建立防火墙	279
第 18 章	脚本范例	263	18.10	自定义开机启动项的 init 脚本	282
18.1	批量添加用户脚本	263	18.11	使用脚本操作 MySQL 数据库	286
18.2	检测服务器存活	266	18.12	基于 LVM 快照的 MySQL 数据库备份	288
18.3	使用 expect 实现自动化输入	268	18.13	页面自动化安装 LAMP 环境	291
18.4	自动登录 ftp 备份	270			

第 1 章

Linux 简介

1.1 Linux 的发展历史

首先我们一起来了解一下应该怎么读 Linux 这个单词，根据 Torvalds（Linux 的发明者）在其多次公开场合中的说明，标准的读音应该是“哩呐科斯”，利用搜索引擎加关键字 Linux pronunciation 进行搜索，就可以看到具体的视频。

说到 Linux 就不得不提到 UNIX，因为 Linux 是一种类 UNIX 的系统。早在 1965 年，贝尔实验室加入了一项由美国通用电气公司和麻省理工学院发起的合作计划，该计划要开发一个多用户、多进程、多层次的 Multics 操作系统。由于该计划实际进展太过缓慢，1969 年便暂停了。不过该计划的参与者之一 Ken Thompson 已经从这项计划中获得了一些点子和收获，当时他有一个被称为“星际旅行”的程序在 GE-635 的机器上运行，因为该机器性能问题，运行“星际旅行”太慢，从而引发了他想将这个程序移植到一台性能更好的 DPD-7 上的想法，只是因为家中有小孩需要照顾而一直没有时间动手。巧合的是，在 1969 年 8 月左右，他的妻儿出门探亲了一个月，就在这一个月的时间里，Thompson 编写了一个操作系统，并成功地将“星际旅行”移植到了 DPD-7 上，而这个操作系统就是 UNIX 的原型。

UNIX 由于具有优秀的移植性而得到了广泛的关注和支持，1974 年 12 月伯克利大学获得 UNIX 的源码，并动手将其修改为适合自己机器的版本，最终命名为 BSD，这也是 UNIX 很重要的一个分支。由于当时还没有足够的版权意识，很多商业公司都开始了基于 UNIX 操作系统的开发，比如 AT&T 的 System V、IBM 的 AIX 等，在这段时期中也形成了 UNIX 的两大分支：System V 和 BSD。

后来 AT&T 公司出于商业考虑（贝尔实验室是从属于 AT&T 公司的），1979 年在发行第七版 UNIX 时开始严格限制对学生提供源码。这对大学教学影响非常大，因为在无法看到源码的情况下，教学工作便很难进行。当时有个叫 Tanenbaum 的教授为避免版权纠纷，在完全不看 UNIX 源码的情况下，自己动手写了一个类 UNIX 的系统，并命名为 Minix，这项工作从 1984 年持续到 1986 年。由于开发这个系统的出发点在于教学，所以用户对 Minix 的新需求往往得不到开发支持，只能基于 Minix 的源码自己进行修改。

1984年, Richard Stallman 创立了 GNU 项目, 由自由软件基金支持, GNU 项目的目标是“开发一个完全自由的 UNIX 操作系统”。

“Hello everybody out there using minix, I'm doing a free operation system”, 1991年8月, 网络上出现了以此开篇的帖子, 这是一名芬兰的大学生为了写一个类 Minix 的系统而在寻找志同道合的伙伴, 他就是著名的 Linux 之父——Linus Torvalds。同年10月5日, 他在网络上发布了大约有1万行代码的 Linux 0.01 版本, 次年已经有约1000人在使用 Linux 了。1993年, 大约有100名程序员参与了 Linux 内核开发工作, 其中核心人员有5名, 此时 Linux 0.99 版本的代码大约有10万行, 用户约为10万人。1994年, Linux 加入了 GNU, 成为 GNU 项目中的一员, 同年 Linux 1.0 版本发布, 代码量大约有17万行, 最早按照完全自由免费的协议发布, 用户可以随意下载、使用、修改, 而不需要通知作者。随后采用了 GPL 协议, 很多开发人员开始将自己的代码贡献给核心小组, 这也就使得当时的 Linux 系统对不同硬件都有着极好的支持, 大大提高了不同平台间的可移植性。1995年, Linux 可以在 Intel、Digital 等主流处理器上运行, 用户量超过50万。1996年, Linux 2.0 版本发布, 并支持多处理器, 此时的 Linux 进入实用阶段, 用户量已经达到350万。1998年, RedHat 公司宣布商业支持计划, 迅猛推进了 Linux 的发展, 至此 Linux 正式成为真正的服务器操作系统并继续成长。

1.2 Linux 的特点

从1991年问世到今天, Linux 在服务器、桌面、行业定制等各级领域都获得了长足的发展, 尤其在服务器领域获得了令人瞩目的成就, 被业界认为是未来最有前途的操作系统之一。在嵌入式领域, 由于 Linux 具有良好的移植性、丰富的代码资源等优点, 也受到了越来越多的关注。下面我们就来看看这个操作系统有哪些主要特点。

第一, 免费开源。Linux 是一款完全免费的操作系统, 任何人都可以从网络上下载到它的源代码, 并可以根据自己的需求进行定制化的开发, 而且没有版权限制。

第二, 模块化程度高。Linux 的内核设计分成进程管理、内存管理、进程间通信、虚拟文件系统、网络5部分, 其采用的模块机制使得用户可以根据实际需要, 在内核中插入或移走模块, 这使得内核可以被高度的剪裁定制, 以方便在不同的场景下使用。

第三, 广泛的硬件支持。得益于其免费开源的特点, 有大批程序员不断地向 Linux 社区提供代码, 使得 Linux 有着异常丰富的设备驱动资源, 对主流硬件的支持极好, 而且几乎能运行在所有流行的处理器上。

第四, 安全稳定。Linux 采取了很多安全技术措施, 包括读写权限控制、带保护的子系统、审计跟踪、核心授权等, 这为网络环境中的用户提供了安全保障。实际上有很多运行 Linux 的服务器可以持续运行长达数年而无须重启, 依然可以性能良好地提供服务, 其安全性已经在各个领域得到了广泛的证实。

第五, 多用户, 多任务。多用户是指系统资源可以同时被不同的用户使用, 每个用户对自己的资源有特定的权限, 互不影响。多任务是现代化计算机的主要特点, 指的是计算机能同时运行多个程序, 且程序之间彼此独立, Linux 内核负责调度每个进程, 使之平等地访问

处理器。由于 CPU 处理速度极快，从用户的角度来看所有的进程好像在并行运行。

第六，良好的可移植性。Linux 中 95% 以上的代码都是用 C 语言编写的，由于 C 语言是一种机器无关的高级语言，是可移植的，因此 Linux 系统也是可移植的。

1.3 系统安装

1.3.1 安装前的规划

可能会有读者正计划学习 Linux 而苦恼于不知道使用哪一个发行版，其实所有的发行版不管是 RedHat、CentOS 还是 Ubuntu，其内核都是来自 Linux 内核官网（www.kernel.org），不同发行版之间的差别在于软件管理的不同，所以不管使用哪一个发行版，只要理解其原理之后，各类发行版的区别其实不大。当然对于初学者来说，拥有广泛的学习资源也是很重要的。由于 RedHat 公司进行了大力商业推广，且得益于其成熟的认证体系，因此使用 RedHat 的用户比较多，同时，它还有丰富的相关技术文档，以及活跃的社区，所以作为入门学习，可以使用 RedHat。不过，近年来，CentOS 发展也很迅猛，这个发行版和 RedHat 几乎完全一样，而且在某些方面还比 RedHat 略胜一筹，所以在本书中后面的所有内容中将主要使用版本为 5.5 的 CentOS，小部分涉及 RedHat 的内容也将采用 5.5 版本。

有读者可能会考虑在一台计算机上安装多个操作系统，比如说在自己的家用计算机上安装 Windows 用于娱乐和日常应用或 Windows 环境下的开发等，另外再安装 Linux 系统用于学习。在这种情况下，最简单的安装方法是先安装 Windows，后安装 Linux，这样开机的时候就自动出现操作系统选择条，可以根据实际需要选择进入不同的操作系统。

由于 Linux 对系统的需求并不高，所以几乎所有计算机都可以安装，但是考虑到入门学习 Linux 需要用到图形界面，所以建议最好不要低于以下配置：CPU，P-3 800MHz；内存，1GB；硬盘，40GB。

在安装 Linux 的过程中，必须要有的两个分区为根分区（/）和 swap 分区（交换分区），当然还有一些其他的分区可以独立出来，比如说 /boot 分区、/var 分区等。

另外，这里介绍几个概念，便于大家理解后面即将出现的一些专业词汇。

什么是交换分区？交换分区是一个特殊的分区，它的作用相当于 Windows 下的虚拟内存，这个分区的大小一般设置为物理内存的两倍，但是不管物理内存有多大，交换分区建议不要超过 8GB，因为大于 8GB 的交换分区其实并没有多大实际意义。

什么是 Grub？Grub 是一个系统引导工具，通过它可以加载内核，从而引导系统启动。

什么是 /boot 分区？/boot 分区用于放置 Linux 启动所用到的文件，如 kernel 和 initrd 文件。

什么是 DHCP？DHCP 是 Dynamic Host Configuration Protocol 的简写，中文称为动态主机配置协议。在 TCP/IP 网络中，每台主机都需要有 IP 地址才能与其他主机通信，在一个大规模的网络中，如果由管理员手动地对每一台主机进行 IP 地址配置是不现实的。由此也就产生了 DHCP 协议，可用它对网络节点上的主机进行 IP 地址配置。