

UNIX 系统

李志良 邱希春 蔡建平 项国甸 薛友义 许梦杰 戴筠 编

上海科学技术文献出版社

(沪)新登字 301 号

UNIX 系统 V

李志良 邱希春 蔡建平 项国甸
薛友义 许梦杰 戴 筠 编

*

上海科学技术文献出版社出版发行
(上海市武康路2号)

全国新华书店经销

上海科技大学印刷厂印刷

*

开本 787×1092 1/16 印张 15.5 字数 386,000
1992 年 6 月第 1 版 1992 年 6 月第 1 次印刷
印数: 1-4,500

ISBN 7-80513-916-4/T.218

定价: 8.60 元

《科技新书目》258-297

前 言

当前，UNIX 这个词对于计算机文化界的人来说已是无人不知、无人不晓了。UNIX 系统，作为计算机系统软件的主流产品之一，已越来越受到人们的青睐。UNIX 系统不但为人们创造了一个舒适良好的软件开发环境，而且也为其自身的不断完善和功能的增强提供了有利条件。

UNIX System V 代表了该系统发展的一个极其重要的阶段。它保留了该系统前期发展中的大部分强有力的功能。System V 的核心部分是“小而精”的。但就在这很小的然而功能极强的机制上建立起一个高功能、高效率的工作平台，给各个领域，诸如出版界、宇航、文字处理、计算机科学及各类软件的开发，都带来了福音。

UNIX System V 提供的软件开发环境包括两大方面，一是功能强大的各个实用工具，二是把这些工具组合起来的机制。各个实用工具好比是一大堆零件，组合机制好比是安装手段，如何装配就依赖于安装者的设计，可以尽情发挥他们的创造力。本书正是从这两大方面对 UNIX 作了介绍，具体特点如下：

1. **完整性** 从系统的启动到各实用工具本书都作了详尽的介绍，所以无论是系统管理员还是编程者都能从中找到所需要的内容；

2. **实用性** 本书一切从实用出发，深入浅出地从简单实例开始，引导读者逐步进入一个更高的认识境界，不断提高使用能力。

3. **指导性** 本书并不只为介绍 UNIX 系统的使用方法而写，在介绍实用工具和组合机制的同时，还指出它们所蕴含的思想精髓，使人有恍然大悟、触类旁通之感。

因此，本书对初学者是一本入门指南，对有实践经验的用户来说，将使你对 UNIX System V 有更深刻的理解，更全面的掌握。你一定会觉得“一书在手，得心应手”。

这本书也可以作为大学本科生及研究生的教科书和参考书。

由于水平所限，错误和缺点在所难免，望读者不吝指正。

编 者

1992 年 2 月

内 容 提 要

本书以完整性、实用性和指导性为特点描述了 UNIX System V 的功能。全书分成五部分共十五章。

第一部分（第一到二章）是系统的历史和初步使用方法；第二部分（第三到六章）是系统的常用命令；包括目录和文件管理、用户通讯以及正文处理；第三部分（第七到八章）是两个编辑程序 vi 和 ed；第四部分（第九到十二章）是文档排版和高级正文处理等实用工具；第五部分（第十三到十五章）是系统的组合机制和管理维护，介绍 shell 程序设计语言、编程和维护工具。

本书可作为高等院校学生和研究生教材，也为应用人员提供了一本 UNIX 操作系统实用指南书。

目 录

第一章 引论

- §1.1 操作系统(1)
概述 UNIX System V 的主要特征
- §1.2 UNIX 系统的演化(2)
- §1.3 内容编排方式(3)
基础核心思想部分 System V 的实用程序部分

第二章 UNIX 系统初步

- §2.1 用户名和口令(5)
- §2.2 注册(5)
- §2.3 修正打错的命令(6)
抹字符 抹行符 用 stty 改变抹字符和抹行符
- §2.4 输入命令(6)
- §2.5 UNIX 系统命令的格式(7)
命令参数 加引号的命令参数
- §2.6 用 passwd 命令改变口令(8)
- §2.7 shell 使用的特殊字符(9)
特殊字符 暂停输出 中断命令
- §2.8 注销(9)
- §2.9 联机手册(10)

第三章 目录与文件

- §3.1 用 pwd 命令显示当前目录和工作目录 (11)
- §3.2 用 ls 命令显示目录中的内容 (11)
- §3.3 用 cd 命令改变工作目录 (14)
- §3.4 全路径名和相对路径名 (16)
- §3.5 文件与目录的命名规则 (17)
- §3.6 用 cat 命令查看文件内容 (18)
- §3.7 用 pg 命令查看文件内容 (19)
- §3.8 用 tail 命令显示文件的尾部 (19)
- §3.9 用 file 命令查看文件类型 (20)
- §3.10 用 mkdir 命令建立用户目录 (21)
- §3.11 用 cp 命令复制文件 (24)
- §3.12 用 mv 命令移动文件和对文件换名 (24)
- §3.13 删除文件命令 rm (26)
- §3.14 删除目录命令 rmdir (27)
- §3.15 元字符 (27)
- §3.16 文件名中的非印刷字符 (30)
- §3.17 所有权及保护 (31)
改变许可权命令 chmod 初始许可权的设置

第四章 进程和标准文件

- §4.1 标准输出的重新定向 (35)
- §4.2 标准输入的重新定向 (36)
- §4.3 重新定向时的一些注意事项 (36)
- §4.4 用 cat 建立文件的简易方法 (37)

- §4.5 用管道连接命令 (38)
- §4.6 用 tee 来查看和存储 (39)
- §4.7 诊断输出 (39)
- §4.8 后台进程 (40)
查看后台运行情况的命令——ps 降低进程优先级的命令——nice 撤消后台进程命令——kill

第五章 用户之间的通讯

- §5.1 用 write 命令发送信息 (45)
- §5.2 用 mesg 控制信息 (48)
- §5.3 使用 mail 的电子邮件系统 (49)
发送邮件 阅读邮件 转送邮件到其他用户
- §5.4 使用 mailx 的电子邮件系统 (54)
用 mailx 发送邮件 用 mailx 阅读邮件
- §5.5 用 calendar 建立工作日程 (56)
- §5.6 用 at 命令定时发送信件 (57)
查看提交的 at 作业 撤消用 at 提交的作业
- §5.7 news 系统 (60)
用 news 命令阅读新闻 添加一个新闻条款

第六章 正文处理

- §6.1 打印文件 (62)
行式假脱机打印命令——lp 准备打印文件——pr
- §6.2 分割文件——split (65)
- §6.3 正文文件的排序——sort (66)
以字母顺序排序 以数字顺序排序 排序后的储存 排序后文件的合并 合并后文件中的重行问题 排序过程中的域分隔符
- §6.4 对文件内容实施统计的命令——wc (72)
- §6.5 在一个文件中查找正文模式命令——grep (72)
检索字符串 反向检索——-v 选择项 正文模式中的正则表达式 关于正则表达式的说明 正则表达式的用法 grep 的其它选择项
- §6.6 快速查找固定字符串命令——fgrep (76)
- §6.7 用 egrep 查找一个完全的正则表达式 (77)
- §6.8 用 sed 命令进行字符串的替换 (77)
- §6.9 用 tr 命令转换字符 (78)
- §6.10 文件的重新组织——cut 和 paste (79)
用 cut 命令剪裁出列 用 cut 命令剪裁出域 用 paste 命令粘合行 在管道中使用 cut 和 paste
- §6.11 再论 pr 命令 (81)
多排打印 多排多文件打印 用 pr 命令粘合行 浏览文件
- §6.12 跟踪文件之间的差异——diff (83)
- §6.13 比较文件——cmp (88)

§6.14 查找两个文件的共同处——comm ... (89)	§8.13 ex 的改进 (121)
---------------------------------	--------------------------

第七章 全屏幕正文编辑程序 vi

§7.1 正文编辑程序 vi (91)	§7.2 vi 的启动和退出 (91)	§7.3 命令格式 (92)	§7.4 基本光标移动方法 (93)	§7.5 改变窗口位置 (94)	§7.6 命令的取消和重复 (94)	§7.7 简单编辑操作 (95)	§7.8 建立新文件 (97)	§7.9 设置编辑选择项 (97)	§7.10 基于单词的编辑操作 (99)	§7.11 基于句段的编辑操作 (100)	§7.12 基于行的编辑操作 (101)	§7.13 行号、搜索串及标记 (102)	§7.14 移动与复制 (103)	§7.15 在 vi 下执行 ex 命令 (104)
x 和 X 命令 r 和 R 命令 i、a 和 s 命令 wrapmargin 选择项 beautify 选择项 autoident 和 shiftwidth 选择项 showmatch 选择项 光标定位 单词删除 单词改变 句段定位 句段删除 句段改变 移动到指定字符和列 改变和删除行的一部分 整行操作 联结上下行 行号 搜索串 标记位置 由删除和放置操作实现移动 用取样和放置实现复制 有名缓冲区														

第八章 行编辑程序 ed 和 ex

§8.1 程序启动 (106)	§8.2 ex 选择项设置 (109)	§8.3 程序退出 (110)	§8.4 命令格式 (110)	§8.5 内容显示 (112)	§8.6 取消命令 (113)	§8.7 正文输入 (114)	§8.8 行的删除 (114)	§8.9 全局命令 (115)	§8.10 代换操作 (116)	§8.11 文件操作 (118)	§8.12 移动和复制 (119)
ed 的启动 ex 的启动 在程序中运行系统命令 选择项设置 预置文件 当前行与行号 最后行和第 0 行 搜索命令及搜索串的自动记忆 标记与寻址表达式 显示命令 步进操作 行号显示 ex 中的改进 读文件操作 更换编辑文件 写文件操作 ed 中的实现 ex 中的改进											

第九章 用 System V 进行文档排版

§9.1 启动 troff (122)	§9.2 预处理程序 (123)	§9.3 宏库 (123)	§9.4 -mm 宏库 (123)	§9.5 段落格式 (124)	§9.6 列表格式 (124)	§9.7 脚注格式 (128)	§9.8 标题 (129)	§9.9 抄录格式 (129)	§9.10 双栏格式 (131)	§9.11 改换字体和正文的尺寸 (131)
列表初始化 自动编号列表 打点列表 破折号列表 标记列表 参考列表 变列表 列表嵌套 关于列表项的进一步讨论 列表的结束 静态抄录段 浮动抄录段 表格 公式 图形 改变正文的尺寸 重音标记										

第十章 文档格式化库

§10.1 用 tbl 进行表格排版 (133)	§10.2 用 eqn 预处理器对数学公式进行排版 (136)	§10.3 pic 图形描述语言 (139)
表格位置居中 数值域的对齐 表格的标题 给表格加框线 在表格中画线 域中的正文块 改变列的格式 空格的使用 下标和上标 分数 用花括号组合复合表达式 根号 圆括号、方括号和花括号 数学符号序列 pic 的基本概念 图形的基本排版 用 pic 对图元定位 图元的相对定位 pic 宏		

第十一章 用 troff 和 nroff 进行排版

§11.1 troff 的基本概念 (142)	§11.2 troff 命令的格式 (142)	§11.3 指定页的格式 (143)	§11.4 填充和调整正文行 (144)	§11.5 用连字号连接 (145)	§11.6 行间距 (146)	§11.7 居中和强调 (146)	§11.8 尺寸和字体 (147)
页偏置的设定 设置行长 设置页长 改变页号 产生页中断 正文块的整体性控制 不对齐正文行 指定调整方式 添充命令的启动与关闭 连接控制 控制行中断 设置行间距 产生空行 正文行居中 强调正文							

设置正文尺寸 设置垂直间距 设置字体	
§11.9 段落和缩排	(147)
编排正文行 一行的临时缩排 段落形式 表和说明	
§11.10 简单的宏和俘获	(149)
设置一个俘获	
§11.11 标题和页号	(151)
设置正文和标题的长度 在标题中得到标 号	
§11.12 特殊字符序列	(152)
§11.13 运行格式化工具	(152)
§11.14 troff 命令的选择项	(153)
只打印特定的页 指定开始页号 调用宏 库	
§11.15 改变 troff 的输入	(154)
§11.16 troff 命令中参数的单位	(156)

第十二章 高级正文处理

§12.1 字符流编辑程序 sed	(157)
在命令行上键入编辑命令 从一个文件中 接受编辑命令 抑制一般的输出 多个文 件名处理 sed 的一些应用例子 sed 中 的模式空间与保存空间	
§12.2 awk 简介	(166)
访问行中的域 预定义模式 预定义变量 用户定义的变量 操作和函数 条件模 式选择 关于 awk 的进一步研究	

第十三章 shell 程序设计语言

§13.1 注册预置文件	(171)
§13.2 shell 过程	(172)
简单 shell 过程 shell 过程的命名	
§13.3 shell 变量	(174)
预定义的 shell 变量 用命令的输出设置 shell 变量	
§13.4 shell 过程的参数	(176)
§13.5 再论 echo 命令	(177)
§13.6 shell 编程	(178)
循环控制 for 语句 条件执行 if 语句 test 命令 if 和 for 语句的组合 循环控制 while 语句 用 break 和 continue 语句变更 循环 until 语句 true 和 false 命令 选 择执行 case 语句 使用 /tmp 空间 shell 程序中的注释 处理中断 用 expr 进行算 术运算 Here 文档	
§13.7 shell 过程的输入方式	(192)
传递标准输入 用 read 编写交互式 shell 过 程	
§13.8 shell 函数	(194)
§13.9 调试 shell 过程	(196)

第十四章 System V 的编程工具

§14.1 C 语言	(198)
运行 C 编译器 连接多个目标文件	
§14.2 标准输入 / 输出 (I/O) 库	(203)
标准 I/O 库中的控制函数 标准 I/O 库中 的数据传送函数 标准 I/O 库中的格式转 换函数	
§14.3 用 ar 命令维护库	(204)
§14.4 性能监测辅助手段	(205)
用 time 命令监视运行时间 用 size 命令 得到程序的大小 建立一个程序的“剖 面图”	
§14.5 其它软件开发工具	(208)
用 lint 检查 C 语言程序 用 cb 命令美化程 序正文格式 用 cxref 交叉访问 用 cflow 分析控制流 其它编程语言	
§14.6 用 make 维护计算机程序	(211)
§14.7 用 SCCS 进行版本维护	(215)

第十五章 系统管理

§15.1 特殊用户	(224)
§15.2 启动和关闭系统	(224)
用 date 命令设置日期 进入多用户状态 系统初始化文件 关闭系统	
§15.3 帐号、用户和组	(226)
password 文件 Group 文件 在系统中增 设新用户 用 chown 和 chgrp 来改变拥有 权 用 passwd 命令修改一个用户的口令 用 chmod 设置专用的文件模式	
§15.4 用 su 命令转换成超级用户	(229)
§15.5 文件系统	(230)
根文件系统 用 mkfs 命令创建一个文件系 统 用 mount 命令安装一个文件系统 用 umount 命令卸下一个文件系统 用 sync 命令同步输入和输出	
§15.6 设备和特殊文件	(232)
§15.7 find 命令	(233)
§15.8 转储和复原	(234)
用 cpio 进行转储和复原 磁带文档程序 tar	
§15.9 用 fsck 命令维护文件系统	(237)
§15.10 其它管理	(239)
改变终端特性 用 cron 定时运行任务	
§15.11 管理联机求助手册	(240)

第一章 引论

十年前, UNIX 系统还只是在大学的计算机科学系及一些有关研究所中的少数学者范围内研究和使用的, 但十年后的今天, 它已成为一种商业实体被广泛承认。随着计算机的不断普及和 UNIX 系统市场的不断扩大, AT&T 和其它一些用户组织决定建立 UNIX 系统的标准, 开始时采用的是 System III, 现在则将 System V 作为商业标准。System V 是由美国新泽西州的贝尔实验室的计算机科学研究室开发的。由于该实验室本身工作性质所致, 该系统在开发计算机软件和管理文档这两方面都特别有效, 在处理正文文件和数字文件等方面有着丰富的工具。本章讲述计算机的整体结构、运行计算机对软件有那些需要以及 System V 是怎样满足这些需要的。

§1.1 操作系统

1.1.1 概述

一个计算机硬件系统说到底无非是由一些金属、导线、磁介质和通讯设备构成的。要让这些设备为用户做些事情, 只有通过应用程序才能完成。应用程序的规模有大有小, 小的可以是进行一个单元信息的加工, 大的可以是对用户程序进行编译, 甚至是产生一个新的操作系统。计算机硬件已经具有通讯、存储数据和信息处理的能力, 但应用软件要想有效地使用这些硬件的功能, 还必须有个“调度员”, 它竭尽全力地管理好这些硬件资源及对文件的存取和与用户的对话等。这些工作都是操作系统的职责范围。目前出现的操作系统有多种。可以说, 所有的操作系统都有某些相似的功能, 都能把一个计算机的各种硬件资源有效地配合起来去完成用户的工作。当然, 把工作完成得更快些、更容易些、更节约些是使用计算机时应首先考虑的问题。

我们所要介绍的 System V 相对来说是一个比较小的操作系统, 但这不等于说这个系统的功能不强。恰恰相反, 它的核心部分的运行机制虽然只是由很少几个原则所组成, 但把这些内容组合起来, 却能形成一个功能很强的用户工作环境。System V 可以说是由三大部分组成的, 它们是文件系统、shell 和核心。前两部分都是面向用户的, 第三部分用户是看不到的, 但却十分重要。这三部分各自的主要功能是:

- 核心 (Kernel) 管理计算机的所有硬件资源, 如磁盘、磁带、打印机、通讯线等内容。
- 文件系统 数据的组织结构。文件系统可以说是 System V 中最重要的部分, 它并不是一个简单的数据陈列室, 而是包含有一些复杂的数据存储格式和组织方法。
- shell 命令解释程序。shell 本身也是一个实用程序, 本质上它不属于系统的核心部分, 而是用户使用操作系统的界面。shell 一旦运行, 就一直恭候着终端的输入, 并不断地把用户的请求转换成系统的动作或其它实用程序的执行。

1.1.2 UNIX System V 的主要特征

System V 是一个交互式操作系统。也就是说, 每当键入一个命令后, 系统就执行该命令, 并给出响应信息, 然后继续等待下一个命令。

System V 是一个多任务的操作系统。即系统可以同时运行几个任务 (称为进程)。这样, 我们就可以把一些任务交给系统放在后台运行, 而用户就可以继续在前台做一些

事情, 不必等那几个任务完成。

System V 还是一个多用户的操作系统。即允许多个用户同时使用该系统。这个特性显然是多任务特性的自然延续, 使人们可以很容易地协同工作, 共享一些信息和软件。

1. 文件系统

System V 的文件系统是一种层次结构的系统, 或有时也称树形系统。文件系统中有一种特殊类型的文件称为目录。目录中存放的是一系列普通文件的文件名, 以及这些文件在系统中的存放位置等有关信息。此外, 目录中还可以存放另一些目录的名字, 称为该目录的子目录。子目录也还可以有自己的子目录, 如此构成了一个树形的文件存储系统。为了有效地使用 System V, 我们还必须深入了解文件系统所提供的功能, 并知道如何正确地加以使用。

2. shell

shell 实际上是一个程序。它从终端上接受用户键入的命令并进行解释, 然后把解释后的内容交给操作系统的核心, 以完成用户要做的事情。shell 可以说是沟通用户与操作系统的一个桥梁。现在的一些计算机系统中有可能同时存在几种类型的 shell, 用户可以任意选择其中之一为自己服务。

3. System V 的实用程序

System V 中除了上述的基本组成部分 (核心、shell 和文件系统) 外, 还配备了大量的实用程序 (或称工具程序), 这些实用程序为用户提供了极大的方便。

实用程序中有相当一部分是处理正文文件的。例如, `grep` 可以从指定文件中选取满足特定准则的所有行; `ed`、`sed`、`tr` 和 `awk` 可以有选择地改变一个文件的内容; `sort` 和 `uniq` 可以将文件中的行重新排序。

还有一些对正文文件进行格式化处理的实用程序, 如 `nroff` 和 `troff`, 后者可以用于激光照排, `eqn` 和 `tbl` 分别用于数学公式和表格的处理。

各种编程语言, 如 C、PASCAL、FORTRAN、SNOBOL、BASIC 等等, 都必须有相应的编译程序。`lex` 和 `yacc` 是构造其它语言产品和相应编译程序的高级工具程序。

所有这些工具程序加上上面的基本组成部分就合称为 System V。如果再应用 shell 所提供的一些功能, 还可以很容易地剪裁系统, 以适应自己的特殊需要。

§1.2 UNIX 系统的演化

UNIX 系统是一个不断发展的系统。在其问世以后, 许多组织都从不同的角度对其进行修改, 形成了各种不同的版本, 例如有: UNIX, Version 3, Version 6, PWB/UNIX, IS/1, Version 7, XENIX, 4.1 BSD, System III, 4.2 BSD, Sun, System V 等。

UNIX 系统的早期几个版本都是 Bell 实验室内使用的。70 年代初期, Bell 实验室把这个系统介绍给了一些大学, 而大学中的计算机专业毕业生又把这个系统传播到各工业部门。

70 年代中叶, IS/1 的出现是 UNIX 系统的重要里程碑, 可以说是第一个商品化的 UNIX 系统。尔后, 由 Microsoft 公司研制的 XENIX 系统又是一个重要的商品化系统。

在科学界和军界, 4.1 和 4.2 BSD 是 DEC VAX 系列计算机的标准操作系统。由此又导出许多基于微机的版本, 并都已成为商品。

UNIX 系统被如此广泛地采用, 其原因何在呢? 为此, 我们可以把它与一些主要的分时系统进行比较。

许多情况下, 用户使用一个多用户的计算机系统, 主要是想共享信息而不是机时。大多数分时系统由于有严格的保密措施, 而使得这种共享十分困难, 有时甚至是不可能的。System V 则为在某个课题下一起进行工作的用户提供了一个良好的共享环境。当然, System V 也有保密措施, 但并不苛刻。

软件开发本身就是系统的一个特殊应用方面。尽管已经开发了许多软件包可供使用, 但许多操作系统总还是不能符合软件开发者的意愿。System V 是在计算机科学的研究环境中诞生和演化成的, 所以在开发工具方面相对要丰富些。

在多个用户共同开发软件时, 这个集体有可能需要一些新的工具或剪裁一些原有的工具。System V 的文件系统和 shell 使得这种要求可以十分容易地得到满足, 并且这些工具可以共享。而多数其它的分时系统不能做到工具的共享, 因为它们并不是为协同工作而开发的。

在一个分时系统中了解当前系统正在做什么是很重要的。System V 中对于系统正在做什么、谁在用系统以及哪个任务在等待打印机等这类情况是可以一目了然的。这显然十分有益。实践证明, 当用户能够确定系统被使用的繁忙程度时, 他们就可以重新安排对系统和资源的使用。

大多数操作系统都是为特定的机器而写的, 在不同种类的机器上移植是相当困难的。而 System V 是高度可移植的, 因为它是由 C 语言写成的。

总之, 与其它任何分时系统比较, System V 是一个易于开展工作并且帮助用户达到目的的系统。当然, 任何事物都不是完美无缺的, System V 也有它的缺点。有人认为, System V 在自定义原则方面走得太远, 不能阻止用户去作愚蠢的事情。然而, 在一般情况下, 提供帮助而不是提供障碍正是 System V 的特征所在。更为重要的是, System V 提供一个使用工具和积累工具的环境, 不管什么应用, System V 都提供了大量的工具以支持用户的处理。简单地讲, 用户所需要的任何工具, 系统都有, 如果真没有的话, 用现成的工具构造一个新的也是十分容易的。

§1.3 内容编排方式

1.3.1 基础核心思想部分

这部分知识包括第二、三、四章。这几章所阐述的是 System V 的基本思想和关键概念。在学习其它章节之前, 必须掌握这几章中所出现的概念。上机实习是学习操作系统的最有效的方式, 因此, 建议读者边看书边操作, 这可以达到事半功倍的效果。

第二章介绍的是 UNIX System V 的启动。就是说, 用户怎样才能进入 System V。这一章我们要引入 System V 的基本概念, 如注册和口令以及命令的格式。

第三章介绍的是目录和文件。介绍文件系统的层次结构, 目录的建立和删除以及目录信息的获得, 文件和目录的命名规则, 还有建立文件、复制文件、文件更名以及文件删除的一些基本命令。

第四章介绍的是进程与标准文件。该章介绍 System V 与用户终端和文件通讯的方式。在这一章, 读者还会学到, 如何才能把实用程序的输入和输出“重定向”到一个非标准位置上。如果深刻理解这三章的内容, 那么其它内容就很容易理解了。因为 System V 中的每样东西都是建立在这些基础思想上的。

1.3.2 System V 的实用程序部分

本书的其余部分都是讨论实用程序的。这一部分都是借助于一些例子由浅入深地介绍实用程序的功能以及借助于这些实用程序都可以做什么类型的事情。

第五章介绍的是通讯，即用户与用户之间用电子邮件实用程序所进行的通讯。

1. 正文处理和正文编辑

正文处理是 System V 的重要领域之一，它是由计算机科学研究领域中发展起来的。

第六章是正文管理，介绍了几个强有力的工具。在这一章中可以学到如何打印一个文件，如何从一个文件中选取一些行，对一个文件如何按你想象的方式排序等等。

第十二章是高级正文管理，介绍了 sed 字符流编辑程序和 awk “可编程报告生成程序”。

第七章介绍全屏幕编辑程序 vi。第八章介绍 ex 和 ed 行编辑程序。vi 编辑程序和 ex 编辑程序原产生于 UNIX 系统的 Berkeley 版本，但在 System V 中也好用。vi 编辑程序具有全屏幕功能，ex 只是 vi 的一个部分，提供的是行式编辑。ed 产生于 UNIX 系统的早期版本，也是行编辑程序。

2. 文档处理

文档处理是 System V 的又一个强有力的领域。虽然这部分工具都是批处理形式的，但提供生成文档的工具是极其有力的。文档处理工具分三章介绍。

第九章是应用 System V 的文档处理，介绍了 -mm 宏库的功能。该宏库是由一组“高级”命令组成的，从而形成一些处理文档的格式化工具。

第十章是文档格式化库，介绍预处理程序。这里介绍的有三个工具：

- tbl 这是一个描述制表功能的工具和语言；
- eqn 这是一个描述数学公式的工具；
- pic 这是一个作图的工具。

第十一章介绍用 troff 进行排版。这一章对正文格式程序 troff 作个简单介绍。

3. 软件开发工具

软件开发工具是 System V 中颇具特色的一部分。

第十三章介绍把 shell 当作编程语言使用的方法。这一章要讲述如何用一些命令组合来构成用户自己的命令。

第十四章介绍 System V 中编程工具。这些编程工具都是程序员使用的以加速软件开发的过程。该系统提供了一个快速 C 语言开发工具。我们还要讨论一些性能监测工具，这里有 make 和 sccs(源代码控制系统)。

4. 系统管理

第十五章我们介绍系统管理。主要介绍系统管理员在运行和维护 System V 时的日常工作内容。

第二章 UNIX 系统初步

本章介绍 UNIX System V 的一些最基本的内容。首先说明如何使用用户名和口令进行注册, 进入系统获得使用权; 其次描述 System V 的命令格式, 及利用参数功能实现命令的各种模式; 最后给出改变口令和实现注销的方法。并提供一个指南, 有助于用户利用联机手册查阅有关条目, 更好地使用 UNIX System V。

§2.1 用户名和口令

首次使用 System V 前, 每一用户都要由系统管理员分配一个用户名, 通常又称为帐号。同时在一个相关课题上工作的几个人, 可以编成一个组, 由系统分配一个组名和组标识符。另外, 任何一个 System V 中都有一个称为超级用户的特殊用户, 名为 root, 它担负着维护系统、设置用户名等系统管理职责。

除了必须拥有一个用户名外, 还可以选用一个保密口令, 以保护自己的帐号, 防止非法使用者的侵入。任何其他用户包括超级用户都不可能知道口令的内容。当你怀疑口令已被他人掌握时, 可以随时改变口令的内容。

§2.2 注册

有了用户名后就可以进入 System V, 这种进入过程叫作注册。假定所使用的终端已经联机, 则打入 RETURN(回车)键, 或按几下 BREAK 键, 最后在屏幕上应看到:

```
login:
```

此时就可以键入用户名并按一下回车键作为输入结束。接着, 如果已经设置了口令, 系统将请求打入口令, 例如:

```
login: maryann
password: wizard
```

矩形框内的字符串就是口令, 它并不回映在屏幕上。键入帐号和口令后, 系统就开始检测是否是已登记的用户以及口令是否正确, 如果用户名或口令与系统所设置的不相符合, 则注册失败, 并显示如下信息:

```
login incorrect
login:
```

这时就要重新注册。若注册成功, 则将显示一系列有关的信息, 最后是显示一个提示符 "\$", 表明系统正等待输入命令。

每当在提示符 \$ 下键入一个命令并完成运行后, 系统都将响应一个新的 \$, 等待输入新的命令。系统还允许用户选择设置自己的提示符。在本书所给出的例子中, 我们都用 \$ 作提示符。注意, 对于所有键入的命令都应以按回车 (RETURN) 键作为结束, 此时系统才能接受, 我们以后将不再说明。

另外还须注意, System V 是严格区分字母大小写的, 例如用户名是由小写字母组成时, 就不能用大写字母来键入用户名。

§2.3 修正打错的命令

当命令打错时，比较好的办法是修正命令行或者告诉系统放弃这个命令，具体方法是采用输入抹字符和抹行符来实现的。

2.3.1 抹字符

输入抹字符可以抹去先前键入的字符。在 System V 中，通常用回空符作为抹字符，在终端键盘上用 Backspace 标志，也有用左箭头“←”标志的。控制符 CTRL-H 表示同时按 CTRL 键和 H 键所输入的字符，它等价于 Backspace。输入后的效果为把终端屏幕上光标左面的字符抹去，同时光标也左移一格。在印刷文本中，习惯上用记号“^”代替 CTRL 键，如：

```
$ dste^H^H^Hate
Thu Jul 18 14:45:34 MDT 1991
$ _
```

2.3.2 抹行符

在一行末尾的抹行符表示抹去刚才所键入的整个行，也就是告诉系统不要执行刚才键入的这个命令行。通常用控制符 CTRL-X 或 CTRL-U 来表示，本书用的是 CTRL-U。抹行符之后不必键入 RETURN，系统实际上已经废除了该行上的所有内容，可以继续在这行上键入所需要的命令，如：

```
$ dste^U date
Thu Jul 18 14:45:34 MDT 1991
$ _
```

2.3.3 用 stty 改变抹字符和抹行符

可以用 stty 命令来改变系统所设置的抹字符和抹行符，比如要把抹字符改变成 CTRL-A，就要这样做：

```
$ stty erase ^A
$ _
```

如果要把抹行符变成 CTRL-B 时，则应做：

```
$ stty kill ^B
$ _
```

当然，亦可以同时改变抹字符和抹行符：

```
$ stty erase ^A kill ^B
$ _
```

但以上这种改变都是暂时的，它的有效作用仅保留到注销以前，下一次注册时，又转换成系统原来的设置。若要自动设置抹字符和抹行符，必须由注册预置文件 .profile 来完成。

§2.4 输入命令

终端显示提示符后，用户就可以输入命令请求系统执行。例如，输入 date 命令后，系统显示当前日期和时间：

```
$ date
Thu Jul 18 14:50:40 MDT 1991
$ _
```

用户的下一个命令可以是 who, 即询问当前有哪些用户在使用系统:

```
$ who
sally      tty00      Jul 9 08:30
peter      tty02      Jul 9 09:32
maryann    tty05      Jul 9 10:45
$ _
```

上面显示的信息中, 第一列表示用户名; 第二列表示终端名; 随后各列表示用户注册的日期和时间。

如果要求系统仅显示用户自己的注册信息, 则可键入如下命令:

```
$ who am i
```

§2.5 UNIX 系统命令的格式

在上述例子中我们可以见到命令的一般格式, 即由一系列被空格分隔的内容所组成, 整个命令由回车符结束, 第一个内容就是命令的名字, 后面的都是这个命令的参数。由于空格是分隔用的, 所以参数本身不能含有空格。如果必须用含有空格的参数, 那么这个参数必须用双引号或单引号括起来。

在同一命令行上可以同时键入几个命令, 中间用 “;” 号间隔, 如:

```
$ who am i ; date
```

在分号的两边都可以有空格, System V 首先执行 who 命令, 然后执行 date 命令。

2.5.1 命令参数

System V 中的命令参数基本上可以分为三类。

1. 选择项 (option)

选择项通常由一个负号 “-” 开头, 例如 “-al”, 它以某种方式控制一个命令的执行, 或者指出这个命令如何运行的细节。选择项的作用对每个命令都是不一样的。也有一些选择项是以 “+” 号开头的, 有的什么符号也没有。

2. 表达式 (expression)

用来描述字符串的组成规则。在最简单的情况下, 表达式就是字符串本身, 对于 grep 等命令而言, 表达式就可能复杂多了。

3. 文件名 (filename)

文件名是各种命令处理文件时代表文件所用的名字。文件名允许使用任何字符组成, 最长可有 14 个字符, 但要区分字母的大小写。系统在同时处理一组文件时, 还允许使用一些特殊的 “元字符” (或称 “通配字符”), 来进行文件名的匹配, 使得命令得以简化。

根据以上所介绍的各种类型参数, 命令的一般格式为:

< 命令 > < 选择项 > < 表达式 > < 文件名 >

例 1 带有两个文件名参数的 rm 命令，可删除 old.news 和 bad.news 这两个文件：

```
$ rm old.news bad.news
$ _
```

例 2 带有两个选择项 (-f 和 -r) 及两个文件名参数的 rm 命令：

```
$ rm -fr goodies.c baddies.o
$ _
```

例 3 在 grep 命令中，表达式是两端加引号的字符串：

```
$ grep -v "Sally Smith" people
.....
```

2.5.2 加引号的命令参数

上面的 grep 命令例子中，参数 “Sally Smith” 是用双引号括起来的，实际上参数亦可放在单引号中。一般讲，使用哪种引号都没有关系，但如果参数中已经含有某种引号，那么你就得把该字符串放在另一种引号之中，如：

```
$ grep "Sally Smith's project" people
.....
```

另外，如果参数中含有 “\$” 时，则必须使用单引号括起来。经验表明，如果一个命令不能工作，则可尝试依次在参数上加上双引号、单引号及用反斜线 \ 对某些字符进行转义，直至该命令能正常工作为止。

§2.6 用 passwd 命令改变口令

每个用户注册时都可使用口令，如果没有口令而要加以设置，或者为保密起见要改变口令，都可使用 passwd 命令来实现。

passwd 是个交互式的命令，其运行过程如下：

```
$ passwd
changing password for maryann
Old password: 
New password: 
Re-enter new password: 
$ _
```

注意，必须把新的口令输入两次，如果两次键入的新口令不同，则系统会提示再重新输入新口令：

```
They don't match; try again
Re-enter new password: _
```

如果用户连续几次都没有正确键入口令，则会给出下面信息而退出：

```
Too many tries; try again later
$ _
```

由上可知，口令的保密性很强，稍有一点差错，就不能注册，所以必须牢牢记住自己的

口令。虽然在系统中有一个 `passwd` 文件，它含有用户名、口令等内容，但其中的口令是经过加密处理的，即该文件中出现的口令和在注册时所键入的口令没有一点共同之处。

§2.7 shell 使用的特殊字符

2.7.1 特殊字符

在 UNIX System V 中，shell 是命令解释程序，它可以交互式地解释和执行从终端键入的命令，一些字符在 shell 之下具有特殊的作用。

1. 转义符

反斜线 “\” 称为转义符，其作用为把紧跟其后的特殊字符的作用去掉。例如，系统的抹字符是 “#”，而如果需要的是字符 “#”，则必须输入 “\#” 才行。

2. 通配符

星号 “*”、问号 “?”、左方括号 “[”、右方括号 “]”、连字符 “-” 等称为通配符，用以表示一组文件名，这在第三章中详细讨论。

3. 置换符

美元符 “\$” 称为置换符，可以置换出变量的内容，详阅第十三章。

4. 终止符

控制符 CTRL-D 称为终止符，对于 shell 来说是一种 “退出” 信号，在从终端上输入一些内容时，作为输入结束的标志。

5. 暂停符

控制符 CTRL-S 称为暂停符，可以暂停终端的显示输出，详见下面的介绍。

2.7.2 暂停输出

到目前为止，我们所输入的命令都是在终端屏幕上产生输出的，其显示的内容至多只有几行。但也有可能一个命令的输出会超过一屏，为了能有足够的时间看清输出内容，希望能将输出暂停一下，等阅完后再继续显示。这种暂停输出的操作可以通过在输出过程中打入 CTRL-S 键来实现，如果要恢复输出，则可键入 CTRL-Q。

2.7.3 中断命令

如果命令的执行过程特别长，临时决定中断，则可按 BREAK 或者 RUBOUT 键，此时就可返回到系统提示符 “\$”。

中断控制符也可由命令 `stty` 来设置，例如：

```
$ stty intr ^c
```

于是中断命令的执行就可由按 CTRL-C 键来实现。

§2.8 注销

当完成工作想退出系统时，必须在提示符 “\$” 状态下按 CTRL-D 键，这样就实现了注销 (logout)。注销后，系统将终止 shell 的运行，显示出新的 “login:” 信息，等待下一个用户的注册。如果没有进行过注销，即使关掉终端，也不会终止命令的运行或终止 shell 的运行，你的帐号将一直挂在系统上。

§2.9 联机手册

UNIX System V 用户手册是典型的联机手册，如果对某个问题搞不清楚，可以在终端上显示出有关的帮助信息，获得某个命令的使用方法。手册中每一章节的典型结构为：

- **NAME(名字)**——命令的名字，简要功能说明；
- **SYNOPSIS(格式)**——命令格式，所有命令参数的位置；
- **DESCRIPTION(说明)**——命令的作用及参数如何应用；
- **FILES(文件)**——构成该命令的所有文件的名字；
- **SEE ALSO(参见)**——指出有关联的命令；
- **DIAGNOSTICS(诊断)**——讨论可能产生的诊断信息。

具体操作方法为先打入“man”，再打入要了解命令的名字。例如，若要阅读有关命令 who 的一节，可打入：

```
$ man who  
.....
```

当然，键入：

```
$ man man  
.....
```

就会显示出有关命令 man 的全部情况。